

Uniwersytet im. Adama Mickiewicza w Poznaniu  
Wydział Antropologii i Kulturoznawstwa  
Katedra Teatru i Sztuki Mediów

Anna Paprzycka

**Oprogramowanie jako system operacyjny kultury. Między historią  
a praktykami współczesności**

Software as a Cultural Operating System. Between History and Contemporary Practices

Rozprawa doktorska  
napisana pod kierunkiem  
prof. UAM dr hab. Agnieszki Jelewskiej  
z Katedry Teatru i Sztuki Mediów UAM

Poznań 2023

## OŚWIADCZENIE

Ja, niżej podpisana

ANNA PAPRZYCKA

---

doktorantka Wydziału Antropologii i Kulturoznawstwa  
Uniwersytetu im. Adama Mickiewicza w Poznaniu  
oświadczam, że przedkładaną pracę doktorską  
pt. **Oprogramowanie jako system operacyjny kultury.  
Między historią a praktykami współczesności**

---

napisałam samodzielnie.

Oznacza to, że przy pisaniu pracy, poza niezbędnymi konsultacjami, nie korzystałam z pomocy innych osób, a w szczególności nie zlecałam opracowania rozprawy lub jej istotnych części innym osobom, ani nie odpisywałam tej rozprawy lub jej istotnych części od innych osób.

Jednocześnie przyjmuję do wiadomości, że gdyby powyższe oświadczenie okazało się nieprawdziwe, decyzja o wydaniu mi dyplomu zostanie cofnięta.

---

(miejscowość, data)

---

(czytelny podpis)

## SPIS TREŚCI

|                                                                                                            |            |
|------------------------------------------------------------------------------------------------------------|------------|
| <b>Wstęp</b>                                                                                               | <b>5</b>   |
| Tematyka pracy                                                                                             | 6          |
| Przedmiot badań                                                                                            | 9          |
| Stan badań i metodologia                                                                                   | 10         |
| Struktura pracy                                                                                            | 13         |
| <b>Rozdział 1</b>                                                                                          | <b>17</b>  |
| <b>Wprowadzenie teoretyczne. Problematyka software'u</b>                                                   | <b>17</b>  |
| 1.1 Rozwój <i>software studies</i>                                                                         | 18         |
| 1.2 Stan badań nad kulturowym znaczeniem oprogramowania                                                    | 20         |
| 1.2.1 Podejście medioznawcze                                                                               | 21         |
| 1.2.2. Podejście społeczne                                                                                 | 23         |
| 1.2.3 Podejście językowe                                                                                   | 26         |
| 1.2.4. Krytyczna analiza kodu                                                                              | 27         |
| 1.2.5. Podejście przestrzenno-geograficzne                                                                 | 29         |
| 1.2.6. Podejście groźnawcze                                                                                | 31         |
| 1.2.7. Praktykowanie oprogramowania                                                                        | 32         |
| 1.2.8. Inne podejścia                                                                                      | 34         |
| 1.3 <i>Software studies</i> w Polsce                                                                       | 35         |
| 1.4 Definicja software'u                                                                                   | 39         |
| 1.4.1 Ekspresywna funkcja oprogramowania                                                                   | 39         |
| 1.4.2 Wielopoziomowa struktura oprogramowania                                                              | 41         |
| 1.4.3 Oprogramowanie jako system operacyjny kultury                                                        | 48         |
| <b>I. Kontekst historyczny dla rozdziału drugiego i trzeciego: automaty</b>                                | <b>51</b>  |
| <b>Rozdział 2</b>                                                                                          | <b>63</b>  |
| <b>Spoleczne aspekty wytwarzania oprogramowania</b>                                                        | <b>63</b>  |
| 2.1 Model – wprowadzenie                                                                                   | 65         |
| 2.2 Społeczne aspekty wytwarzania oprogramowania – trzy perspektywy                                        | 68         |
| 2.2.1 Perspektywa pierwsza. Modelowanie wobec analizy ilościowej                                           | 69         |
| 2.2.2 Perspektywa druga. Modelowanie kognitywne – relacja nauki i technologii                              | 78         |
| 2.2.3 Perspektywa trzecia. Historyjki użytkownika – znaczenie języka w procesie wytwarzania oprogramowania | 86         |
| 2.3 Związek oprogramowania z językiem naturalnym                                                           | 93         |
| 2.4 <i>EMET</i> : przykład własnej praktyki artystycznej jako metody badawczej                             | 101        |
| 2.5 Przejawy zależności pomiędzy oprogramowaniem a językiem                                                | 103        |
| 2.6 Podsumowanie                                                                                           | 106        |
| <b>Rozdział 3</b>                                                                                          | <b>110</b> |
| <b>Spoleczne aspekty interakcji z technologią cyfrową</b>                                                  | <b>110</b> |
| 3.1 Nerozerwalność strategii projektowania i interakcji z oprogramowaniem                                  | 111        |

|                                                                                                                 |            |
|-----------------------------------------------------------------------------------------------------------------|------------|
| 3.2 Afektywny wymiar relacji człowieka z technologią                                                            | 112        |
| 3.3 Tworzenie narracji o oprogramowaniu w odbiorze społecznym                                                   | 117        |
| 3.4 Co właściwie sprawdza test Turinga?                                                                         | 123        |
| 3.5 Problem braku dobrych metafor dla opisu oprogramowania                                                      | 125        |
| 3.6 Podsumowanie. Propozycja metafory oprogramowania                                                            | 136        |
| <b>II. Kontekst historyczny dla rozdziału czwartego: media masowe</b>                                           | <b>139</b> |
| Media analogowe – media masowe                                                                                  | 140        |
| Regulacje prawne                                                                                                | 143        |
| <b>Rozdział 4</b>                                                                                               | <b>146</b> |
| <b>Media społecznościowe jako narzędzie społecznej kontroli</b>                                                 | <b>146</b> |
| 4.1 Media cyfrowe – media społecznościowe                                                                       | 147        |
| 4.2 Prawo w przestrzeni cyfrowej                                                                                | 150        |
| 4.3 Cambridge Analytica: oprogramowanie jako inżynieria społeczna                                               | 152        |
| 4.4 Software jako broń                                                                                          | 159        |
| 4.5 Transparentność i edukacja jako opór                                                                        | 162        |
| 4.6 Podsumowanie                                                                                                | 166        |
| <b>III. Kontekst historyczny dla rozdziału piątego: rewolucje przemysłowe i programowalny obiekt robotyczny</b> | <b>169</b> |
| <b>Rozdział 5</b>                                                                                               | <b>177</b> |
| <b>Ekonomie oprogramowania</b>                                                                                  | <b>177</b> |
| 5.1 Współczesny użytkownik jako robotnik cyfrowy                                                                | 179        |
| 5.2 <i>Facebook Algorithmic Factory</i>                                                                         | 184        |
| 5.3 <i>The Sheep Market</i>                                                                                     | 189        |
| 5.4 Cyfrowe fabryki niematerialnej pracy                                                                        | 194        |
| 5.5 W poszukiwaniu alternatyw: wolne oprogramowanie                                                             | 197        |
| 5.5.1 Richard Stallman, Free Software, GNU, GPL i <i>copyleft</i>                                               | 199        |
| 5.5.2 Dmytri Kleiner, <i>venture communism</i> i <i>copyfarleft</i>                                             | 201        |
| 5.5.3 Przejmowanie wartości wolnego oprogramowania przez kapitalizm                                             | 207        |
| 5.7 Podsumowanie                                                                                                | 211        |
| <b>Podsumowanie</b>                                                                                             | <b>214</b> |
| Wnioski kończące                                                                                                | 215        |
| Dalsze perspektywy                                                                                              | 219        |
| <b>Bibliografia</b>                                                                                             | <b>224</b> |
| <b>Spis Ilustracji</b>                                                                                          | <b>243</b> |
| <b>SUMMARY</b>                                                                                                  | <b>245</b> |
| <b>STRESZCZENIE</b>                                                                                             | <b>246</b> |

**Wstęp**

## Tematyka pracy

Praca ta poświęcona została analizie oprogramowania i jego roli w kształtowaniu współczesnej kultury cyfrowej. W niniejszej rozprawie nie będę rozpatrywać oprogramowania jedynie jako problemu technologicznego, ale chciałabym przede wszystkim zwrócić uwagę na jego kulturotwórczy potencjał i to, w jaki sposób ujawnia się ono w procesach społecznych, politycznych i ekonomicznych. W tytule dysertacji ujęłam oprogramowanie jako system operacyjny kultury, ponieważ staram się pokazać, że software odgrywa kluczową rolę w rozmaitych przekształcaniach cyfrowej rzeczywistości. Przede mną inni badacze opierali swoje założenia na podobnej analogii. Lev Manovich, jeden z inicjatorów sformalizowania na gruncie akademickim nurtu *software studies* sformułował określenie „oprogramowanie kulturowe” (*cultural software*)<sup>1</sup>. W ramach tej koncepcji badacz wskazywał na szereg „akcji kulturotwórczych”, które oprogramowanie umożliwia. Jak pisał, software jest medium, za pomocą którego użytkownik tworzy wartości kulturowe, które stają się później częścią obiegu danych w cyfrowej sieci komunikacyjnej, dzięki czemu inni użytkownicy mogą się do nich odnosić i wchodzić w interakcję z nimi.<sup>2</sup> Inny inicjator nurtu *software studies*, Matthew Fuller w książce *Behind The Blip. Essays on Culture of Software*<sup>3</sup> pisał o tym, że oprogramowanie nie jest jedynie technologicznym narzędziem, ale ma charakter kulturowy.<sup>4</sup> We wprowadzającym rozdziale, zatytułowanym *Software as Culture*, poprzez analizę wytwarzania interfejsów (*Human-Computer Interface*), badacz podkreślał, wpływ oprogramowania na kulturę i odwrotnie. Fuller argumentował za tym, że projektowanie interfejsu i, co za tym idzie, interakcji technologii z użytkownikiem, jest procesem głęboko zakorzenionym w kulturze, który czerpie z narzędzi wypracowanych na gruncie psychologii, socjologii i sztuk wizualnych, po to, aby projektować możliwe zachowania użytkowników.<sup>5</sup> Badacz zaznaczał przy tym, że z drugiej strony, z perspektywy samego wytwarzania oprogramowania, proces programowania jest aktem kulturowym, ale również sam software można rozumieć jako język uwarunkowany społecznie.<sup>6</sup>

Na polskim gruncie naukowym również pojawiały się już tezy o kulturowej roli oprogramowania, co jest widoczne w publikacjach Mirosława Filiciaka dotyczących

---

<sup>1</sup> L. Manovich, *Software Takes Command*, Bloomsbury, Nowy Jork 2013 s. 20. Szczegółowo koncepcję Manovicha dotyczącą *cultural software* i *cultural actions* będę omawiać w pierwszym rozdziale przy okazji teoretycznego wprowadzenia dotyczącego dotychczasowego stanu badań.

<sup>2</sup> Tamże, s. 20-23.

<sup>3</sup> M. Fuller, *Behind The Blip. Essays on Culture of Software*, Autonomedia, Nowy Jork 2003.

<sup>4</sup> Tamże, s. 15.

<sup>5</sup> Tamże, s. 12-15.

<sup>6</sup> Tamże, s. 15-16.

kulturowego znaczenia software'u<sup>7</sup>, w dorobku naukowym Piotra Celińskiego, który przybliży kulturowo-społeczne aspekty technologii cyfrowej,<sup>8</sup> w tekstach Macieja Ożóga skupionych na krytyce sztuki nowych mediów,<sup>9</sup> w pracach Anny Nacher, która podejmuje tematy antropologii i socjologii kultury cyfrowej,<sup>10</sup> w badaniach nad wirtualnością Michała Ostrowickiego (pseudonim Sidey Myoo),<sup>11</sup> jak też w filozoficznych rozważaniach nad programowalnym charakterem kultury Rafała Ilnickiego<sup>12</sup> oraz w pracach Jana K. Argasińskiego dotyczących popularyzacji dyscypliny *software studies* na polskim gruncie naukowym.<sup>13</sup>

Jednym z ważnych pojęć, powiązanych z namysłem o oprogramowaniu, które w pewien sposób stworzyło podwaliny dla rozwoju nauk o kulturowym znaczeniu software'u, jest też pojęcie „cyberkultury”, a więc skrótu językowego od określenia „kultury cyfrowej”. Kategoria ta wskazuje na to, że postrzeganie kultury w jej cyfrowym wymiarze, którego podstawą są procesy obliczeniowe wynikające z działania oprogramowania, jest silnie ugruntowaną strategią badawczą i wynika z oglądu przemian, jakie w tym względzie dokonały się od II połowy XX wieku. Termin cyberkultura wprowadzony został już w 1963 roku przez Alice Hilton, założycielkę Institute for Cybercultural Research. Później, na początku XXI wieku, pojęcie to zostało spopularyzowane między innymi przez francuskiego socjologa, Pierre'a Levy'ego i posłużyło za podstawową kategorię opisu kondycji współczesnego społeczeństwa wobec rozwijających się technologii cyfrowych i komunikacyjnych.<sup>14</sup> W polskim kontekście naukowym kategorię cyberkultury rozpowszechnił głównie Piotr Zawojski, który opierał swoje tezy na przekonaniu o nierozzerwalności nauki, sztuki i technologii w procesach kulturotwórczych.<sup>15</sup> W książce *Cyberkultura. Syntopia sztuki, nauki i technologii* badacz traktuje dzieła sztuki jako punkt

---

<sup>7</sup> M. Filiciak, *Zajrzeć pod powierzchnię ekranu*, „Kultura Popularna” 2008, nr 4(22), s. 15-18.

<sup>8</sup> P. Celiński, *Postmedia. Cyfrowy kod i bazy danych*, Wydawnictwo Marii Skłodowskiej-Curie, Lublin 2013.

<sup>9</sup> M. Ożóg, *Życie w krzemowej klatce. Sztuka nowych mediów jako krytyczna analiza praktyk cyfrowego nadzoru*, Wydawnictwo Uniwersytetu Łódzkiego, Łódź 2018.

<sup>10</sup> A. Nacher, *Cyfrowa humanistyka na styku sztuki, nauki i technologii*, „Czas Kultury” 2015, nr 2, s. 20-26.

<sup>11</sup> S. Myoo, *Ontoelektronika*, Wydawnictwo Uniwersytetu Jagiellońskiego, Kraków 2013.

<sup>12</sup> R. Ilnicki, *Od programu komputerowego do kultury jako programu. Perspektywa studiów nad oprogramowaniem (software studies)*, „Media, Kultura, Społeczeństwo” 2011, nr 1(6), s. 27-39.

<sup>13</sup> J.K. Argasiński, *Software Studies. Kultura, sztuka i oprogramowanie*, 2014, online: [https://ruj.uj.edu.pl/xmlui/bitstream/handle/item/41743/Argasinski\\_Software\\_studies\\_Kultura\\_sztuka\\_i\\_oprogramowanie\\_2014.pdf?sequence=1&isAllowed=y](https://ruj.uj.edu.pl/xmlui/bitstream/handle/item/41743/Argasinski_Software_studies_Kultura_sztuka_i_oprogramowanie_2014.pdf?sequence=1&isAllowed=y), dostęp: 06.11.2023; J.K. Argasiński, *Software Studies – co to takiego?*, rozm. przepr. A. Marjankowska, E. Drygalska, 10.2016, online: <https://www.miesiecznik.znak.com.pl/software-studies-co-to-takiego>, dostęp: 06.11.2023.

<sup>14</sup> Zob. P. Levy, *Cyberculture*, University Minnesota Press, Minneapolis 2001.

<sup>15</sup> P. Zawojski, *Cyberkultura. Syntopia sztuki, nauki i technologii*, Wydawnictwo Poltext, Warszawa 2010.

wyjścia do analizowania procesów technologicznych i społecznych, które z kolei traktuje jako zasadnicze składowe kultury.

W pracy tej będę prezentować podejście badawcze, służące analizowaniu wybranych aspektów praktyk kulturowych, społecznych czy ekonomicznych, powstałych na skutek i jako efekt działania oprogramowania. Oczywiście, nie wykluczam istnienia innych procesów kulturotwórczych, które lokują się poza technologią cyfrową, jednak w ramach prowadzonych przeze mnie rozważań skupiać się będę na określonej części szerokiej kategorii kultury, która bezpośrednio związana jest z kondycją współczesnego społeczeństwa cyfrowego. Należy tu jasno podkreślić, że odnosić się będę do tego rodzaju kultur i społeczeństw, które infrastrukturalnie powiązane są z cyfrowymi technologiami charakterystycznymi dla kapitalizmu platformizacyjnego, nie zaś do kultury jako ogólnego pojęcia.

Przywołana w tytule pracy koncepcja oprogramowania jako systemu operacyjnego kultury ma wskazywać na to, że współcześnie software stał się na tyle istotnym elementem rzeczywistości, że nie można prowadzić badań dotyczących praktyk społeczno-kulturowych, usytuowanych w ramach tego paradygmatu, z jego pominięciem. Zdaję sobie sprawę również z tego, że systemy operacyjne nie wyczerpują definicji oprogramowania, ponieważ to szerokie pojęcie obejmuje również instrukcje i procedury zapisane w kodzie komputerowym, ale też aplikacje, programy czy nawet złośliwe oprogramowanie. Wykorzystane w tytule pracy pojęcie systemu operacyjnego, ponieważ odnosi się ono najpełniej do funkcji oprogramowania, do których będę nawiązywać w rozprawie, a więc do aktualizacji, wykonywania, przetwarzania i rejestrowania wartości kulturowych.

Prowadząc refleksję na temat współczesnych fenomenów cyfrowej kultury będę również budować analogię do procesów kulturowych mających miejsce przed początkiem epoki cyfrowej. Tym samym będę starała się pokazać, że opisywane przeze mnie współczesne zjawiska kulturowe są nie tylko bezpośrednim efektem działania oprogramowania, ale wiele z ich aspektów zostało zainicjowanych jeszcze przed pojawieniem się pierwszych komputerów. Co oznacza, iż wiele procesów kulturowych, ekonomicznych czy medialnych antycypowało i doprowadziło do powstania koncepcji oprogramowania jako systemu operacyjnego dla kultury technologicznej. A zatem, oprogramowanie pełni współcześnie rolę systemu operacyjnego kultury, który pośredniczy w kontynuacji i rozwoju uformowanych również w przeszłości zjawisk kulturowych. W tym kontekście oprogramowanie przetwarza i aktualizuje zastane wartości kulturowe.

## Przedmiot badań

Przedmiotem badań niniejszej pracy jest oprogramowanie oraz sposoby, w jakie manifestuje się ono w kulturze poprzez oddziaływanie na zjawiska społeczne, polityczne i ekonomiczne. Analizie i interpretacji poddawane będą w tej pracy systemy cyfrowe i aplikacje, praktyki artystyczne dotyczące oprogramowania, ale także dzieła kultury, takie jak film czy przykłady sztuki medialnej. Oprogramowanie nie jest prostym do zdefiniowania przedmiotem badań, przez wzgląd na jego ontologiczną niejednorodność i wielopoziomowy charakter jego budowy. Należy podkreślić, że w pracy będę zamiennie stosowała określenie oprogramowanie i jego angielskojęzyczny odpowiednik *software*<sup>16</sup>, mocno utrwalaony już w użyciu w polskim języku. Na gruncie nauk ścisłych istnieje wiele definicji tego pojęcia<sup>17</sup>, z kolei w badaniach humanistycznych każdy naukowiec rozpoczyna refleksję od zdefiniowania tego, w jaki sposób ujmuje on *software*. Z tego powodu moją definicję oprogramowania w sposób szczegółowy przedstawiam w drugiej części pierwszego rozdziału, a tutaj, we wstępie, przedstawię ją przekrojowo. Opiera się ona na trzech fundamentalnych założeniach dotyczących oprogramowania.

Po pierwsze, zakładam, że oprogramowanie ma charakter ekspresywny, a więc wyraża się w działaniu systemów. Ich analiza pozwala prowadzić interpretacje dotyczące ukrytego za interfejsem *software'u*. Z tak sformułowanego założenia wynikają pewne decyzje metodologiczne podjęte w tej pracy, chociażby te dotyczące sposobu prowadzenia przeze mnie refleksji, w której nie skupiam się na interpretacji na poziomie samego kodu, ale głównie na analizie manifestacji oprogramowania w działaniu systemów. Decyzja ta umotywowana jest również faktem, że w przypadku interesujących mnie zjawisk społecznych, oprogramowanie najczęściej należy do dużych firm, które traktują je jak swoją własność intelektualną i nie pozwalają na jego upublicznianie. W takich przypadkach analiza działania systemu jest jedyną drogą do interpretacji kryjącego się za nim *software'u*.

Drugim założeniem, będącym podstawą mojej definicji oprogramowania, jest teza o wielopoziomowej, hierarchicznej strukturze *software'u*, która daje się podzielić na języki niskiego i wysokiego poziomu, ale można ją również poszerzyć o poziom interfejsu i dalej

---

<sup>16</sup> W pracy przyjmuję zasadę pisania wszelkich obcojęzycznych terminów kursywą. Jedynym wyjątkiem jest słowo *software*, które zdecydowałam się nie zapisywać kursywą przez wzgląd na utrwalenie w użyciu w języku polskim i na częstotliwość wykorzystywania go w niniejszej pracy.

<sup>17</sup> Zgodnie z najbardziej powszechną definicją oprogramowanie jest po prostu zbiorem instrukcji, które przekazują komputerowi, co ma zrobić. Oprogramowanie obejmuje cały zestaw programów, procedur i procedur związanych z działaniem systemu komputerowego. Zob. *Software w: Encyclopedia Britannica*, online: <https://www.britannica.com/technology/software>, dostęp: 17.11.2023.

o to, w jaki sposób oprogramowanie oddziałuje na kulturę. Wartości wyrażone na różnych poziomach tej struktury są jakościowo równe. Przyjęcie założenia o warstwowej strukturze oprogramowania jest kolejnym argumentem za tym, że badanie sposobu działania systemów i ich kulturowych oddziaływań dostarcza tropów interpretacyjnych do wnioskowania na temat samego oprogramowania.

Trzecim założeniem, kluczowym dla zrozumienia prezentowanej przeze mnie definicji, jest teza, że oprogramowanie jako system operacyjny kultury przetwarza i aktualizuje zastane aspekty społeczne, polityczne czy ekonomiczne, co staram się przedstawić na przykładzie wybranych przeze mnie procesów kulturowych, które rozwijały się też przed rozpoczęciem ery cyfrowej. Tym samym, software przetwarza zastane wartości kulturowe i aktualizuje je we współczesnym kontekście cyfrowej rzeczywistości. Z tego założenia wynika inna metodologiczna decyzja zawarta w niniejszej pracy, w ramach której każdy rozdział posiada wprowadzenie, które przybliży kontekst historyczny omawianych w części właściwej zjawisk kulturowych. W rezultacie, każdy omawiany problem kulturowy, taki jak sprawowanie kontroli społecznej za pomocą technologii, rola modeli wyobraźniowych w procesie tworzenia technologii i interakcji z nią czy próba ustosunkowywania się członków społeczeństwa do panującego systemu ekonomicznego, interpretowany jest z perspektywy współczesnej, ale z uwzględnieniem historycznej ciągłości tych procesów i ze wskazaniem ich obecności i rozwoju w przeszłości.

## **Stan badań i metodologia**

Stosunkowo młody, bo rozwijający się od około 2008 roku, kierunek badań *software studies*, w którym sytuuję moją rozprawę doktorską, skupia się na prowadzeniu refleksji naukowej dotyczącej znaczenia oprogramowania w kulturze. Stan badań związany z tą dyscypliną szczegółowo przedstawiam w osobnym rozdziale teoretycznym, który stanowi pierwszy rozdział niniejszej dysertacji. Analiza, uwzględniająca podział na reprezentowane podejścia naukowe i stosowaną metodologię ujmowania software'u została wprowadzona w tej pracy jako osobna część, aby mogła służyć za swojego rodzaju przewodnik dla osób, które chcą zapoznać się z dotychczasowymi kierunkami i tendencjami w obszarze *software studies*. W pierwszym rozdziale przedstawiam zatem rozwój nurtu *software studies*, przywołuję serię wydawniczą opublikowaną pod hasłem *Software Studies* i jej najważniejsze publikacje, a w dalszej części przedstawiam, uwzględniając podział na podejścia badawcze,

wyznaczone do tej pory kierunki badań nad oprogramowaniem i ich reprezentantów i reprezentantek.

*Software studies* wyróżnia się brakiem jednoznacznie określonej metodologii badawczej, ponieważ zaprasza on badaczy i badaczki wszelkich dziedzin do tego, by wykorzystywali wypracowane na innym gruncie metody postępowania naukowego do prowadzenia refleksji dotyczącej funkcjonowania oprogramowania w kulturze. W ramach niniejszej dysertacji będę korzystać z metod analizy i interpretacji zjawisk kulturowych, praktyk artystycznych oraz działania systemów i aplikacji. Podążając szlakiem wyznaczonym przez badaczy i badaczki *software studies*, w poszczególnych rozdziałach pracy będę czerpać z różnych, wypracowanych do tej pory sposobów naukowego ujmowania oprogramowania. W rozdziale drugim, który problematyzować będzie społeczne aspekty procesu wytwarzania oprogramowania, będę odnosić się do założeń wypracowanych na gruncie cybernetyki, a więc nauki o działaniu systemów, po to jednak, aby pokazać społeczne konsekwencje wykorzystywania tej właśnie metodologii. Korzystać będę również z podejścia badawczego, zgodnie z którym oprogramowanie można rozumieć jako realizację niedoskonałych modeli matematycznych (O’Neil, Chun). W tym samym rozdziale będę odwoływać się również do podejść badawczych, które ujmują oprogramowanie jako język (Cockburn, Cox), przyjmując tym samym wszelkie kulturowe zależności tak postawionej definicji. W rozdziale trzecim sięgać będę po inne metodologie, aby opisać afektywne zależności procesu interakcji z technologią cyfrową. Przede wszystkim, korzystać będę z założeń, zgodnie z którymi oprogramowanie można rozumieć jako maszynę afektywną, która wzbudza w odbiorcy emocjonalną reakcję (Turkle). W czwartym rozdziale, w którym zawarta została analiza tego, w jaki sposób media społecznościowe mogą być wykorzystywane jako narzędzie politycznej kontroli, będę odnosić się do podejść, które ujmują software jako narzędzie ideologiczne i polityczne (Galloway, Chun, O’Neil). W ostatnim rozdziale analizuję to, jak oprogramowanie oddziałuje na system ekonomiczny, a w szczególności to, jak redefiniuje sposoby pracy i wytwarzania wartości. W tym celu odwoływać się będę do post-marksistowskich ujęć problematyki software’u (Kleiner, Stallman, Lazzarato). W moim podejściu metodologicznym istotne będzie również przywoływanie kontekstów historycznych, choć nie tylko związanych z rozwojem technologii cyfrowej, ale przede wszystkim skupiających się na historii procesów kulturowych, które rozwijały się przed epoką cyfrową, a które współcześnie oprogramowanie (jako system operacyjny kultury) przetwarza i aktualizuje. Osadzanie oprogramowania w kontekście historycznym jest silnie utrwalonym podejściem

naukowym (Manovich, Chun, Cox, Galloway).<sup>18</sup> Inspirując się strategią badawczą wymienionych wyżej naukowców i naukowczyń, będę skupiać się na analizie kodów kulturowych wpisanych w cyfrową rzeczywistość, a szczególnie na refleksji dotyczącej przejawiania się oprogramowania w kulturze.

Jak już starałam się zaznaczyć, *software studies* jest dziedziną otwartą na propozycje badawcze, w której ceni się przede wszystkim indywidualną perspektywę, którą wnoszą badacze i badaczki podejmujący się refleksji w tym obszarze. Chciałabym też w niniejszej pracy przedstawić moją osobistą perspektywę, a więc doktorantki-badaczki, która jest także pasjonatką kreatywnego programowania, autorką kilku instalacji artystycznych, edukatorką prowadzącą warsztaty z programowania w języku Processing oraz certyfikowaną testerką oprogramowania, która na co dzień w pracy zawodowej uczestniczy w procesie wytwarzania software'u.

Zasadniczym pytaniem, które można zadać w kontekście badań nad oprogramowaniem jest to, czy zdolności programistyczne są niezbędne do prowadzenia krytycznej refleksji na temat znaczenia software'u w kulturze? Moim zdaniem specjalistyczna wiedza informatyczna nie jest niezbędna do prowadzenia kulturowych badań nad oprogramowaniem, ale badacz i badaczka podejmujący się tego rodzaju problematyki powinni posiadać jakiegokolwiek rodzaju doświadczenie pisania kodu lub chociażby obserwowania procesu jego powstawania, czy też rozumienia sposobu jego funkcjonowania poza interfejsem. Tego rodzaju doświadczenie diametralnie zmienia postrzeganie oprogramowania, pozwala też skonfrontować wcześniejsze wyobrażenia na temat software'u, często zaczerpnięte z kultury popularnej, z faktycznym działaniem oprogramowania. Obecnie istnieje wiele narzędzi edukacyjnych, które powstało z myślą o tym, by umożliwiać osobom spoza specjalistycznego, informatycznego kręgu nabywanie zdolności programistycznych, takich jak kreatywne języki programowania Processing czy P5.js.<sup>19</sup> Kierunek rozwoju *software studies* pokazuje również, że akademicki dyskurs na temat oprogramowania prowadzony „z pozycji fotela” przestaje być wystarczający. Badacze i badaczki związani z tą dziedziną coraz częściej w ramach badań naukowych albo wchodzą

---

<sup>18</sup> Szczegółowo najważniejsze tezy wymienionych w tym podrozdziale badaczy przedstawię w pierwszym rozdziale, wprowadzającym teoretyczne podstawy badań nad oprogramowaniem oraz później, w poszczególnych rozdziałach.

<sup>19</sup> Kwestii roli jaką odgrywają zdolności techniczne w badaniach nad kulturą cyfrową i dostępnych narzędzi edukacyjnych, pozwalających poszerzać tego rodzaju kompetencje poświęciłam artykuł: A. Paprzycka, *Znaczenie praktyki dla badacza kultury cyfrowej*, „Acta Universitatis Lodzensis. Folia Litteraria Polonica” 2021, nr 60(1), s. 79–91.

we współpracę z programistami, hakerami, koderami<sup>20</sup>, albo sami praktykują aktywny sposób uprawiania nauki o oprogramowaniu poprzez programowanie jako prowadzenie badań, krytyka systemów i szukanie rozwiązań.<sup>21</sup>

## Struktura pracy

Praca jest podzielona na pięć rozdziałów. Każdy rozdział, z wyjątkiem pierwszego, poprzedzony jest wprowadzeniem, które przybliży kontekst historyczny dla procesów kulturowych w nim omawianych. Rozdział drugi i trzeci posiadają wspólne wprowadzenie, ponieważ są one ze sobą mocno powiązane.

Rozdział pierwszy stanowi teoretyczne wprowadzenie, przedstawia dotychczasowy stan badań nad kulturowym znaczeniem oprogramowania, a także rozwój nurtu *software studies*, prezentuje sylwetki i tezy najważniejszych badaczy i badaczek z tego obszaru, a także podsumowuje stan badań na polskim gruncie naukowym. W drugiej części rozdziału przedstawiona została definicja oprogramowania, która stanowi podstawę rozważań zawartych w dalszej części pracy.

Rozdział drugi i trzeci ukazują dwie strony tego samego problemu – wpływu ludzkiej świadomości, wyobrażeń, i strategii poznawczych na proces wytwarzania i interakcji z oprogramowaniem. Z tego powodu oba rozdziały posiadają wspólną wprowadzającą część historyczną, która poświęcona jest historii automatów i mechanizmów z XVIII wieku, mających na celu imitację życia i ludzkich zdolności intelektualnych. Na przykładzie mechanicznej „Kaczki” Jacquesa Vaucansona i „Turka” Wolfganga von Kempelena przedstawione tu zostało to, w jaki sposób wynalazki funkcjonują nie tylko na poziomie technologicznym, ale również społecznym. Kontekstowa część historyczna ma na celu pokazanie tego, jak ludzkie wyobrażenia i przekonania na temat technologii mogą warunkować postęp technologiczny. Rozważania zawarte w rozdziale drugim oparte zostały na kategorii modelu. W ramach definiowania tego pojęcia przedstawiona została perspektywa rozwoju cybernetyki i jej wpływu na systemowe podejście w różnych dziedzinach naukowych – ścisłych i humanistycznych. Pojęcie modelu zostało tu

---

<sup>20</sup> Przykładem takiej współpracy jest publikacja: G. Cox, A. McLean, *Speaking Code. Coding as Aesthetic and Political Expression*, The MIT Press, Cambridge, Mass. 2013.

<sup>21</sup> Badaczami i badaczkami, którzy łączą praktykę programistyczną z teoretycznym dyskursem są chociażby Alexander Galloway, Winnie Soon, Jichen Zhu, Adrian Mackenzie, Vladan Joler, Paweł Janicki, Michał Szota.

wykorzystane w celu opisanego procesu przenoszenia się treści kulturowych w postaci wyobrażeń i przekonań na temat świata do warstwy technologicznej. Proces modelowania został zanalizowany z uwzględnieniem trzech perspektyw: perspektywy prawnej i problemu modelowania wobec narzędzi statystycznych i analizy ilościowej, perspektywy związanej z modelowaniem kognitywnym, która ma na celu podkreślenie relacji między wytwarzaniem wiedzy przez naukę a wytwarzaniem technologii oraz perspektywy znaczenia języka naturalnego w procesie produkcji oprogramowania. Konsekwencją procesu modelowania jest to, że oprogramowanie przez swoje działanie może realizować stereotypy lub doprowadzać do sytuacji dyskryminujących czy krzywdzących. W rozdziale zebrane zostały przykłady tego, w jaki sposób uprzedzenia funkcjonujące w nieświadomości twórców oprogramowania przenikają do samych programów, i są w dalszej kolejności egzekwowane przez samo działanie systemu. W tym rozdziale przywołane zostały też znane i dobrze opisane zagadnienia z dziedziny cybernetyki i kognitywistyki, jednak zdecydowałam się użyć ich na nowo, aby pokazać korzenie problematyki projektowania modeli i działania systemów w kontekście oprogramowania. Celem przywoływania tak dobrze utrwalconych rozpoznań jest pokazanie ciągłości myśli naukowej i jej zastosowania w kontekście badań nad kulturowym znaczeniem oprogramowania.

Z kolei rozdział trzeci skupia się na tym, jak treści podświadome, lęki, wyobrażenia i uprzedzenia są projektowane na technologię przez użytkowników podczas interakcji z nią. Rozdział poświęcony jest społecznym aspektom usytuowanym po stronie użytkownika oprogramowania. Omówiony w nim został problem nawiązywania afektywnej relacji pomiędzy człowiekiem a technologią cyfrową oraz analiza tego, jak konstruują się narracje społeczne z nią związane. W odniesieniu do testu stworzonego przez Alana Turinga, przedstawione tu zostało to, że interakcja z użytkownikiem jest kluczowym elementem wartościującym i warunkującym rozwój niektórych technologii cyfrowych. Część ta została zwieńczona rozważaniami na temat problemu braku dobrych metafor oprogramowania, a więc takich, które rozpowszechniałyby jego właściwe działanie i wpływ społeczny, a nie powielalyby wytworzone na gruncie kultury popularnej lęk i wyobrażenia. Przez siłę oddziaływania nieprawidłowych metafor, rozpowszechnionych przez kulturę popularną, użytkownicy często nie posiadają wiedzy na temat faktycznego działania technologii, ale bazują na współdzielonych przekonaniach, lękach i wyobrażeniach.

Czwarty rozdział przedstawia polityczne zależności pomiędzy społeczeństwem, oprogramowaniem a jednostkami sprawującymi kontrolę, takimi jak rząd oraz wielkie koncerny informatyczne. Historyczna część wprowadzająca do rozdziału została

poświęcona masowym mediom analogowym, a w szczególności radiu i skupia się przede wszystkim na analizie problematyki nadawania i odbierania informacji w epoce przedcyfrowej. Zaś sam rozdział drugi skupia się przede wszystkim na działaniu mediów społecznościowych, które powstały w celu umożliwienia komunikacji międzyludzkiej, a rozwinęły się do postaci narzędzia kontroli, manipulacji oraz niekiedy wręcz broni. Współczesne społeczeństwo okazało się być ogromnym zbiorem danych, który da się kategoryzować, filtrować, określać trendy i tendencje i poprzez takie praktyki wpływać na nie. W rozdziale przedstawione zostaną strategie wykorzystywania mediów społecznościowych w celach politycznych (takich jak chęć zdobycia wyborców, militarnych, wśród których skupię się przede wszystkim na kategorii wojny psychologicznej (PsyOps)) oraz ekonomicznych, typowych dla konsumpcyjnej, kapitalistycznej epoki. Wiodąca teza tego rozdziału opiera się na przekonaniu, że oprogramowanie jest medium dwustronnym, które może służyć jako narzędzie kontroli społecznej oraz jednocześnie dostarczać sposobów na uwolnienie się spod jej wpływu. Dlatego też rozdział zamyka przedstawienie strategii przeciwdziałania wcześniej opisanym manipulacjom.

Ostatni, piąty rozdział skupia się na określeniu roli oprogramowania w kulturze w kontekście ekonomicznym. Historyczne wprowadzenie do rozdziału poświęcone zostało zjawisku rewolucji przemysłowych. Zawarty w nim został opis tego, jak od początku XVIII wieku przekształcał się światowy model gospodarczy, co zapoczątkowane zostało dzięki serii wynalazków umożliwiających zautomatyzowaną produkcję. Rewolucje przemysłowe zainicjowały rozwój kapitalizmu przemysłowego i standaryzacje masowej produkcji wartości. Wywołały one również silną reakcję społeczną, która przejawiała się chociażby w uformowaniu się ruchu luddystów, którzy byli przeciwnikami postępu technologicznego i próbowali bronić dotychczasowych strategii wytwarzania kapitału. Kontekst powstania i rozwoju kapitalizmu przemysłowego służy w tym przypadku jako wprowadzenie do rozdziału piątego, który bada to, w jaki sposób wypracowane w tamtym czasie strategie eksploatacji pracowniczey i tendencje do maksymalizacji produkcji zyskują nowe formy manifestowania się w kulturze cyfrowej i kapitalizmie informacyjnym, dzięki zapośredniczeniu przez oprogramowanie. Rozważania dotyczące współczesności skupiają się przede wszystkim na kategorii „pracy niematerialnej” i zjawisku kapitalizacji aktywności użytkowników na masową skalę. W rozdziale zawarta została również analiza przejmowania narzędzi kapitalistycznych przez zdecentralizowane inicjatywy, proponujące nowe sposoby rozumienia własności i jej wytwarzania w ramach modelu kapitalistycznego.

Pracę zamykają wnioski, które podsumowują dotychczasowe rozważania i zestawiają ze sobą najważniejsze konkluzje, które wynikają z poszczególnych rozdziałów. Są one jednocześnie otwarciem na kolejne wątki i tematy powiązane z zagadnieniami podejmowanymi w dysertacji.

## **Rozdział 1**

### **Wprowadzenie teoretyczne. Problematyka software'u**

*Większość dyskusji o oprogramowaniu wymaga na początku znacznej ilości wyjaśnień, z powodu wielu nieporozumień i trudności związanych z tym pojęciem.*<sup>22</sup>

Alexander R. Galloway

### **1.1 Rozwój *software studies***

Nauka o oprogramowaniu (*software studies*) jest dyscypliną badawczą łączącą w sobie dorobek głównie dwóch obszarów: nauk o kulturze i informatyki. Obie te dziedziny wiedzy o charakterze interdyscyplinarnym rozwijały się intensywnie po zakończeniu drugiej wojny światowej. Zanim *software studies* ukształtowało się jako niezależna dyscyplina badawcza, w dyskursie naukowym pojawiały się postulaty, dotyczące potrzeby wprowadzenia kulturowych badań czerpiących z dorobku informatyki. Stawiał je chociażby niemiecki teoretyk mediów, Friedrich Kittler, który sformułował mocną tezę, że należy znać co najmniej dwa języki oprogramowania, by móc w ogóle mówić o tym czym jest kultura.<sup>23</sup>

Idea stworzenia *software studies*, kierunku badań, który skupiałby się na badaniu oprogramowania i jego wpływu na przemiany kulturowe, została po raz pierwszy sformułowana przez Lva Manovicha w ikonicznej książce z obszaru medioznawstwa: *Język nowych mediów*. Badacz napisał w niej, że:

Żeby zrozumieć nowe media, musimy odwołać się do informatyki. To właśnie tam znajdziemy nowe terminy, kategorie i funkcje charakteryzujące media, które stały się programowalne. Od medioznawstwa zmierzamy w stronę czegoś, co można by nazwać programoznawstwem, czyli od teorii mediów do teorii oprogramowania.<sup>24</sup>

W 2007 roku Lev Manovich wraz z Matthew Fullerem założyli w Nowym Jorku laboratorium badawcze Software Studies Initiative, a rok później wspólnie z Noah Wardrip-Fruinem zostali redaktorami serii publikacji wydawanych przez MIT Press pod szyldem *Software Studies*. Celem serii było rozpowszechnienie badań nad kulturowym

---

<sup>22</sup> A.R. Galloway, *The Interface Effect*, Polity Press, Cambridge/Malden 2012, s. 58. Tłumaczenie A.P.

<sup>23</sup> F. Kittler, „Technologies of Writing/Rewriting Technology”, *New Literary History* 1996, nr 27(4), s. 731-742.

<sup>24</sup> L. Manovich, *Język...*, dz. cyt., s. 117; Piotr Cypriański przetłumaczył tu „software studies” na „teorię oprogramowania”.

znaczeniem oprogramowania, jak również przedstawienie ich możliwych kierunków rozwoju. Wstępne założenia grupy wyrażone zostały w pierwszej publikacji z tej serii, którą było *Software Studies: A Lexicon*, gdzie zebrano teksty reprezentantów rozmaitych dziedzin, praktyków i teoretyków oprogramowania.<sup>25</sup>

Fuller podkreślał, że *software studies* ukonstytuowało się w opozycji do uogólniającego, nieprecyzyjnego mówienia o kulturze cyfrowej, które charakteryzuje się pominięciem kwestii oprogramowania w opisie zjawisk związanych ze współczesną technologią i zamiast tego posługiwaniem się ogólnikami typu „cyberprzestrzeń” czy „wirtualność”.<sup>26</sup> Metodologia przyjmowana przez badaczy *software studies*, jak już było wspomniane, stanowi rozproszoną mapę podejść i praktyk badawczych. Aby je zrozumieć niezbędne jest uprzednie określenie problemów, na których się skupiają. Oprogramowanie stanowi niełatwy do uchwycenia przedmiot badawczy, który na poziomie materialnym ma wymiar technologiczny, jednak przez swój sprawczy charakter współtworzy modele polityczne, ekonomiczne, komunikacyjne, społeczne i kulturowe. Stąd też próby analizy związanych z nim problemów, takich jak chociażby zrozumienie podstaw procesów obliczeniowych oraz tego, w jaki sposób treści kulturowe mogą wyrażać się poprzez narzędzia cyfrowe lub też tego, jak oprogramowanie kształtuje rzeczywistość. Badania te przybierają różnorodne formy i korzystają z odmiennych metod zaczerpniętych z wielu dyscyplin, takich jak chociażby w ramach nauk ścisłych inżynieria programistyczna, matematyka, fizyka, geografia, a w obrębie nauk społecznych i humanistycznych medioznawstwo, filozofia, groznawstwo, językoznawstwo, kulturoznawstwo, sztuka mediów i inne.

Pierwsze pięć publikacji wydanych w ramach serii *Software Studies* stały się konstytutywnymi dla wyznaczenia kierunku rozwoju badań całego nurtu. Należały do nich: praca zbiorowa *Software Studies: A Lexicon* z 2008 roku, *Expressive Processing: Digital Fictions, Computer Games, and Software Studies* z 2009 roku, autorstwa Noaha Wadripa-Fruina,<sup>27</sup> *Programmed Visions: Software and Memory* z 2011 roku, autorstwa

---

<sup>25</sup> *Software Studies: A Lexicon*, red. M. Fuller, MIT Press, Cambridge, Mass. 2008.

<sup>26</sup> M. Fuller, *Software Studies Methods*, w: *The Routledge Companion to Media Studies and Digital Humanities*, red. J. Sayers, Routledge, Nowy Jork 2018, s. 250. W tym samym miejscu Fuller posłużył się ciekawą paralełą porównując wyodrębnienie się *software studies* z medioznawstwa do ukształtowania się *software artu*, który opowiedział się w opozycji do sztuki mediów po to, aby podkreślić potrzebę prowadzenia krytycznej narracji wobec samego tworzywa sztuki cyfrowej, którym jest oprogramowanie.

<sup>27</sup> N. Wardrip-Fruin, *Expressive Processing: Digital Fictions, Computer Games, and Software Studies*, The MIT Press, Cambridge, Mass. 2009.

Wendy Hui Kyong Chun,<sup>28</sup> *Code/Space. Software and Everyday Life* z 2011 roku, której autorami byli Rob Kitchin i Martin Dodge<sup>29</sup> oraz *Speaking Code. Coding as Aesthetic and Political Expression*, autorstwa Geoffa Coxa i Alexa McLeana.<sup>30</sup> Szczegółowo o tezach przedstawionych przez badaczy w wyżej wymienionych publikacjach będę pisać w dalszej części rozdziału. I kiedy po kilku intensywnych latach dyskusji naukowych wydawało się, że software studies „przypiera”, bowiem niewiele nowych prac było wydawanych w tej serii, w czerwcu 2022 wydawnictwo MIT Press ogłosiło, że seria wydawnicza *Software Studies* zostanie wznowiona, a jej redaktorkami i redaktorem będą: Wendy Hui Kyong Chun, hakerka, artystka i badaczka Winnie Soon, artystka, projektantka UI, Jichen Zhu oraz obecny w poprzednim składzie redaktorskim, Noah Wardrip-Fruin.<sup>31</sup> Jest to znak, że dyskusja na temat oprogramowania wciąż nie została wyczerpana, a w najbliższych latach można się spodziewać kolejnych publikacji pod hasłem *Software Studies*.

## 1.2 Stan badań nad kulturowym znaczeniem oprogramowania

Badania nad oprogramowaniem rozwijają się dynamicznie od początku rozwoju dziedziny *software studies* i zbierają w sobie różnorodność podejść naukowych. Ta rozpiętość metodologiczna wynika z tego, że oprogramowanie stanowi niełatwy do jednoznacznego uchwycenia przedmiot badań i przenika większość aspektów kultury i ludzkiego życia. Interdyscyplinarność jest wpisana w założenia nurtu po to, aby prowadzić wieloaspektową i zróżnicowaną dyskusję na temat software’u. W tej części rozdziału chciałabym przedstawić stan badań nad oprogramowaniem, uwzględniając podział na wykorzystywane w tej dyscyplinie podejścia. Następny podrozdział ma służyć za swoistego rodzaju słownik pojęć i perspektyw badawczych ułatwiający poruszanie się po tym polu naukowym.

---

<sup>28</sup> W. Hui Kyong Chun, *Programmed Visions: Software and Memory*, The MIT Press, Cambridge, Mass. 2011.

<sup>29</sup> R. Kitchin, M. Dodge, *Code/Space. Software and Everyday Life*, The MIT Press, Cambridge, Mass. 2011.

<sup>30</sup> G. Cox, A. McLean, dz. cyt.

<sup>31</sup> K.S. Wilson, *The MIT Press relaunched the Software Studies series*, 13.01.2022, online: <https://news.mit.edu/2022/mit-press-relaunched-software-studies-series-0613>, dostęp: 06.11.2023.

### 1.2.1 Podejście medioznawcze

#### Lev Manovich

Jak już wspominałam na początku rozdziału, idea założenia nurtu *software studies* jako odrębnej dyscypliny badań wywiedziona została z obszaru medioznawstwa, a jej pomysłodawcą był Lev Manovich. W klasycznej już dziś dla nauk o mediach książce *Język nowych mediów*, Manovich sformułował pięć cech mediów cyfrowych: reprezentacja numeryczna, modularność, automatyzacja, wariacyjność i transkodowanie. Najważniejsza z perspektywy rozwoju *software studies* jest ta ostatnia – transkodowanie. Zakłada ona, że nowe media składają się z dwóch warstw: kulturowej i komputerowej i mogą być dowolnie transferowane w ich obrębie. To właśnie przy okazji omawiania tej cechy Manovich doszedł do wniosku, że medioznawstwo nie dostarcza wystarczającego aparatu teoretycznego do mówienia o kulturze cyfrowej i istnieje potrzeba powołania *software studies*.<sup>32</sup>

W 2013 roku Manovich wydał kolejną publikację, *Software Takes Command*. Jej tytuł był nawiązaniem do książki historyka, Siegfrieda Giediona *Mechanization takes command. A contribution to Anonymous History* z 1947 roku. Ważnym postulatem, jaki Manovich zawarł w tej publikacji była teza o ahistoryczności nowych mediów, a więc braku powszechnej świadomości na temat tego, jak rozwinęły się technologie cyfrowe.<sup>33</sup> Ponadto w *Software Takes Command* Manovich podzielił oprogramowanie na „szare” (*grey*) i „widoczne” (*visible*).<sup>34</sup> *Visible software* to ten, z którymi użytkownicy wchodzi w interakcję, a więc aplikacje, platformy i narzędzia cyfrowe. *Grey software*, to ten, którego nie widać i nie jest zrozumiały bez specjalistycznej wiedzy, ale też najczęściej nie jest również dostępny dla użytkowników. A to właśnie, jak przekonuje Manovich, *grey software* odpowiada za zarządzanie systemami współczesnego społeczeństwa. Badacz jednocześnie podkreśla, że w *Software Takes Command* skupia się na głębokiej analizie *visible software*.

W ramach *visible software* Manovich przedstawia kategorię „oprogramowania kulturowego” (*cultural software*)<sup>35</sup> i rozróżnia siedem „działań kulturowych” (*cultural actions*), które opierają się na wykorzystywaniu oprogramowania w sposób kulturotwórczy, a należą do nich:

---

<sup>32</sup> L. Manovich, *Język nowych mediów*, tłum. P. Cypryański, Wydawnictwa Akademickie i Profesjonalne, Warszawa 2006, s. 117.

<sup>33</sup> L. Manovich, *Software Takes Command*, dz. cyt., s. 39-42.

<sup>34</sup> Tamże, s. 21. Tłumaczenie A.P.

<sup>35</sup> Tamże, s. 20. Tłumaczenie A.P.

- 1) kreowanie artefaktów kulturowych i interaktywnych usług (*services*), które mają reprezentować idee, wierzenia i estetyczne wartości (strony www i aplikacje; edytowanie muzycznych wideo, projektowanie dodatkowego oprogramowania dla produktów);
- 2) udostępnianie, dodawanie, dzielenie się i remiksowanie tych artefaktów (lub ich części) online (np. czytanie gazety online, oglądanie youtube, dodawanie komentarza do bloga);
- 3) kreowanie i dzielenie się informacjami i wiedzą online (artykuły na wikipedii, dodawanie miejsc w Google Earth, włączając w to linki na Twitterze);
- 4) komunikowanie się z innymi ludźmi używając emaila, komunikatory, czaty tekstowe i wideo, możliwości mediów społecznościowych: posty, zaczepki, wydarzenia, tagi, oznaczanie, lokalizacja);
- 5) angażowanie się w interaktywne doświadczenia kulturowe np. gry komputerowe;
- 6) partycypowanie w online ekologii informacyjnej poprzez wyrażanie swoich preferencji i dodawanie metadanych (np. lajki na fejsbuku, tweety na twitterze);
- 7) rozwijanie oprogramowania (np. zaprogramowanie biblioteki dla Processingu, która umożliwia przesyłanie danych przez internet, zaprogramowanie nowego narzędzia dla photoshopa).<sup>36</sup>

Głównym pytaniem zawartym w tej książce jest to, czym są media po wynalezieniu oprogramowania<sup>37</sup>. Badacz stara się tu wytworzyć nowe sposoby mówienia o mediach w kontekście oprogramowania. Manovich wykorzystał termin „metamedium”, stworzony przez Alana Kay’a, w odniesieniu do komputera. Jest to medium, które zbiera w sobie wiele różnych mediów<sup>38</sup>, a więc zaliczył tutaj technologie do przetwarzania obrazu, dźwięku, maszyny obliczeniowe i inne. Mimo postulatów Manovicha o potrzebie odejścia od tradycyjnego medioznawstwa i jego niewątpliwego wkładu w rozwój *software studies*, trzeba zaznaczyć, że analizy, które zawarł w *Software Takes Command* są mocno zakorzenione w dorobku medioznawstwa.

Klasyczne dla *media studies* koncepcje Manovicha, takie jak chociażby koncepcja transkodowania, zostały później skrytykowane przez innych przedstawicieli *software studies*. Wendy Hui Kyong Chun wskazywała na płytkość terminu i zbyt mocne zakorzenienie w medioznawstwie.<sup>39</sup> Chun podkreślała, że Manovich sprowadził w ramach tego pojęcia procesy cyfrowe do translacji obrazu lub tekstu (który jest nośnikiem treści kulturowych) na kod i odwrotnie. Badaczka zaznaczyła, że proces ten jest o wiele bardziej zaawansowany i angażuje w sobie szereg nie dających się postrzegać w bezpośrednim

---

<sup>36</sup> Tamże, s. 23. Tłumaczenie A.P.

<sup>37</sup> Tamże, s. 335-337.

<sup>38</sup> Tamże, s. 101-106.

<sup>39</sup> W. Hui Kyong Chun, *On Software or the Persistence of Visual Knowledge*, „Grey Room” 2004, nr 18, s. 46.

poznaniu procesów kulturowych i społecznych. Za przykład posłużył jej program Media Player, własność firmy Windows, który nie służy wyłącznie do odtwarzania muzyki z płyt, ale poza percepcją użytkownika wysyła informacje dotyczące płyty do firmy Microsoft i informuje o używaniu pirackich plików. Jest dostępna możliwość wyłączenia tej opcji, ale trzeba być świadomym jej istnienia, żeby móc to zrobić. Warunkiem instalacji Media Playera jest zgoda na zasady używania i bezpieczeństwa, które wiążą się ze ściąganiem software'u, który może komunikować się z oprogramowaniem na komputerze użytkownika. Już na tym prostym przykładzie Chun zdołała uchwycić to, że koncepcja transkodowania w zbyt prosty sposób opisuje działanie medium i pomija kwestie społecznych zależności i oddziaływań oprogramowania.<sup>40</sup>

### 1.2.2. Podejście społeczne

#### Wendy Hui Kyong Chun

Badaczka kanadyjskiego pochodzenia, Wendy Hui Kyong Chun, prowadzi krytyczny dyskurs w kontekście cyfrowej kultury. Chun bada oprogramowanie jako problem technologiczno-społeczny i skupia się przede wszystkim na takich kategoriach jak: kontrola, tożsamość, pamięć i materialność oprogramowania. W swoich dotychczasowych publikacjach badaczka podkreślała dwoistą naturę internetu, jako siły, która zarówno wyzwala, jak i kontroluje swoich użytkowników (*Control and Freedom: Power and Paranoia in the Age of Fiber Optics* (2006)).<sup>41</sup> Podkreślała ona, że oprogramowanie nie jest prostym narzędziem, ale sposobem myślenia i mechanizmem kontroli (*Programmed Visions: Software and Memory* (2011)),<sup>42</sup> badała koncepcję nawyku w kontekście technologii cyfrowych, twierdziła, że interakcja z nowymi mediami jest nawykowa i definiuje tożsamość człowieka (*Updating to Remain the Same: Habitual New Media* (2016))<sup>43</sup> oraz, w swojej najnowszej publikacji, poświęciła uwagę kwestii dyskryminujących

---

<sup>40</sup> Tamże, s. 46.

<sup>41</sup> W. Hui Kyong Chun, *Control and Freedom: Power and Paranoia in the Age of Fiber Optics*, The MIT Press, Cambridge, Mass. 2006.

<sup>42</sup> W. Hui Kyong Chun, *Programmed Visions: Software and Memory*, The MIT Press, Cambridge, Mass. 2011.

<sup>43</sup> W. Hui Kyong Chun, *Updating to Remain the Same: Habitual New Media*, The MIT Press, Cambridge, Mass. 2016.

tendencji algorytmów przetwarzających dane (*Discriminating Data: Correlation, Neighborhoods, and the New Politics of Recognition* (2021))<sup>44</sup>.

Podjmując próby definicji oprogramowania Chun podkreślała, że software poprzez języki programowania tworzy system dowodzenia i kontroli, który dyscyplinuje zarówno swoich programistów, jak i użytkowników tworząc dyskretną sieć zależności społeczno-technologicznych.<sup>45</sup> Badaczka sformułowała ciekawą paralelę pisząc, że software należy rozumieć jako analogię ideologii.<sup>46</sup> Zarówno oprogramowanie, jak i ideologia mają bowiem mieć wspólną tendencję do tego, aby przekształcać rzeczywistość, sposoby myślenia i rozumienia świata, przy czym same pozostają w ukryciu. Podstawowym zadaniem oprogramowania jest wyrażanie treści, tworzenie interfejsów i systemów, a więc sprawianie, że pewne konceptualne wartości stają się widzialne, namacalne i sprawcze, jednak, jak podkreśla Chun, w procesie tym zachodzi pewna sprzeczność, bo samo oprogramowanie w nim niknie, ukrywa się pod powierzchnią, wymyka się bezpośredniej percepcji.<sup>47</sup> Dlatego tak ważne jest, aby prowadzić krytyczny dyskurs na temat oprogramowania i przywracać jego obecność w ludzkiej świadomości i percepcji.

W innych rozważaniach, dotyczących magicznego, przypominającego prymitywną religijność, myślenia o działaniu oprogramowania, badaczka proponuje redefinicję rozumienia tego, czym jest kod źródłowy i jaką pełni rolę w działaniu systemów cyfrowych i interakcji z użytkownikiem. Chun opisuje to, w jaki sposób programy są wykonywane przez komputer, zaznaczając, że część instrukcji zawartej w kodzie źródłowym może zostać wykonana, podczas gdy inna zostaje pominięta, czekając na określone warunki „do spełnienia”. W związku z tym, kod nie pełni roli źródła od samego początku, ale dopiero w określonym momencie wykonywania programowania. Tutaj badaczka proponuje, aby nie myśleć o kodzie źródłowym jako o źródle (*source*), ale jako o zasobie (*re-source*).<sup>48</sup> Ta pozornie prosta zmiana myślenia niesie ze sobą kilka znaczących konsekwencji w pojmowaniu tego, czym jest oprogramowanie i jak oddziałuje na kulturę. Po pierwsze, traktowanie czegoś jako źródła zakłada jego aksjomatyczny charakter, a więc w przypadku kodu, staje się on oczywistą tezą, która nie wymaga udowadniania jej prawdziwości lub poddawania jej w wątpliwość.<sup>49</sup> Chun podkreśla konieczność prowadzenia krytycznego

---

<sup>44</sup> W. Hui Kyong Chun, *Discriminating Data: Correlation, Neighborhoods, and the New Politics of Recognition*, The MIT Press, Cambridge, Mass. 2021.

<sup>45</sup> W. Hui Kyong Chun, *On Software...*, dz. cyt. s. 27-28.

<sup>46</sup> Tamże, s. 43.

<sup>47</sup> Tamże, s. 44.

<sup>48</sup> W. Hui Kyong Chun, *Programmed Visions...*, dz. cyt., s. 20. Tłumaczenie A.P.

<sup>49</sup> Tamże, s. 22, 49-50.

dyskursu o oprogramowaniu i nie traktowania go nigdy jako czegoś oczywistego i danego odgórnie. Po drugie, dopiero kiedy uzna się kod źródłowy za zasób, można postawić pytanie o to, gdzie znajduje się faktyczne źródło działania oprogramowania, a to pytanie otwiera miejsce na poszerzenie refleksji na poziomie technologicznym o poziom społecznych i kulturowych zależności oprogramowania. Właśnie w tym kontekście Chun pisała, że my, odbiorcy technologii cyfrowych, niczym wyznawcy prymitywnej religii czcimy kod źródłowy jako byt magiczny i niezrozumiały, jako źródło cyfrowej przyczynowości, podczas gdy tak naprawdę jego moc leży gdzie indziej, a konkretnie w tym, jak oddziałuje społecznie.<sup>50</sup>

### Alexander R. Galloway

Alexander R. Galloway to badacz, który wniósł wiele oryginalnych koncepcji do teorii mediów, cyfrowej kultury i krytycznej filozofii. Podobnie jak Chun, on również zainteresowany jest kategorią kontroli i społecznego oddziaływania oprogramowania. Galloway oprócz działalności naukowej jest również programistą i artystą, a więc łączy praktykę z teoretyzowaniem na temat software'u. Należy podkreślić, że jego publikacje wydawane były poza serią *Software Studies*, jednak jeśli chodzi o tezy badawcze, sytuują się one również w obrębie tej dyscypliny. W książce *Protocol: How control exists after decentralisation* (2005) badacz przedstawił podstawowy model komunikacyjny sieci internetowej – protokół TCP/IP (*Transmission Control Protocol/Internet Protocol*), jako metaforę władzy. Galloway zaprzeczył konwencjonalnemu przekonaniu, że decentralizacja jest jednoznaczna z brakiem kontroli i wolnością, podkreślając tym samym, że każde protokoły sieci internetowych wyrażają pewne formy zarządzania i władzy.

W innej publikacji *The Interface Effect* (2012) badacz skupił się przede wszystkim na zdefiniowaniu pojęcia interfejsu. Zgodnie z zaproponowaną przez niego definicją interfejs nie jest określonym elementem technologicznym, ale efektem działania wielu czynników i procesów. Interfejs jest procesem translacji w ciągłym stanie wytwarzania się.<sup>51</sup> Co ciekawe, cały drugi rozdział tej publikacji jest opracowywaniem tezy Chun na temat tego, że oprogramowanie jest analogią ideologii. Galloway zgadzał się z tezą badaczki, podkreślając przy tym, że oprogramowanie jest technologią symulacji, a więc jego podstawową zdolnością jest przybieranie dowolnej postaci. Ponadto, oprogramowanie jest z definicji formalne, a więc lepiej niż obraz czy słowa przystosowuje się do porównań strukturalnych.

---

<sup>50</sup> Tamże, s. 51.

<sup>51</sup> A.R. Galloway, *The Interface Effect*, dz. cyt., s. 33.

Dlatego właśnie łatwo jest zbudować funkcjonalną analogię pomiędzy strukturą oprogramowania a ideologią.<sup>52</sup> Co więcej, Galloway proponował, aby pojmować samą ideologię jako problematyczny, konceptualny interfejs, który podtrzymuje generowane wewnątrz siebie zagadnienia i idee.<sup>53</sup> Następnie badacz przedstawił tezę, że samo oprogramowanie jest ideologią, która stała się maszyną.<sup>54</sup> Galloway podkreślał, że software przez swój językowy charakter sprawia, że treści ideologiczne przenikają do struktur danych i logiki symbolicznej.<sup>55</sup> Konkluzją zatem jest teza, że oprogramowanie jest jednocześnie językiem i maszyną, przy czym to, co językowe i to, co maszynowe nie może zostać uznane za jednakowe.<sup>56</sup> W rozważaniach na temat językowego charakteru oprogramowania badacz zawarł porównanie do illokucyjnych aktów mowy i teorii performatywów. Jak sam zauważył, oprogramowanie zdaje się być jedynym językiem, który się sam wykonuje.<sup>57</sup> Tym samym, Galloway sytuuje się na styku badań nad społecznym i językowym aspektem oprogramowania.

### 1.2.3 Podejście językowe

#### Geoff Cox i Alex McLean

Badaniem językowej natury oprogramowania zajmował się też profesor sztuki i kultury cyfrowej, Geoff Cox. Zaprosił on programistę i artystę Alexa McLeana do wspólnego napisania książki *Speaking Code. Coding as Aesthetic and Political Expression* (2013), która opublikowana została w serii wydawniczej *Software Studies* The MIT Press. Zadaniem McLeana było napisanie fragmentów kodu, które stały się częścią publikacji. Za główną część merytoryczną książki odpowiadał Cox. Badacz, podobnie jak Galloway, podkreślał sprawczą moc oprogramowania jako języka, również porównując je do performatywów Austina.<sup>58</sup> Opisując software używał metafory samospełniającej się przepowiedni, ponieważ zawiera on w sobie opis przyszłości, który w określonych okolicznościach się urzeczywistnia.<sup>59</sup> Cox przedstawia w publikacji trzy perspektywy myślenia o społecznym oddziaływaniu oprogramowania. Pierwszą perspektywą jest to, w jaki sposób za pomocą

---

<sup>52</sup> Tamże, s. 55.

<sup>53</sup> Tamże, s. 58.

<sup>54</sup> Tamże, s. 69.

<sup>55</sup> Tamże, s. 70.

<sup>56</sup> Tamże, s. 73.

<sup>57</sup> Tamże, s. 70.

<sup>58</sup> G. Cox, A. McLean, dz. cyt., s. 35.

<sup>59</sup> Tamże, s. X.

oprogramowania można wyrażać treści kulturowe. W tej części badacz przede wszystkim przedstawiał różne praktyki artystyczne, takie jak publiczne odczytywanie kodu wirusa (Franco Berardi) czy pisanie dwuznacznego kodu, który w nazwach zmiennych przemycił manifesty, albo stara się wyrazić krytykę podziału klasowego (Graham Harwood). Drugą perspektywą jest związek oprogramowania i pracy, a więc jest to analiza ekonomicznych oddziaływań software'u w kulturze. Badacz zawarł w tej części rozważania na temat kwestii licencji oprogramowania, podkreślając przy tym performatywny charakter samego języka prawa, które zmienia informacje w towary.<sup>60</sup> Następnie Cox przedstawia różne próby tworzenia licencji alternatywnych wobec kapitalistycznego modelu ekonomicznego, jednak podkreśla przy tym, że nie są one możliwe do zrealizowania, ponieważ sam język licencji musi określić takie kapitalistyczne wartości, jak chociażby własność.<sup>61</sup> Trzecia perspektywa dotyczy oddziaływania oprogramowania w sferze publicznej. Autor analizuje tu kwestie udostępniania własnych danych na profilach społecznościowych i opisuje trudność wyrejestrowania się i usunięcia konta z tego rodzaju platform.<sup>62</sup> Podsumowując, do najważniejszych założeń Coxa należy przekonanie o językowym charakterze oprogramowania, które oddziałuje społecznie na wielu poziomach.

#### 1.2.4. Krytyczna analiza kodu

##### Mark C. Marino

Istotną propozycję badawczą dotyczącą oprogramowania sformułował również Mark C. Marino, który zdefiniował podejście *critical code studies*. W wydanej w 2020 książce *Critical Code Studies* badacz opisał metodę czytania kodu źródłowego w sposób krytyczny, a więc tak, by nie rozumieć go jedynie jako aksjomatyczny zestaw instrukcji dla komputera, ale starać się czytać oprogramowanie jak tekst kultury i szukać jego społecznych zależności i oddziaływań.<sup>63</sup> Metoda zawarta w *critical code studies* odróżnia się od innych podejść analizujących oprogramowanie jako problem społeczny tym, że schodzi na poziom samego języka programowania. Jest to nie tyle społeczna analiza samych skutków działania software'u, ale przede wszystkim interpretacja kodu, a więc wykorzystanych wartości, zmiennych, funkcji i danych. To podejście wymaga umiejętności rozumienia języka

---

<sup>60</sup> Tamże, s. 53.

<sup>61</sup> Tamże, s. 52-53.

<sup>62</sup> Tamże, s. 85-87.

<sup>63</sup> M. C. Marino, *Critical Code Studies*, The MIT Press, Cambridge, Mass. 2020.

programowania zarówno od autora, jak i czytelnika takich rozpoznań. Marino podkreśla, że kod nie powinien być tekstem, którego czytają i rozumieją jedynie programiści. Oprogramowanie jest tak istotnym elementem cyfrowej rzeczywistości, że coraz częściej czytają je chociażby prawnicy, dziennikarze, aktywiści, artyści czy badacze naukowci.<sup>64</sup>

W książce *Critical Code Studies* badacz przedstawia na przykładzie afery związanej z wyciekiem danych projektu Climategate z 2007 roku, jak ważna i potrzebna jest krytyka na poziomie kodu i danych. W przypadku Climategate ujawnienie oprogramowania wykorzystywanego do tworzenia wizualizacji danych na temat globalnego ocieplenia pozwoliło odkryć niepokojącą prawdę na temat intencjonalnego fałszowania informacji. W kodzie wykorzystano funkcję *adjust*, żeby dostosować dane tak, aby oparte na nich wizualizacje wskazywały na ocieplenie klimatyczne.<sup>65</sup> Przykład ten ma udowadniać, że dane, algorytmy i funkcje nie mogą być uznawane za oczywiste, jako matematyczne wartości, ale należy prowadzić krytyczny dyskurs na poziomie samych informacji. W kolejnych krokach badacz przedstawia również to, jak oprogramowanie może wyrażać treści kulturowe. Na przykład jak może być formą protestu, manifestować treści polityczne, autor analizuje także możliwości sztuki generatywnej.

### **Adrian Mackenzie**

Fizyk, socjolog i programista, Adrian Mackenzie należy do badaczy, którzy na początku XXI wieku popularyzowali przekonanie o tym, że oprogramowanie należy badać jako problem społeczny. W jego książce z 2006 roku, *Cutting Code: Software and Sociality*<sup>66</sup> badacz podkreślał niejednorodność, procesualność oprogramowania oraz jego społeczny wymiar. Należy podkreślić, że ta publikacja została wydana przed założeniem Software Studies Initiative i przed popularną serią wydawniczą Software Studies, dlatego Mackenziego można uznawać za jednego z prekursorów zwrotu ku oprogramowaniu w naukach o kulturze i społeczeństwie.

---

<sup>64</sup> Tamże, s. 1.

<sup>65</sup> Tamże, s. 105-128. Przykład skandalu związanego z Climategate pokazuje jak przez tego typu działania zdyskredytowany został sam dyskurs naukowy o globalnym ociepleniu. Wyciek danych został wykorzystany przez ośrodki medialne (szczególnie te bezkrytyczne i populistyczne) do tego, aby poddać w wątpliwość całokształt idei globalnego ocieplenia. Szczegółowo zjawisko to przedstawione zostało w filmie *Climategate: Science of a Scandal* z 2019 w reżyserii Steve'a O'Hagana. Na przykładzie tego wydarzenia daje się zaobserwować, jak ogromną rolę pełni oprogramowanie w procesie produkcji wiedzy. Tym bardziej istotny staje się postulat o prowadzeniu krytycznego dyskursu związanego z software'em i sposobami jego wytwarzania.

<sup>66</sup> A. Mackenzie, *Cutting code: Software and Sociality*, Peter Lang, Nowy Jork 2015.

W swojej książce Mackenzie podkreślał, że od początku lat 90., a więc od kiedy silnie rozwijają się dyscypliny naukowe związane z badaniem nowych mediów, kultury cyfrowej i wirtualnych społeczności, samo oprogramowanie nie jest uwzględniane w głównych nurtach akademickich i nieakademickich. Badacz podkreślał, że w ramach fundamentalnych koncepcji związanych z kulturą cyfrową, takich jak nowe media, kultura konwergencji czy społeczeństwo sieci, oprogramowanie stanowi wartość peryferyjną, daleki kontekst, a nie zasadniczy punkt odniesienia. Stawiał on tezę, że software'u nie można sprowadzić jedynie do poziomu technologicznego i traktować jako część maszyny lub zestaw instrukcji, oraz wskazywał na to, że oprogramowanie jest w aktywnym procesie wytwarzania relacji społecznych i kulturowych.<sup>67</sup>

Mackenzie określał software jako technologiczno-ludzką hybrydę, uwzględniając tym samym znaczenie i rolę człowieka jako twórcy i odbiorcy oprogramowania.<sup>68</sup> Badacz skupia się przede wszystkim na trzech cechach oprogramowania: sprawczości, materialności i społecznym wymiarze. Omawiał te cechy na przykładzie historii systemu Linux i języka Java. Warto tu podkreślić, że tym samym badacz wychodził naprzeciw postulatом późniejszych badaczy oprogramowania (Manovich, Chun, Cox), którzy podkreślali potrzebę tworzenia i utrwalania historii oprogramowania w powszechnej świadomości społecznej.

### 1.2.5. Podejście przestrzenno-geograficzne

#### **Rob Kitchin i Martin Dodge**

Zasadniczy wkład w rozwój myślenia o oprogramowaniu w kategoriach przestrzennych wnieśli Rob Kitchin i Martin Dodge, którzy są autorami wydanej w 2009 roku, w ramach serii *Software Studies* publikacji *Code/Space. Software and Everyday Life*. Autorzy sformułowali w niej pojęcie „kodo/przestrzeni” (*code/space*), które określało to, w jaki sposób funkcjonowanie oprogramowania wykracza poza materialność komputera i obejmuje także fizyczne, niecyfrowe obiekty i przestrzenie. Oprogramowanie oddziałuje w kodo/przestrzeni przez „zakodowane obiekty” (*coded objects*), a więc materialne artefakty, które same w sobie nie posiadają oprogramowania, jednak są przez nie zarządzane, na przykład przez chip lub kod kreskowy.<sup>69</sup> Doskonałym przykładem jest tu

---

<sup>67</sup> Tamże, s. 2-3.

<sup>68</sup> Tamże.

<sup>69</sup> R. Kitchin, M. Dodge, dz. cyt., s. 5-6. Tłumaczenie A.P.

karta kredytowa lub jakikolwiek produkt posiadający kod kreskowy, dzięki któremu staje się on częścią rozległego, cyfrowego systemu zarządzania i kontroli. Poprzez koncepcję zakodowanych obiektów autorzy przedstawiają przestrzenne działanie oprogramowania, które ma moc obdarzenia fizycznych elementów świata rzeczywistego zdolnością do raportowania oraz bycia rejestrowanymi i śledzonymi. Badacze przedstawiają więc strategię tego, w jaki sposób oprogramowanie rozszerza się poza technologię będącą jego nośnikiem i zaczyna kształtować i kontrolować przestrzeń fizyczną.

Kitchin i Dodge podkreślają to, że podstawową funkcją oprogramowania zarówno w przestrzeni geograficznej, jak i wirtualnej jest przeprowadzanie pomiarów i zbieranie danych. Badacze zaznaczają również istotną rolę wszelkich materialnych przekazników, dzięki którym możliwy jest obieg danych w systemie, a co za tym idzie, możliwość zarządzania przestrzenią. Odnoszą się tu oni do całokształtu elektronicznej infrastruktury, w której skład wchodzi kable, światłowody, sieci bezprzewodowe, systemy satelitów oraz wszystkich innych sensorów przekazników, które umożliwiają ciągłą cyrkulację danych w przestrzennym systemie oprogramowania.<sup>70</sup>

Autorzy przedstawiają to, w jaki sposób poszczególne struktury kodo/przestrzeni zagnieżdżają się w sobie nawzajem, zaczynając od zakodowanych obiektów, które łączą się w „zakodowane infrastruktury” (*coded infrastructures*), które z kolei zaczynają współoddziaływać na siebie w postaci „zakodowanych procesów” (*coded processes*), a te tworzą między sobą wieloaspektowe zależności, określone przez badaczy mianem „zakodowanych asamblaży” (*coded assemblages*).<sup>71</sup> Zakodowane asamblaże są zautomatyzowanymi, zarządzanymi przez oprogramowanie środowiskami, takimi jak magazyny, systemy transportowe, szpitale czy lotniska.

Zgodnie z przedstawioną tezą w pojmowaniu oprogramowania jako wartości przestrzennej kluczowe jest więc podkreślanie tego, w jaki sposób na poziomie materialnym oddziałuje ono na ożywione i nieożywione obiekty, które mogą funkcjonować w różnych skalach i czasach, współtworząc tym samym rozmaite formy spacialne.<sup>72</sup> Zasadnicza rola oprogramowania leży więc w określaniu warunków, zgodnie z którymi społeczeństwo i przestrzeń są przez nie wytwarzane.

---

<sup>70</sup> Tamże, s. 4-5.

<sup>71</sup> Tamże, s. 7. Tłumaczenie A.P.

<sup>72</sup> Tamże, s. 13.

## **Benjamin H. Bratton**

Inna koncepcja, która również wpisuje się w nurt ujmowania oprogramowania w kategoriach geograficzno-przestrzennych została sformułowana przez Benjamin H. Brattona. W wydanej w 2016 roku książce *The Stack. On Software and Sovereignty* badacz zaproponował wielopoziomową definicję oprogramowania, które należy rozumieć jako warstwową megastrukturę, którą można podzielić na następujące sfery: Ziemia, Chmura, Miasto, Adres, Interfejs, Użytkownik. Poziom Ziemi obejmuje kwestie geologiczne związane z zasobami energetycznymi i materialnymi. Chmura określa wszechobecne usługi cyfrowe ogromnych firm takich jak Google, Amazon czy Facebook, ze szczególnym podkreśleniem osobliwej suwerenności, która staje się ich udziałem. Miasto odnosi się do poziomu życia codziennego, zarządzania ruchem, rozbudową przestrzeni i przekazem informacji w systemie miejskim. Adres skupia się na problemie identyfikacji – na przykład, każdy interfejs sieciowy posiada swój adres IP, za pomocą którego można określić jego położenie geograficzne. Identyfikacja jest w tym przypadku formą kontroli. Interfejs dotyczy strategii wizualnych, za pomocą których oprogramowanie daje się postrzegać. Poziom użytkownika odnosi się do zjawiska mediacji i interakcji nawiązywanych pomiędzy użytkownikiem a komputerem, które uruchamiają ciąg procesów obliczeniowych i pozwalają aktywnie uczestniczyć w cyfrowej kulturze.<sup>73</sup>

Bratton proponuje aby ujmować oprogramowanie w skali planetarnej. Zaproponowany przez niego model stosu (tytułowy *stack*) przedstawia oprogramowanie jako rozległy system obejmujący Ziemię, a nawet wnikający w jej strukturę na poziomie geologicznym. Tak sformułowana definicja oprogramowania ma wyrażać współczesną kondycję antropocenu.

### **1.2.6. Podejście groznawcze**

Jeden z edytorów zarówno pierwszej, jak i drugiej serii wydawniczej *Software Studies*, Noah Wardrip-Fruin, podkreślał, że historia rozwoju oprogramowania jest bezpośrednio związana z grami, co wynika z ewolucji sztucznej inteligencji, która stanowiła istotny element gier komputerowych od początku ich istnienia. Wardrip-Fruin wychodził z założenia, że oprogramowanie jest trudne do zdefiniowania, a koncepcje z nim związane

---

<sup>73</sup> B.H. Bratton, *The Stack: On Software and Sovereignty*, The MIT Press, Cambridge, Mass. 2016.

stają się bardziej przystępne, kiedy omawia się je na przykładach. Z tego powodu wspomniana już przeze mnie książka *Expressive Processing: Digital Fictions, Computer Games, and Software Studies* z 2009 roku stanowi zbiór przykładów, *case studies*, które analizują różne obliczeniowe silniki gier. Autor wychodzi z założenia, że oprogramowanie wyraża się w działaniu systemów, do czego odnosi się tytułowe pojęcie „ekspresywne przetwarzanie” (*expressive processing*). Zgodnie z przedstawioną przez niego tezą, analiza na poziomie działania systemów, w tym przypadku gier komputerów, jest wystarczająca do interpretacji kryjącego się za interfejsem oprogramowania. Badacz ujmował gry jako obliczeniowe modele świata wytworzone przez oprogramowanie, które posiadają własną dynamikę, zasady działania i bohaterów niezależnych.<sup>74</sup>

Innym badaczem gier, który włączył w swoje rozważania analizę oprogramowania jest Ian Bogost. W publikacji z 2007 roku *Persuasive Games: The Expressive Power of Videogames* sformułował określenie retoryki proceduralnej, za pomocą którego podkreślał, że procedura programu jest zarówno ideologiczna, jak i obliczeniowa. Korzystając z założeń *Object Oriented Ontology* Bogost analizował technologiczne i filozoficzne aspekty oprogramowania w kontekście gier komputerowych.<sup>75</sup> Zgodnie z założeniami retoryki proceduralnej gry stanowią próbę symulacji realnego świata za pomocą zaprogramowanej reprezentacji, na którą składają się różne zaprogramowane procedury (choćby dynamika, mechanika, estetyka, tematyka gry). Owa procesualna reprezentacja świata, którą jest sama gra, ma perswazyjny charakter, co oznacza, że dąży do tego, aby przekonać odbiorcę o swojej prawdziwości. Jest to o tyle skuteczne, że ideologia czy filozofia prezentowana przez grę potwierdza się w ramach jej działania, w obrębie wytworzonego przez nią świata przedstawionego.<sup>76</sup> Podsumowując, retoryka proceduralna jest więc ideologicznym działaniem zaprogramowanego systemu gry, który dzięki swemu ekspresywnemu i perswazyjnemu funkcjonowaniu oddziałuje na odbiorcę.

### 1.2.7. Praktykowanie oprogramowania

Zasadniczą trudnością w prowadzeniu badań nad oprogramowaniem jest to, że umiejętność programowania nie jest powszechną zdolnością, a nawet samo doświadczenie czytania czy

---

<sup>74</sup> N. Wardrip-Fruin, dz. cyt.

<sup>75</sup> I. Bogost, *Persuasive Games: The Expressive Power of Videogames*, The MIT Press, Cambridge, Mass. 2007.

<sup>76</sup> Tamże, s. 2-3.

też obcowania z kodem nie jest popularnym zjawiskiem, przez jego niedostępność. Współcześnie można mówić o pewnego rodzaju cyfrowym analfabetyzmie – większość społeczeństwa korzysta z produktów cyfrowej kultury, ma styczność z rezultatami działania oprogramowania bez umiejętności czytania, pisanie czy rozumienia go. Może to doprowadzać to do wielu nadużyć ze strony podmiotów dystrybuujących usługi cyfrowe, o czym będę więcej pisać w kolejnych rozdziałach. Teraz jednak chciałabym podkreślić, jak bardzo istotne jest, żeby aktywiści, artyści i badacze posiadający programistyczne zdolności dzielili się opisem własnych działań, przedsięwzięć i przemyśleń.

Ciekawym przykładem teoretyzowania na temat praktyki programowania jest książka *10 PRINT CHR\$(205.5+RND(1));:GOTO 10* z 2012.<sup>77</sup> Koncepcja publikacji pod redakcją Nicka Montforta została zawarta w samym tytule – każdy z autorów miał napisać artykuł dotyczący umieszczonej w tytule linijki kodu w języku programowania BASIC, który generuje graficzny wzór o charakterze mozaiki lub labiryntu, wykorzystanego również jako okładka książki. Antologia zawiera w sobie dziesięć tekstów napisanych przez specjalistów zajmujących się praktykowaniem kreatywnego programowania oraz teoretyków zainteresowanych informatyką i kulturą cyfrową. Każdy z autorów odnosił się do zawartej w tytule linijki kodu, przedstawiając inną perspektywę myślenia o randomizacji, historii komputeryzacji, języku BASIC czy hardware ATARI, na którym program był uruchamiany. Książka ma być dowodem na to, że nawet krótki fragment oprogramowania może być punktem wyjścia do zgromadzenia wielu różnych badaczy i prowadzenia poszerzonych dyskusji na jego temat. Publikację tę można by również zaklasyfikować jako *critical code studies*, ponieważ wszelkie rozważania w niej zawarte skupiają się na analizie kodu.

Innym ważnym przykładem dokumentowania praktyk związanych z programowaniem jest publikacja, nad którą pracowali Alan F. Blackwell, Emma Cocker, Geoff Cox, Alex McLean i Thor Magnusson pod tytułem: *Live Coding: A User's Manual*.<sup>78</sup> Książka stanowi przewodnik po historii rozwoju praktyk *live-coding*, zarysowuje ich zakres i charakterystykę oraz wyznacza kierunki rozwoju tego nurtu. Programowanie na żywo jest jedną z rzadkich zestawów praktyk, które czynią oprogramowanie widocznym, pozwalają mu manifestować się poza interfejsem. Z tego powodu popularyzacja, ale też uporządkowanie i zebranie wiedzy dotyczącej tego rodzaju praktyk, jest cennym wkładem dla rozwoju wiedzy na temat kulturowego znaczenia oprogramowania. Przez wzgląd na swój performatywny charakter

---

<sup>77</sup> *10 Print Chr#36;(205.5+rnd(1));:Goto 10*, red. N. Monfort, The MIT Press, Cambridge, Mass. 2012.

<sup>78</sup> A.F. Blackwell, E. Cocker, G. Cox, A. McLean, T. Magnusson, *Live Coding: A User's Manual*, The MIT Press, Cambridge, Mass. 2022.

praktyki te mają formę efemerycznych i niepowtarzalnych spektakli, tym bardziej cenne jest ich utrwalanie chociażby przez opis.

### 1.2.8. Inne podejścia

Nurt *software studies* jest z założenia interdyscyplinarny i zaprasza badaczy wszelkich dziedzin i dyscyplin do tego, aby zaproponowali swoje własne ujęcie problematyki oprogramowania. Z tego powodu można wyróżnić wiele różnych podejść w badaniu kulturowego znaczenia oprogramowania. W niniejszej części starałam się przedstawić najważniejsze podejścia metodologiczne, jednak trzeba zaznaczyć, że nie jest to wyczerpujące rozpoznanie rozpiętości ujęć badawczych. Wśród niewymienionych do tej pory znajduje się chociażby *format studies*, a więc propozycja naukowa opierająca się na tym, by prowadzić badania od strony formatu. Badaczem realizującym ten postulat jest Jonathan Sterne, autor publikacji *MP3: The Meaning of a Format* z 2012 roku.<sup>79</sup> W książce tej autor przedstawia historię, ewolucję i kulturowe znaczenie powszechnego formatu dla plików dźwiękowych, jakim jest MP3.

Inną, ciekawą propozycją badawczą, która nie została do tej pory przeze mnie przedstawiona jest koncepcja Annette Vee zawarta w książce *Coding Literacy: How Computer Programming Is Changing Writing* z 2017 roku, w której autorka przedstawia przekształcającą się na przestrzeni ostatniego wieku relację między programowaniem a pisanem.<sup>80</sup> Badaczka analizuje historię rozwoju obu praktyk i proponuje, aby myśleć o kodzie jako o nowym rodzaju literatury, natomiast o programistach jako o pisarzach, którzy wprowadzają indywidualne wartości do szerokiego kontekstu programowania.

Z podobnej strategii, a więc opierającej się na tym, aby prześledzić historyczny rozwój oprogramowania równoległe z rozwojem innej dyscypliny, korzystał inny badacz, Warren Sack, autor publikacji *The Software Arts* z 2019 roku.<sup>81</sup> Badacz przedstawił w niej historię oprogramowania śledzoną z perspektywy rozwoju sztuk wyzwolonych i mechanicznych, zaczynając od XVIII-wiecznej Francji. Sack proponował lekturę tekstów informatycznych, a więc kodu, algorytmów i oprogramowania, która podkreśla ciągłość między prozą a programami. Innymi słowy badacz, podobnie jak Vee, proponuje, aby

---

<sup>79</sup> J. Sterne, *MP3: The Meaning of a Format*, Duke University Press, Durham 2012.

<sup>80</sup> A. Vee, *Coding Literacy: How Computer Programming Is Changing Writing*, The MIT Press, Cambridge, Mass. 2017.

<sup>81</sup> W. Sack, *The Software Arts*, The MIT Press, Cambridge, Mass. 2019.

traktować oprogramowanie jako tekst i w podobny sposób go interpretować. W tym celu autor korzystał z pojęć i kategorii pochodzących z obszaru sztuk wyzwolonych i mechanicznych, takich jak: logika, retoryka, gramatyka, algorytm czy symulacja. Sack argumentował za tym, że arytmetyzacja wiedzy, a więc rozwój wymienionych kategorii, odbywał się na długo przed rozpoczęciem epoki cyfrowej.

### 1.3 *Software studies* w Polsce

Kiedy w 2008 roku Manovich wraz z Fullerem i Wardrip-Fruinem wydali pierwszą publikację mającą na celu rozpowszechnianie nurtu *software studies*, a była nią wspomniana już *Software Studies. A Lexicon*, wydarzenie to natychmiast zyskało zainteresowanie na polskim gruncie naukowym, ponieważ jeszcze w tym samym roku wydany został specjalny numer czasopisma „Kultura Popularna” poświęcony wyłącznie zagadnieniu nowo kształtującej się dziedziny naukowej, związanej z badaniem oprogramowania. Zamieszczono w nim artykuł Mirosława Filiciaka *Zajrzeć pod powierzchnię ekranu*, który przybliżał rozwój dyscypliny i jej założenia.<sup>82</sup> W numerze zamieszczone zostało również tłumaczenie wybranych rozdziałów *Software Studies: A Lexicon* oraz tłumaczenie artykułu Alexandra R. Gallowaya *Język chce by go nie dostrzegać*<sup>83</sup>, który potem stanowił drugi rozdział jego publikacji *Efekt Interfejsu*. Numer zawierał też w sobie wywiad z Alexandrem Gallowayem *Ręce ubrudzone kodem*<sup>84</sup>, a także artykuły polskich artystów, praktyków i krytyków oprogramowania: Michała Szoty *Magiczna sztuka programowania komputerów*,<sup>85</sup> w którym autor spiera się z tezą Gallowaya, który stwierdził, że każde użycie komputera jest rodzajem programowania oraz wywiad, który Michał Szota przeprowadził z innym, znaczącym, polskim artystą medialnym, Pawłem Janickim: *Efekt widzialności*.<sup>86</sup>

Rafał Ilnicki w 2011 roku, w artykule *Od programu komputerowego do kultury jako programu. Perspektywa studiów nad oprogramowaniem (software studies)* podkreślał kulturotwórczy potencjał oprogramowania.<sup>87</sup> W tym tekście Ilnicki odwoływał się do

---

<sup>82</sup> M. Filiciak, dz. cyt.

<sup>83</sup> A.R. Galloway, *Język chce, by go nie dostrzegać*, tłum. M. Szota, „Kultura Popularna” 2008, nr 4(22), s. 23-35.

<sup>84</sup> A.R. Galloway, *Ręce ubrudzone kodem*, rozm. przepr. M. Halawa, „Kultura Popularna” 2008, nr 4(22) s. 19-22.

<sup>85</sup> M. Szota, *Magiczna sztuka programowania komputerów*, „Kultura Popularna” 2008, nr 4(22), s. 37-40.

<sup>86</sup> P. Janicki, *Efekt widzialności*, rozm. przepr. M. Szota, „Kultura Popularna” 2008, nr 4(22), s.41-43.

<sup>87</sup> R. Ilnicki, dz. cyt., s. 27-39.

definicji programowania sformułowanej przez Alexandra R. Gallowaya, która głosiła, że każde użycie komputera jest rodzajem programowania.<sup>88</sup> Badacz przedstawił tezę, zgodnie z którą użytkownicy programują kulturę poprzez interakcję z programami i innymi cyfrowymi artefaktami. Powołując się na wcześniejsze rozpoznania Siegfrieda Schmidta, autor przedstawia kulturę jako program społeczny, którego celem jest rekonstrukcja zbiorowej wiedzy.<sup>89</sup>

Niezwykle istotną publikacją, która nie tylko rozpowszechniła założenia *software studies* na polskim gruncie naukowym, ale również stanowi przykład tego, w jaki sposób realizować założenia nurtu w celu prowadzenia analizy cyfrowej rzeczywistości postmedialnej, jest książka *Postmedia. Cyfrowy kod i bazy danych* Piotra Celińskiego.<sup>90</sup> Wydana w 2013 roku publikacja dzieli się na trzy zasadnicze części. Pierwsza z nich zarysowuje krajobraz przemian w rzeczywistości postmedialnej. Badacz opisuje tu zwrot w medioznawstwie polegający na odejściu od analizy obiektu medialnego w stronę głębszego poziomu badań nad procesami obliczeniowymi. Zawarta tu została również analiza transformacji modeli komunikacyjnych od tradycyjnych, strukturalnych form po media taktyczne i społecznościowe z podkreśleniem aktywistycznego i prospołecznego charakteru nowych mediów. Celiński śledzi zachodzące na poziomie społecznym zmiany kompetencji technologicznych i komunikacyjnych wraz z przekształcaniem się wrażliwości i percepcji użytkowników mediów cyfrowych.

W drugiej części publikacji zawarta została analiza kategorii kodu. Badacz przedstawia zarówno kulturowe, jak i matematyczne teorie software'u, a następnie sam interpretuje status oprogramowania w kulturze i jego relację z językiem naturalnym. Autor publikacji interpretuje oprogramowanie jako rozległy proces, który z jednej strony, przez wzgląd na swoją językową naturę, jest uwikłany w zależności społeczne, z drugiej zaś strony badacz podkreśla jego materialność i, powołując się na tezy Kittlera, pokazuje, że podstawą procesów cyfrowych jest energia i jej przepływ w przewodach, płytkach i innych hardware'owych nośnikach. W tym ujęciu *software* jawi się jako problematyczny, wieloaspektowy przedmiot badawczy, który łączy w sobie materialność energii i niematerialność informacji. W kolejnym kroku badacz analizuje emanacje kodu, a więc to

---

<sup>88</sup> Tamże, s. 31. Należy tutaj sprostować pomyłkę autora artykułu. Ilnicki wskazuje, że przywoływaną definicję programowania Galloway sformułował w artykule *Język chce by go nie dostrzegać*, który również wcześniej już przywoływałam. Jest to pomyłka, a sformułowanie Gallowaya dotyczące tego, że każde użycie komputera jest rodzajem programowania zostało wypowiedziane w wywiadzie: A.R. Galloway, *Ręce ubrudzone kodem*, dz. cyt., s. 21.

<sup>89</sup> Tamże, s. 38.

<sup>90</sup> P. Celiński, *Postmedia...*, dz. cyt.

w jaki sposób manifestuje się on w kulturze poprzez protokoły, poszczególne warstwy kodu i oprogramowanie. Badacz przedstawia to jak samo oprogramowanie wpływa na kształtowanie się kultury.

Trzecia część publikacji odnosi się do kategorii baz danych. Autor zaczyna od wyjaśnienia podstawowych pojęć związanych z tym zjawiskiem, takich jak: dane, metadane, magazyny danych czy ekosystemy danych. Następnie badacz przywołuje metaforę drzewa życia, symbolu, który jest elementem wspólnych wielu religii, kultur i systemów politycznych, po to aby w odniesieniu do niego opisywać złożoność pojęcia baz danych i ich kulturowe znaczenie. Poprzez analogię do mitycznej figury drzewa autor pokazuje, że same bazy danych, przez swoją strukturę i sposób funkcjonowania w zbiorowej wyobraźni, stają się metaforycznym, ikonicznym i uniwersalnym symbolem cyfrowej kultury. Podobnie jak figura drzewa, która była podtrzymywana w różnych porządkach religijnych, narracyjnych i światopoglądowych, tak samo dane cyrkulują w sieci cybernetycznej przekraczając granice geograficzne, kulturowe, czaso-przestrzenne, technologiczne, społeczne, polityczne czy ekonomiczne.<sup>91</sup> W dalszej części autor analizuje procesy interpretacji danych, w ramach których tworzy się informacje lub całe narracje, takie jak *data mining* czy *data journalism*.

O publikacji tej można myśleć jako o projekcie totalnym, który sam w sobie wypełnia założenia tytułowego postmedium. Grafika będąca okładką książki jest treścią książki przetworzoną algorytmicznie do postaci wizualnej. Publikacja posiada również wersję elektroniczną, umieszczoną na stronie internetowej.<sup>92</sup> W tej postaci pozwala się ona czytać nielinearnie, używając słów kluczy lub wizualnych podpowiedzi w postaci linii metra, łączących poszczególne wątki ze sobą. Ze strony internetowej można pobrać książkę w całości lub jej fragmenty w sposób legalny, dzięki czemu przedsięwzięcie stanowi niepowtarzalny gest na rzecz popularyzacji badań nad kulturowym znaczeniem oprogramowania.

W obszarze sztuki i designu założenia *software studies* realizuje Marcin Składanek z Uniwersytetu Łódzkiego, który w swojej działalności naukowej poświęcił się przede wszystkim zagadnieniu sztuki generatywnej, a więc dzieł wytwarzanych poprzez działanie algorytmów. Badacz jest autorem książki *Sztuka generatywna. Metoda i praktyki*<sup>93</sup>, która jest bardzo ważną pozycją, dostarczającą narzędzi badawczych do analizowania cyfrowych dzieł i praktyk artystycznych z uwzględnieniem działania oprogramowania, leżącego u ich

---

<sup>91</sup> Tamże, s. 150.

<sup>92</sup> Zob. *Postmedia*, online: <http://www.postmedia.umcs.lublin.pl>, dostęp 06.11.2023.

<sup>93</sup> M. Składanek, *Sztuka generatywna. Metoda i praktyki*. Wydawnictwo Uniwersytetu Łódzkiego, Łódź 2017.

podstaw. Badacz poruszał również inną, istotną kwestię, którą jest problem społecznej odpowiedzialności w projektowaniu wizualizacji danych.<sup>94</sup>

Założenia *software studies* rezonują na polskim gruncie naukowym również w obszarze badań nad grami, czego dowodem jest praca doktorska Jana K. Argasińskiego, pracownika Zakładu Technologii Gier Wydziału Fizyki, Astronomii i Informatyki Stosowanej na Uniwersytecie Jagiellońskim oraz członka UBU Labu, a więc przedsięwzięcia, które zajmuje się demosceną i ożywianiem starych modeli komputerów, a także programów i gier, które były na nich uruchamiane. Argasiński w 2014 obronił pracę doktorską pod tytułem *Software Studies. Kultura, sztuka i oprogramowanie*.<sup>95</sup> Badacz również zajmował się popularyzacją idei *software studies* poza ściśle akademickim otoczeniem udzielając wywiadu, w którym wyjaśniał podstawowe założenia nurtu.<sup>96</sup> Innym przykładem tego, że podejście *software studies* pozostaje żywe na polskim gruncie naukowym związanym z badaniami nad grami są chociażby rozpoznania Mateusza Felczaka w *Obiekt i kod. Badanie gier wideo w perspektywie software studies oraz realizmu spekulatywnego*.<sup>97</sup>

Przywołany wybór stanowisk oczywiście nie jest wyczerpującym przedstawieniem wpływu *software studies* na badania naukowe w Polsce, ale pokazuje istotność rozwoju tej dyscypliny. Coraz więcej młodych badaczy i badaczek dołącza do grupy osób wykorzystujących metodologię badań nad kulturowym znaczeniem oprogramowania, czego dowodem jest również niniejsza praca. W Polsce powstaje też coraz więcej inicjatyw o charakterze interdyscyplinarnych ośrodków wymiany wiedzy i doświadczeń, mających na celu poszerzanie kompetencji cyfrowych.<sup>98</sup> Doskonałym przykładem takiego przedsięwzięcia jest UBU Lab w Krakowie, który współdziała z Uniwersytetem Jagiellońskim, ponieważ powołany został przy Instytucie Kultury tej uczelni w 2016 roku. Ośrodek stworzył i rozwija bibliotekę naukową dotyczącą demosceny i cyfrowej kultury oraz pracownię, która jest miejscem warsztatów, badań oraz praktyk artystycznych.<sup>99</sup> Innym, cennym przykładem medialabowej działalności jest Medialab Katowice, założony w 2012 roku, który jest aktywnie działającym ośrodkiem realizującym kreatywne inicjatywy

---

<sup>94</sup> M. Składanek, *Od Isotype do Many Eyes. Społeczna odpowiedzialność designu w praktykach wizualizacji danych*, „Przegląd Kulturoznawczy” 2020, 9(1), s. 37-54.

<sup>95</sup> J.K. Argasiński, *Software Studies. Kultura, sztuka i oprogramowanie*, dz. cyt.

<sup>96</sup> J.K. Argasiński, *Software Studies – co to takiego?*, dz. cyt.

<sup>97</sup> M. Felczak, *Obiekt i kod. Badanie gier wideo w perspektywie software studies oraz realizmu spekulatywnego*, „Przegląd Kulturoznawczy” 2016, nr 3(29), s. 305-317.

<sup>98</sup> Więcej o znaczeniu praktyk medialabowych w Polsce pisałam w artykule: A. Paprzycka, *Znaczenie praktyki dla badacza kultury cyfrowej*, dz. cyt.

<sup>99</sup> J.K. Argasiński, *Software Studies – co to takiego?*, dz. cyt.

badawcze, edukacyjne i artystyczne. Do tej pory medialab organizował liczne warsztaty, interwencje w przestrzeni publicznej, wykłady, dyskusje i festiwale. Przedsięwzięcia skupiały się przede wszystkim na analizie i wizualizacji danych, procesach projektowania i programowania interakcji wykorzystujących software i hardware.<sup>100</sup>

## 1.4 Definicja software'u

Jak wynika ze wszystkich przywołanych powyżej stanowisk badawczych i podejść metodologicznych, istotnym elementem *software studies* jest analiza i definicja tego, czym jest sam software. Definicja oprogramowania zawarta w niniejszej pracy opiera się na trzech kluczowych założeniach, wyprowadzonych z idei już utrwalonych na gruncie badań nad kulturowym znaczeniem oprogramowania. Te trzy założenia zakładają, że oprogramowanie jest a) ekspresywne, b) wielopoziomowe oraz c) jest pośrednikiem procesów kulturowych. W dalszej części tego rozdziału dokładnie rozwinę, co rozumiem przez te cechy.

### 1.4.1 Ekspresywna funkcja oprogramowania

Pierwszym założeniem, leżącym u podstaw wykorzystywanej przeze mnie definicji, jest to, że oprogramowanie posiada funkcję ekspresywną i wyraża się poprzez działanie systemów, które tworzy, a także poprzez interakcje z użytkownikami. Myśl tę sformułował już Noah Wardrip-Fruin w ramach książki *Expressive Processing: Digital Fictions, Computer Games, and Software Studies*, jednej z konstytutywnych pozycji inicjujących nurt *software studies*. Wardrip-Fruin analizował systemy gier i na podstawie ich działania badał kryjące się za nimi oprogramowanie, bez schodzenia na poziom analizy kodu. Jak wspominałam powyżej, kluczowym pojęciem dla jego metodologii było sformułowane już w tytule ekspresywne przetwarzanie<sup>101</sup> rozumiane jako podstawowa cecha oprogramowania. Zakłada ono, że sposób w jaki system cyfrowy funkcjonuje i na jakie interakcje z użytkownikiem pozwala, jest manifestacją oprogramowania, które nim zarządza.<sup>102</sup> Nie oznacza to, że badanie tego w

---

<sup>100</sup> Zob. <https://medialabkatowice.eu/en>, dostęp 25.11.2023.

<sup>101</sup> Tłumaczenie tego pojęcia jest moją własną propozycją. Książka nie została jeszcze przetłumaczona na język polski i nie mam możliwości odniesienia się do polskiego opracowania tej kategorii. Muszę jednak zaznaczyć, że angielskie „processing” odwołuje się przede wszystkim do procesów obliczeniowych i mój wybór polskiego słowa „przetwarzanie” także ma odnosić się do tego porządku.

<sup>102</sup> N. Wardrip-Fruin, dz. cyt., s. XI-XII.

jaki sposób działa dany program (lub w przypadku analiz Wardrip-Fruina – gra) pozwala domyślać się jak sformułowany został kryjący się za nim algorytm, jakich dokładnie funkcji użyto czy też jak wygląda struktura odpowiadającego mu kodu. Ekspresywne przetwarzanie należy rozumieć w taki sposób, że sama analiza logiki działania programu (Wardrip-Fruin określa ten poziom „logiką operacyjną”<sup>103</sup>) jest wystarczająca do przeprowadzenia krytyki i interpretacji zarządzającego nim oprogramowania w szerokim kontekście społecznym, politycznym i ideologicznym.

Postulat o ekspresywnym charakterze oprogramowania stawiany był również przez innych badaczy. Geoff Cox wraz z Alexem McLeanem w publikacji *Speaking Code. Coding as Aesthetic and Political Expression* skupili się na tym, aby przedstawić to, jak za pomocą oprogramowania można wyrażać treści ideologiczne, polityczne, społeczne czy artystyczne. Oprogramowanie może służyć jako środek artystycznego wyrazu, można za jego pomocą manifestować sprzeciw polityczny, proponować nowe modele ekonomiczne, jak chociażby w przypadku wolnego oprogramowania. Autorzy zaznaczają jednak, że zawarte w tytule „speaking code” pozwala się też rozumieć w odwrotny sposób, który Franco Berardi we wstępie do publikacji określa przez zwrot „kod wyraża się nami” (*code is speaking us*).<sup>104</sup> W ujęciu Coxa i McLeana ekspresywna funkcja oprogramowania ma dwie strony: z jednej z nich stanowi medium wyrazu treści i wartości, ale z drugiej jest też narzędziem kontroli, które ogranicza działania użytkownika do określonego pola interakcji, albo zamyka go w utrwalonych wewnątrz modelach działania systemu. Innymi słowy, za każdym razem, kiedy używa się oprogramowania, i co za tym idzie, korzysta się z jego ekspresywnej funkcji w celu wyrażenia jakichś treści, samo oprogramowanie również wyraża się w swoim działaniu, które zamyka użytkownika w określonym zakresie interakcji, działania systemu i modelu reprezentacji świata, który wpisany jest w określony software.

W ramach niniejszej dysertacji rozpoznania odnośnie ekspresywnego charakteru oprogramowania zostaną zastosowane w odniesieniu nie do gier, jak czynił to Wardrip-Fruin, ale do funkcjonujących powszechnie cyfrowych systemów, programów i aplikacji, z którymi użytkownicy na co dzień podejmują interakcję. Wszelkie przywoływane w pracy przykłady rozumiane są właśnie jako manifestacja działania oprogramowania.

---

<sup>103</sup> Tamże, s. XI.

<sup>104</sup> F. Berardi, *Foreword: Debt, Exactness, Excess*, w: G. Cox, A. McLean, dz. cyt., s. IX.

### 1.4.2 Wielopoziomowa struktura oprogramowania

Drugie założenie, kluczowe dla zrozumienia przedstawianej tu definicji oprogramowania, opiera się na rozpoznaniach dotyczących wielopoziomowego charakteru struktury software'u. Oprogramowanie dzieli się na oprogramowanie wysokiego poziomu (wyrażone w języku zrozumiałym dla człowieka) i niskiego poziomu (wyrażone w postaci instrukcji zrozumiałych dla komputera). Warstwowość oprogramowania wynika bezpośrednio z historii jego rozwoju. Adele Mildred Koss, programistka, która pracowała przy pierwszych komputerach i była świadkiem automatyzacji procesu programowania, opublikowała w postaci artykułu naukowego, reportaże dotyczące jej doświadczeń z lat 40. i 50. XX wieku. Koss zawarła w nich opis tego, jak powstawał niski i wysoki poziom działania oprogramowania.<sup>105</sup> Działanie komputera zarówno współcześnie, jak i u początku rozwoju tej technologii, opiera się na elektryczności i sterowaniu napięciem. W przypadku pierwszych komputerów, np. maszyn ENIAC (Il. 1), a więc w czasach poprzedzających rozwinięcie się oprogramowania w postaci semantycznego języka, wydawanie instrukcji komputerowi i kontrolowanie przepływu napięcia odbywało się poprzez manualne przepinanie kabli. Zrozumiały dla komputera komunikat wyrażony był w języku maszynowym (binarnym), który niesie informację o dwóch stanach: jeden (prąd ma przepływać) i zero (prąd nie ma płynąć). Choć taka instrukcja nie stanowi problemu dla komputera, jeśli chodzi o interpretację i egzekucję, to jednak dla człowieka formułowanie myśli w języku maszynowym jest trudne i czasochłonne, z tego powodu potrzebna była ewolucja przekazywania instrukcji do komputera. Koss opisuje tę potrzebę w następujących słowach: „Sam hardware, jakkolwiek genialny by nie był, nie mógł rozwiązać szerokiego spektrum problemów; konieczne było wynalezienie oprogramowania.”<sup>106</sup> Przygotowywanie instrukcji w języku maszynowym nie było też prostym przełożeniem badanego problemu na kod, ale wieloetapowym procesem najpierw samodzielnego rozwiązywania problemu za pomocą narzędzi nauk ścisłych, a potem rozbijanie go na mniejsze zadania, które dopiero w kolejnych etapach można było tłumaczyć na język komputerowy. Koss opisuje proces przygotowywania zadań dla pierwszych komputerów w następujący sposób:

---

<sup>105</sup> A.M. Koss, *Programming on the Univac I: A woman's account*. „IEEE Annals of the History of Computing” 2003, nr 25(1), s. 48-59.

<sup>106</sup> Tamże, s. 49.

Zanim komputer mógł rozwiązać problem, najpierw człowiek musiał sam to zrobić, używając matematyki, fizyki lub procesu biznesowego. Człowiek musiał podzielić rozwiązanie na małe bloki, z którymi maszyna mogłaby sobie poradzić, i użyć kodu maszynowego do napisania programu, który wykonałby ten proces poprawnie, z dużą prędkością i przez wiele iteracji bez interwencji człowieka. Problemy można było rozwiązywać na wiele różnych sposobów, ale opracowanie poprawnego i eleganckiego rozwiązania – podkreślam eleganckiego – było wyzwaniem.<sup>107</sup>

Jak wynika z zapisanych obserwacji programistki, pierwsze komputery były zdolne do rozwiązywania skomplikowanych zadań i problemów tylko jeśli wcześniej zostały one odpowiednio przygotowane przez człowieka. Ewolucja tej technologii opierała się więc na stopniowym automatyzowaniu tych zadań, które wykonywał człowiek w celu usamodzielnienia działania komputera. Koss przywołuje rozwój tego procesu:

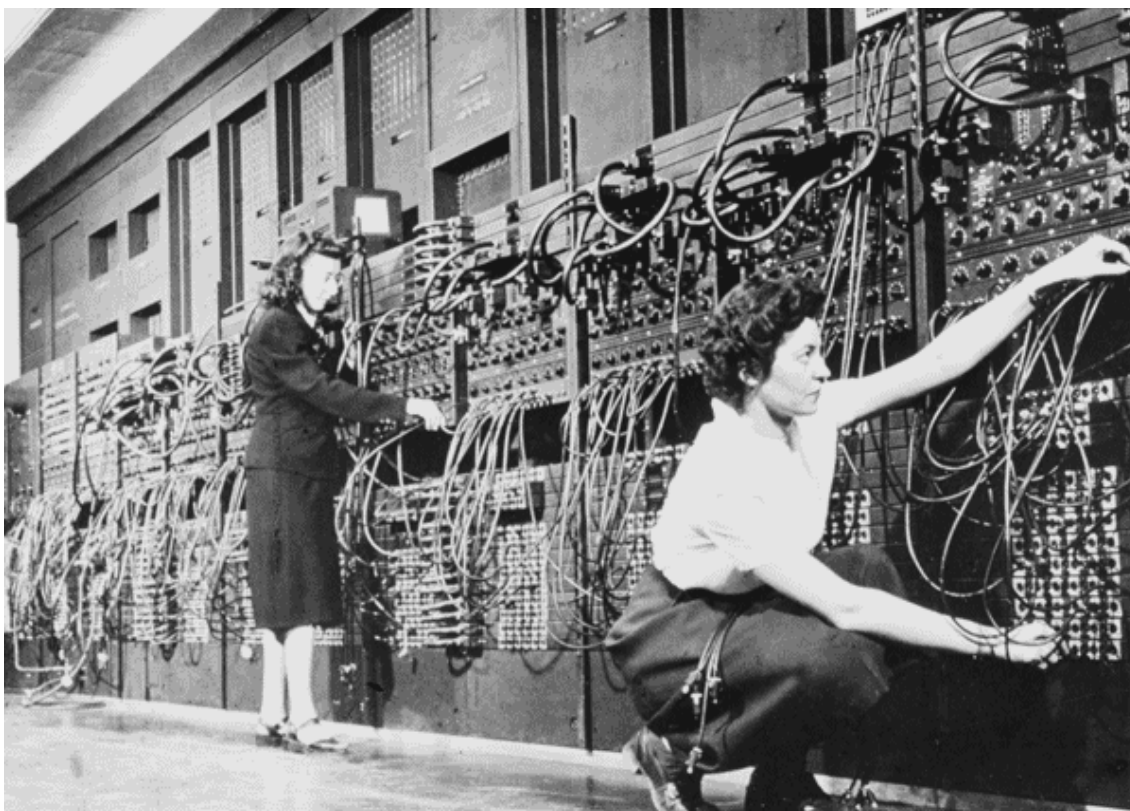
Pisanie kodu maszynowego wymaga kilku żmudnych kroków – rozbicia procesu na dyskretne instrukcje, przypisania określonych miejsc w pamięci do wszystkich poleceń i zarządzania buforami wejścia/wyjścia. Po wykonaniu tych kroków w celu zaimplementowania procedur matematycznych, biblioteki podprogramów i programów sortujących, naszym zadaniem było przyjrzenie się większemu procesowi programowania. Musieliśmy zrozumieć, jak możemy ponownie użyć przetestowanego kodu i jak maszyna może pomóc w programowaniu. Programując, badaliśmy ten proces i próbowaliśmy myśleć o sposobach abstrakcji tych kroków, by włączyć je do języka wyższego poziomu. Doprowadziło to do powstania interpreterów, assemblerów, kompilatorów i generatorów – programów zaprojektowanych do działania na lub tworzenia innych programów, czyli do programowania automatycznego.<sup>108</sup>

Języki wysokiego poziomu powstały po to, aby można było wyrażać instrukcje dedykowane maszynie komputerowej w języku przybliżonym do naturalnego, a więc przystępnym ludziom. Dzięki ich rozwojowi, za tłumaczenie tych instrukcji na języki niższego poziomu, a więc przygotowywanie zadań tak, aby były możliwe do interpretacji i egzekucji przez procesor komputera, przestał odpowiadać człowiek, a zaczęto korzystać w tym celu z narzędzi, o których wspomina Koss, takich jak assablery, kompilery i generatory.

---

<sup>107</sup> Tamże.

<sup>108</sup> Tamże, s. 56.



Il. 1. Komputer ENIAC, 1940. Źródło: Liz Huang, *Refrigerator Ladies: The First Computer Programmers*, online: <https://tanzu.vmware.com/content/blog/refrigerator-ladies-the-first-computer-programmers>, dostęp: 20.11.2023.

Wendy Hui Kyong Chun poruszała kwestie widzialności i percepcji oprogramowania w odniesieniu do jego wielopoziomowej struktury i podkreślała przy tym, że języki programowania wysokiego poziomu powstały w celu wyrażenia czegoś, co pozornie zdawało się niemożliwe do przełożenia, a więc translacji działania maszyny na język semantyczny.<sup>109</sup> Umożliwienie tego przełożenia znacznie przyspieszyło rozwój komputeryzacji, ale jednocześnie, jak podkreśla badaczka, sprawiło, że sam komputer i jego działanie stało się ukryte, zasłonięte, oddalone od percepcji użytkownika. W konsekwencji niższe warstwy oprogramowania stały się utajone. Chun opisuje ten proces w następujących słowach:

---

<sup>109</sup> W. Hui Kyong Chun, *On Software...*, dz. cyt. s. 44.

Języki programowania wyższego poziomu, w przeciwieństwie do języka maszynowego, wysyłają instrukcje i pozwalają zapomnieć o maszynie. Umożliwiają one uruchomienie programu na więcej niż jednej maszynie – właściwość, którą obecnie uważa się za „naturalną” cechę oprogramowania.<sup>110</sup>

Automatyzacja oprogramowania i jego usamodzielnienie się przyczyniło się do tego, że warstwy programowania niższych poziomów przestały się manifestować w sposób oczywisty. Właśnie z tego powodu Friedrich Kittler sformułował w latach 90. prowokacyjną tezę, głoszącą, że oprogramowanie nie istnieje („There is no software”)<sup>111</sup>, ponieważ przez redukcję jego działania warstwa po warstwie, daje się ono sprowadzić do różnicy napięć w przewodach. Badacz starał się w ten sposób podkreślić, że mimo iż oprogramowanie stało się abstrakcyjne i pozornie oddzieliło się od komputera, to nadal pozostaje funkcją hardware’u. W innym miejscu Kittler porównywał języki wysokiego poziomu do postmodernistycznej wieży Babel, ponieważ, jak twierdził, „po prostu nie wiemy, co nasze pisanie sprawia”.<sup>112</sup> Konsekwencją wprowadzenia języków wyższego i niższego poziomu jest rozdzielenie poziomu semantycznego i obliczeniowego w działaniu komputera. Podkreślał to także pionier informatyki, Edsger Dijkstra podczas jednego z wykładów:

podczas gdy maszyny muszą być w stanie wykonywać programy (bez ich zrozumienia), ludzie muszą być w stanie je zrozumieć (bez ich wykonywania). Te dwie czynności są tak całkowicie rozłączne – jedna może mieć miejsce bez drugiej.<sup>113</sup>

Podsumowując konsekwencje podziału na wysokie i niskie języki programowania, trzeba zaznaczyć, że software funkcjonuje zarówno w obszarze technologicznym i materialnym. Przez wzgląd na obliczeniowy charakter języków niskiego poziomu, które kontrolują przepływ elektryczności w komputerze, jak i jednocześnie, na poziomie języków wysokiego poziomu, staje się on problemem wyrażonym w języku programowania, a więc odnoszącym się do porządku semantycznego. Ten drugi poziom otwiera oprogramowanie na wszelkie konsekwencje funkcji języka i przez to sprawia, że funkcjonuje ono (podobnie jak język) jako problem społeczny oraz narzędzie kulturotwórcze. Tę zależność podkreślał już między innymi Ian Bogost w swojej koncepcji retoryki proceduralnej, przez którą wyraził, że procesy obliczeniowe mogą zawierać w sobie modele perswazji, struktury symboliczne

---

<sup>110</sup> Tamże, s. 30.

<sup>111</sup> F. Kittler, *There is No Software*, w: *Literature, Media, Information Systems: Essays*. red. J. Johnston, Overseas Publishers Association, Amsterdam 1997, s. 150.

<sup>112</sup> Tamże, s. 148.

<sup>113</sup> E.W. Dijkstra, *A Review of the 1977 Turing Award Lecture by John Backus*, 23.07.2006, online: <http://www.cs.utexas.edu/users/EWD/transcriptions/EWD06xx/EWD692.html>, dostęp 14.03.2023.

i interpretacje<sup>114</sup>. Tego rodzaju procedura jest więc jednocześnie obliczeniowa i ideologiczna.

Nie można powiedzieć, że granicą oprogramowania jest warstwa języków wysokiego poziomu. Z podobnego założenia wychodzili również przywoływani już przeze mnie badacze realizujący przestrzenno-geograficzne ujmowanie oprogramowania, tacy jak Bratton, Kitchin i Dodge. Na rzecz konstruowanej przeze mnie definicji również uznaję, że tradycyjny podział warstw oprogramowania nie jest wystarczający, ponieważ pomija się wówczas co najmniej dwa elementy: po pierwsze, wynikające z ekspresywnej funkcji oprogramowania działanie systemu, czyli efekt egzekucji oprogramowania, który będę określać mianem interfejsu oraz po drugie, wszelkie społeczne i kulturowe oddziaływanie oprogramowania, które zachodzi jako część procesu działania software'u. Z tego powodu w prezentowanej przeze mnie definicji tradycyjny podział struktury oprogramowania poszerzony zostanie o te dwa poziomy i będzie prezentował się w następujący sposób:

- języki niskiego poziomu
- języki wysokiego poziomu
- interfejs
- impakt społeczny i kulturotwórczy

Znaczenie niskiego i wysokiego poziomu wyjaśniłam już we wcześniej części tego podrozdziału, ale dwie dodane przeze mnie warstwy wymagają jeszcze dokładniejszego przedstawienia. Zaczynając od interfejsu, należy podkreślić, że rozumiem to pojęcie podobnie jak Alexander R. Galloway, który kładł nacisk na to, że „interfejs nie jest rzeczą, interfejs zawsze jest efektem. Zawsze jest to proces translacji”.<sup>115</sup> Badacz w ten sposób zaznaczał, że interfejsu nie należy rozumieć jako określonego obiektu wizualnego, ale raczej jest to ciągle ustanawiający się proces, którego wynikiem jest właśnie efekt interfejsu. Ważne jest tu, że składa się na niego wiele czynników, rozgrywających się na wielu poziomach technologicznych, włączając w to przywoływane już języki wysokiego i niskiego poziomu, ale także wszelkie pośredniczące elementy hardware'u, które ostatecznie wpływają na to, w jaki sposób manifestuje się działanie oprogramowania.

Nie jest łatwo wyznaczyć granice interfejsu i odróżnić go jako warstwę odrębną od pozostałych. Przede wszystkim chcę zaznaczyć, że interfejs rozumiem jako dające się

---

<sup>114</sup> I. Bogost, *Persuasive Games...*, dz. cyt., s. IX.

<sup>115</sup> A.R. Galloway, *The Interface Effect*, dz. cyt., s. 33.

postrzegać ludzkiej percepcji działanie oprogramowania, które jest w ciągłym procesie wytwarzania własnego statusu. Jednak przyjąwszy tę definicję, należy uznać, że języki wysokiego poziomu można rozumieć jako interfejs języków niższego poziomu. W tym przypadku samo oprogramowanie staje się również interfejsem, a programista jego użytkownikiem. Na zależność tę zwracał uwagę już Galloway, pisząc że:

należy podkreślić, że oprogramowanie nie może być rozumiane wyłącznie jako obiekt społeczny i technologiczny, ale jako problematyczny interfejs – taki, który jest w ciągłym procesie wytwarzania swojego społecznego i technologicznego statusu.<sup>116</sup>

Z tego powodu warstwa interfejsu posiada najmniej określone granice i w zależności od perspektywy może obejmować również obszar języków wysokiego poziomu. Poziom impaktu społecznego i kulturowego, a więc kolejna warstwa, o którą poszerzam definicję oprogramowania, należy rozumieć jako sposób, w jaki oprogramowanie oddziałuje na swoich użytkowników oraz jak przyczynia się do kształtowania rzeczywistości.

Trzeba tutaj zaznaczyć, że dodane przeze mnie poziomy, a więc zarówno proces wytwarzania się interfejsów, jak i społeczne oraz kulturowe oddziaływanie oprogramowania, są bezpośrednio związane z pierwszą cechą software'u, którą przywoływałam, a więc z jego ekspresywną funkcją. Oprogramowanie wyraża się w działaniu systemu, a impakt społeczny i kulturowy są dalszymi konsekwencjami tego procesu. Istotną cechą warstwowej struktury oprogramowania jest fakt, że wyższe jego warstwy starają się ukryć kryjące się pod nimi poziomy.

Kluczowym założeniem dla przeprowadzanych przeze mnie analiz jest teza dotycząca tego, że elementy wyrażone na różnych poziomach oprogramowania są sobie jakościowo równoważne. Koncepcja ta była prezentowana już przez Kittlera w jego sformułowaniu, że oprogramowanie wysokiego poziomu daje się sprowadzić do różnicy napięć w przewodach, a następnie odwoływał się do niej Galloway, który pisał, że:

Nieskompilowany kod źródłowy jest logicznie równoważny temu samemu kodowi skompilowanemu do języka assemblera i/lub połączony z kodem maszynowym. Na przykład, absurdalne jest twierdzenie, że pewna wartość wyrażona jako liczba szesnastkowa (podstawa 16) jest bardziej lub mniej fundamentalna niż ta sama wartość wyrażona jako liczba binarna (podstawa 2). Są to po prostu dwa wyrażenia tej samej wartości.<sup>117</sup>

---

<sup>116</sup> Tamże, s. 53.

<sup>117</sup> A.R. Galloway, *Protocol: how control exists after decentralization*, MIT Press Cambridge, Massachusetts, 2004, s. 167.

Należy tu podkreślić, że zarówno Kittler, jak i Galloway odnosili się do relacji między wartościami języków wysokiego i niskiego poziomu, a więc ich porównanie pozostawało w obszarze technicznym. W ramach wypracowywanej przeze mnie definicji przyjmuję założenie, że tę właściwość można przełożyć również na kolejne, dodane przeze mnie warstwy oprogramowania. Tym samym, wartości wyrażone na poziomie interfejsu lub społecznego i kulturowego impaktu są jakościowo równoważne odpowiadającym im wartościom niższych poziomów z porządku technologicznego. To podejście uzasadnia, dlaczego zawarte w niniejszej dysertacji rozważania są zorientowane wokół oprogramowania i można je wpisać w dyskurs nauk *software studies*, mimo że nie przywołuję w pracy przykładów oprogramowania wyrażonych w językach wysokiego poziomu. Ważne, żeby podkreślić tutaj, że zdaję sobie sprawę, z tego, że Kittler i Galloway odwoływali się w swoich założeniach do wartości wyrażonych na poziomie technologicznym, natomiast ja proponuję jakościowe zrównanie sobie elementów wyrażonych na poziomie technologicznym z elementami wyrażonymi na poziomie wizualnym, społecznym i kulturowym. Podążając za oboma badaczami, jednocześnie rozumiem, że oprogramowanie manifestuje się na każdym z poziomów w inny sposób pod względem ontologicznym, jednak wychodzę z założenia, że krytyczna analiza w obrębie dodanych przeze mnie poziomów nie zamyka się jedynie do ich obszaru, ale odwołuje się również do ukrytych pod nimi poziomów z porządku technologicznego.

Tak wyprowadzone założenie pozwala dotrzeć do punktu wspólnego pierwszego i drugiego aspektu skonstruowanej przeze mnie definicji *software'u*. Łączą się one w tym, że analiza działania systemu jest krytyką oprogramowania, które za nim stoi. W przypadku pierwszego aspektu wynika to z ekspresywnej funkcji oprogramowania, natomiast w przypadku drugiego z tezy na temat jakościowej równości odpowiadających sobie elementów różnych poziomów oprogramowania. We współczesnej kulturze cyfrowej użytkownik często nie ma wglądu w oprogramowanie, z którego interfejsami wchodzi w interakcję, ponieważ stanowi ono własność intelektualną firm, które je posiadają. Z tego powodu prezentowane przeze mnie podejście, które skupia się na działaniu systemu często okazuje się jedyną możliwością przeprowadzenia jakiegokolwiek krytyki oprogramowania.

### 1.4.3 Oprogramowanie jako system operacyjny kultury

Wielu badaczy kulturowego znaczenia oprogramowania stawiało postulaty dotyczące potrzeby przywoływania historycznego kontekstu rozwoju software'u. Opierała się na nich Chun, która na podstawie historii rozwoju języków niskiego i wysokiego poziomu analizowała problematykę z jednej strony ekspresywnej funkcji oprogramowania, a z drugiej strony jego przeciwnej tendencji do ukrywania znaczeń ideologicznych i kryjących się pod nim poziomów. O potrzebie przywoływania historii oprogramowania pisał także Manovich najpierw w *Języku nowych mediów*, a później w *Software Takes Command*. W pierwszej publikacji badacz udawał, że nowe media są konsekwencją przecięcia się dwóch, rozwijających się niezależnie procesów kulturowych – metod obliczeniowych i technik utrwalania obrazu. Dwuaspektowy początek procesu zapoczątkowany został z jednej strony przez maszynę obliczeniową Charlesa Babbage'a, z drugiej zaś przez dagerotyp Louisa Daguerre'a. Rozwój historyczny tych dwóch wydarzeń doprowadził do ich przecięcia się momencie powstania interfejsu cyfrowego<sup>118</sup>. W późniejszej publikacji badacz podkreślał problematykę ahistoryczności mediów cyfrowych oraz samego software'u w potocznym rozumieniu. Pisał, że historia perspektywy malarskiej istnieje w świadomości społecznej, czego nie można jednak powiedzieć o historii oprogramowania, np. aplikacji Photoshop.<sup>119</sup> Jak zauważył Manovich, już sam fakt, że komputery osobiste po prostu pojawiły się w latach dziewięćdziesiątych w domach użytkowników, wpływa na to, że postrzegane są jak obiekty bez przeszłości. W pewnym momencie stały się one dostępne w sklepach i zostały przyniesione do domu.

Na problem ahistoryczności oprogramowania i jednocześnie potrzebę wpisania je w przekształcające się procesy kulturowe zwracał uwagę również Geoff Cox, który w 2007 roku, w ramach konferencji *re:place*, w Berlinie wystąpił z referatem pod tytułem *Software Art has No History*. Fraza zawarta w tytule wystąpienia została zapożyczona od Johna Robertsa i jego książki wydanej w 1994 roku: *Art Has No History! The Making and Unmasking of Modern Art*. Ten z kolei zapożyczył ją od Louisa Althussera, autora tezy „Ideology has no History”<sup>120</sup>. Ostatni z badaczy podkreślał, że ideologia ma postać niezmienną struktury, która trwa od lat, a zmianie ulega tylko treść, którą zostanie wypełniona. Inny badacz, Fredric Jameson zaznaczał, w tym względzie, że już samo użycie

---

<sup>118</sup> L. Manovich, *Język nowych mediów*, dz. cyt., s. 82.

<sup>119</sup> L. Manovich, *Software Takes Command*, dz. cyt., s. 4.

<sup>120</sup> L. Althusser, *Ideological State Apparatuses*, w: *Mapping Ideology*, red. S. Žižek, Verso, Londyn, Nowy Jork 1994, s. 120-123.

słowa „ideologia”, wyraża krytyczny stosunek mówiącego do niej, a więc ujawnia ją i odsłania, co też podkreślał Galloway w artykule *Język chce by go nie dostrzegać*.<sup>121</sup> Przywoływanie historycznych aspektów związanych z ideologią dostarcza narzędzi do jej opisu, a więc pozwala usytuować ją w określonym kontekście, i tym samym zanegować jej przezroczystość. Podobny cel chcieli osiągnąć badacze software’u i poprzez przywoływanie jego historyczności starali się uczynić namacalnymi te poziomy technologiczne, które oprogramowanie oddala, i sprawia, że pozostają zapomnianymi.

Istotną publikacją, która skupia się na przywoływaniu historycznego kontekstu rozwoju mediów, jest *Archeologia Mediów* autorstwa Siegfrieda Zielińskiego. Książka stanowi alternatywną propozycję opowiadania o historii mediów i przedstawia głębokie, często wybiegające daleko w przeszłość. Nie jest to dosłowne przedstawienie ciągłości rozwoju wynalazków technologicznych, ale raczej prezentacja marginalnych, nierzadko osobiwych procesów, pomysłów i zjawisk, które umożliwiły rozwój poszczególnych mediów. Badacz oparł swoje podejście na założeniach Michela Foucaulta, autora słynnej koncepcji archeologii wiedzy. Dla Foucaulta odniesienia historyczne również były kluczowym elementem analizy współczesnych idei, nauk filozoficznych, myśli, czy po prostu dyskursów. Nie rozumiał on pojęcia historii w sposób tradycyjny, ciągły, ale skupiał się na nieciągłej, rozproszonej w czasie sumie faktów danego dyskursu.<sup>122</sup> Podobne podejście zostało zawarte w tej pracy doktorskiej, jednak bardziej niż na przywoływaniu historii technologii zależało mi na przedstawieniu relacji historycznej pewnych procesów społeczno-kulturowych, które są związane z technologią.

W niniejszej pracy proponuję, aby myśleć o oprogramowaniu jako o systemie operacyjnym kultury. W myśl tej idei software jest ideologiczną machiną, która rejestruje zjawiska kulturowe cyfrowej rzeczywistości, przetwarza je i poddaje ciągłej aktualizacji. Historyczne proveniencje zjawisk kulturowych mają w tym przypadku dla mnie szczególne znaczenie, ponieważ wychodzę z założenia, że oprogramowanie aktualizuje zastane wartości kulturowe, które rozwijały się przed epoką cyfrową i nadaje im nową formę w ramach współczesności.

Z racji tego, że w niniejszej dysertacji skupiam się na analizie interfejsów i impaktu społeczno-kulturowego jako wyższych warstw oprogramowania, w ramach każdego rozdziału przywołuję najpierw kontekstową część wprowadzającą, która ma na celu

---

<sup>121</sup> A.R. Galloway, *Język chce by...*, dz. cyt., s. 24.

<sup>122</sup> M. Foucault, *O archeologii nauk. W odpowiedzi Kołu epistemologicznemu*, tłum. A. Bandura, w: *Język, dyskurs, społeczeństwo*, red. L. Rasiński, Wydawnictwo Naukowe PWN SA, Warszawa 2009, s. 313.

wskazanie, że procesy kulturowe, do których się odwołuję, i które obecnie zapośredniczone są przez oprogramowanie, nie są jednak wyłącznie jego wytworem, ale rozwijały się w pewnych swoich aspektach już przed epoką cyfrową. Celem odwoływania się do historycznych kontekstów, poprzedzających epokę cyfrową, jest pokazanie, że pewne zjawiska kojarzone współcześnie bezpośrednio z oprogramowaniem, nie muszą wynikać z cyfrowego charakteru współczesnych mediów, ale były obecne już wcześniej i mają charakter społeczny. Nie muszą być one zdeterminowane technologicznie, lecz społecznie, a technologia pełni tu rolę mediatora, katalizatora społecznych zjawisk i napięć. Manovich postawił prowokacyjne pytanie o to: „Czy ma sens teoretyzowanie o współczesności, skoro zmienia się ona tak szybko?”<sup>123</sup>. Praca ta zakłada, że jest to możliwe, a ciągłość procesów społeczno-kulturowych dostarcza pomocnych tropów interpretacyjnych.

---

<sup>123</sup> L. Manovich, *Język nowych mediów*, dz.cyt., s. 64.

## **I. Kontekst historyczny dla rozdziału drugiego i trzeciego: automaty**

XVIII wiek w Europie to czas zintensyfikowanych poszukiwań nowych rozwiązań technologicznych, przede wszystkim w dziedzinie mechanicznej automatyzacji. Właśnie wtedy ogromną popularność zyskały eksperymenty nad stworzeniem sztucznej, mechanicznej imitacji życia. Powszechnie powstawały wówczas automaty, które w sposób mimetyczny naśladowały ludzi lub zwierzęta. Były to wynalazki wykonujące rozmaite czynności: od najprostszych, takich jak poruszanie się, przez bardziej skomplikowane, jak pisanie listów<sup>125</sup>, rysowanie,<sup>126</sup> czy gra na instrumentach<sup>127</sup>, aż po prawdziwe zagadki technologiczne rekonstruujące procesy fizjologiczne czy intelektualne. Mechanizacja i postęp w technologiach automatyzacji były tym, co w dalszej perspektywie bezpośrednio przyczyniło się do rewolucji przemysłowej mającej miejsce na przełomie XVIII i XIX wieku, o której będę pisać we wprowadzeniu do rozdziału piątego.

Zjawiska technologiczno-społeczne z XVIII wieku, które poniżej przywołam wynikają z pewnej tendencji rozumienia świata i sposobu prowadzenia badań na temat rzeczywistości, które realizują założenia nurtu filozoficznego zwanego mechanicyzmem. Ufundowany został on na przekonaniu, że wszelkie procesy życiowe, włączając w to świadomość oraz życie, można wyjaśnić za pomocą zasad mechaniki.<sup>128</sup> Ta koncepcja filozoficzna rozwijana była już od wieku XVI, a do jej rozwoju przyczyniły się odkrycia naukowe, związane z formułowaniem zasad mechaniki klasycznej. W XVI wieku jednym z pionierów mechaniki klasycznej i pierwszymi filozofami mechanicyzmu byli fizycy: Galileusz, Sebastien Basson, matematyk Thomas Harriot i filozofowie: Nicholas Hill i David van Goorle.<sup>129</sup> Należy jednak podkreślić, że w XVI wieku nie uformowała się żadna

---

<sup>124</sup> T. Standage, *The Turk. The Life and Times of The Famous Eighteenth-Century Chess-Playing Machine*, Walker & Company, Nowy Jork 2002, s. 2. Tłumaczenie A.P.

<sup>125</sup> Zob. Automat „Pisarz” z 1772 roku, autorstwa zegarmistrza Henri-Louisa Jaqueta Droza. Mechanizm przypominał wyglądem chłopca, potrafił moczyć pióro w kałamarzu i pisać krótkie fragmenty tekstu,

<sup>126</sup> Zob. Automat „Rysownik” Droza, który potrafił tworzyć rysunki, sportretował on Ludwika XV, Ludwika XVI i królową Marię Antoninę. W trakcie pracy zdmuchiwał z arkusza pozostałości pyłu ołówkowego.

<sup>127</sup> Zob. Automat „Muzyk” autorstwa Droza lub „Fleciście” i „Tamburyniście” Vaucansona.

<sup>128</sup> G.E. Allen, *Mechanism, Vitalism and Organicism in Late Nineteenth and Twentieth Century Biology. The Importance of Historical Context*, „Studies in History and Philosophy of Biological and Biomedical Sciences” 2005, tom 36, nr 2, s. 261-283.

<sup>129</sup> M.J. Osler, *Mechanical Philosophy*, w: *New Dictionary of Ideas*, red. M. C. Horowitz, Charles Scribner's Sons, Nowy Jork 2005, s. 1389-1390. Mechanicyzm był dominującym sposobem myślenia

spójna formuła idei mechanicyzmu, a wymienieni naukowcy przedstawiali różne, czasem wykluczające się wzajemnie punkty widzenia.<sup>130</sup> Pierwsza usystematyzowana koncepcja mechanicyzmu została wyrażona w XVII wieku w dziełach Kartezjusza. W *Medytacjach o pierwszej filozofii* naukowiec prowadził refleksje ontologiczne dotyczące ciała, duszy i umysłu. Francuski uczony opisywał ciało jako maszynę, składającą się z członków i organów, która posiada możliwość wprawiania się w ruch i wyposażoną w zmysły, które pozwalają poznawać otoczenie i inne ciała.<sup>131</sup> Kartezjusz sformułował też tezę głoszącą, że ludzie nie tylko są zdolni tworzyć maszyny, ale sami są niczym samonapędzające się automaty. Według francuskiego uczonego, wszystkie procesy życiowe zachodzące w ludzkim ciele podporządkowane są fizycznym prawom mechaniki z wyjątkiem myślącej duszy, którą uczony umiejscowił w szyszynce.<sup>132</sup>

Pełną postać filozofia mechanistyczna uzyskała pod koniec XVII wieku, przede wszystkim dzięki odkryciom Isaaca Newtona w dziedzinie fizyki, które przyczyniły się do powszechnego wykorzystywania pojęć z języka mechaniki, takich jak materia, ruch czy siła do opisu rzeczywistości.<sup>133</sup> W XVIII wieku filozoficzne teorie mechanicystyczne potrafiły przybierać radykalną formułę, czego przykład stanowi tu książka *Człowiek-Maszyna*, napisana przez lekarza, Juliana Offray de La Mettrie w 1748 roku. Autor publikacji postawił znak równości pomiędzy człowiekiem a maszyną, przyjmując założenie, że każdy proces zachodzący w ciele i psychice (duszy) człowieka ma charakter mechaniczny, to znaczy musi mieć swoją materialną przyczynę. Za opublikowanie tak radykalnych teorii jej wydawca został ukarany grzywną i wygnany z miasta.<sup>134</sup>

W późnej filozofii mechanicystycznej XVIII wieku organizmy żywe były uznawane za zaawansowane mechanizmy.<sup>135</sup> Samo określenie *automaton* etymologicznie pochodzi z języka greckiego (αὐτόματον) i oznacza „poruszający się samodzielnie”, „obdarzony wolą”. Do XVIII wieku zakres znaczeniowy tego terminu obejmował również człowieka,

---

i rozumienia świata od XVI do XIX wieku, aż do momentu rozwoju innych teorii fizycznych, takich jak teoria względności i mechanika kwantowa, które sprawiły, że klasyczna mechanika stała się niewystarczającym narzędziem opisu rzeczywistości. Na przykładzie rozwoju tego nurtu filozoficznego widać jak odkrycia z dziedziny fizyki wpływają na rozwój myśli filozoficznej danego okresu.

<sup>130</sup> Tamże.

<sup>131</sup> R. Descartes, *Medytacje o pierwszej filozofii*, tłum. M. i K. Ajdukiewiczowie, S. Swieżawski, I. Dąbska, Wydawnictwo ANTYK, Kęty 2001, s. 49-50.

<sup>132</sup> Zob. C.U. Smith, *Descartes' Pineal Neuropsychology*, „Brain and Cognition” 1998, tom 36, nr 1, s. 57-72.

<sup>133</sup> M.J. Osler, dz. cyt., s. 1391.

<sup>134</sup> Zob. J. O. de La Mettrie, *Człowiek – maszyna*, tłum. S. Rudniański, Warszawa 1984, s. XXX-XXXII.

<sup>135</sup> Tamże.

ponieważ termin ten określał podmiot zdolny do podejmowania samodzielnych działań.<sup>136</sup> Jednak rozpowszechnianie się samodzielnych mechanizmów spowodowało potrzebę reinterpretacji tego określenia. Spośród wielu mimetycznych mechanizmów powstających od początku XVIII wieku dwa w sposób szczególny przeszły do historii, stały się sensacją swoich czasów oraz symbolem eksperymentów nad stworzeniem imitacji sztucznego życia.<sup>137</sup> Pierwszym z nich jest „Kaczka” Jacquesa de Vaucansona z 1739 roku, natomiast drugim „Turek” Wolfganga von Kempelena z 1770 roku. Oba mechanizmy oprócz tego, że były przedsięwzięciami komercyjnymi, mającymi na celu rozrywkę i demonstrację mechanicznej pomysłowości, nieintencjonalnie podjęły próbę udratyzowania problemów filozoficznych i społecznych tamtego okresu.

„Kaczka”, nazywana również „Trawiącą Kaczką” (Il. 2), prezentowana była zazwyczaj pomiędzy dwoma innymi automatami Vaucansona – mechanicznymi „Flecistą” i „Tamburynistą”. Te trzy wynalazki były efektem starań autora, by zrealizować marzenie o stworzeniu sztucznego człowieka. Ich twórca poświęcił się odtwarzaniu poszczególnych funkcji życiowych, takich jak oddychanie, wydalenie czy krążenie krwi. Jego pierwszy automat, „Flecista”, posiadał system cyrkulacji powietrza, dzięki któremu mechanizm mógł odgrywać melodie na instrumencie.<sup>138</sup> Mechaniczna „Kaczka” powstała jako zwieńczenie obserwacji tego gatunku ptaków w naturalnym środowisku. Przedstawiała naturalnej wielkości kaczkę, która potrafiła poruszać się, pić wodę i jeść ziarna, a także, co było najbardziej widowiskowe, wydalać pożywienie. Podczas ekspozycji mechanizm przedstawiano w taki sposób, jakby wewnątrz zachodził proces trawienia. „Trawiąca Kaczka” skłaniała do filozoficznej refleksji na temat tego, czy procesy życiowe ludzi i zwierząt mogą zostać sprowadzone do postaci mechanizmu.<sup>139</sup> Sam Vaucanson określał „Kaczkę” mianem „małego laboratorium”<sup>140</sup>, a więc również nie traktował swojego wynalazku jako eksperymentu czysto mechanicznego, lecz próbę powtórzenia procesów fizjologicznych w warunkach laboratoryjnych.<sup>141</sup>

---

<sup>136</sup> E. Huhtamo, *Z ust lalki, niejasno. O sztuce Kena Feingolda*, w: *Ken Feingold: Figury mowy*, red. R.W. Kluszczyński, Centrum Sztuki Współczesnej Łaźnia, Gdańsk 2014, s. 46-47.

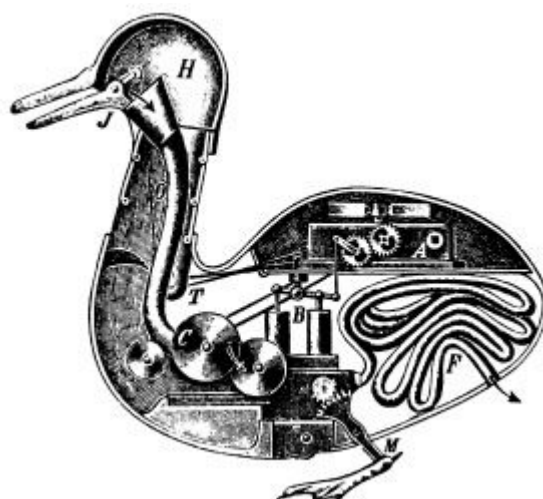
<sup>137</sup> Mam tu na myśli całokształt starań, które miały na celu zrozumienie procesów typowych dla organizmów żywych za pomocą mechanizmów. Nie chcę tu odwoływać się do koncepcji Artificial Life, pisanej wielkimi literami, która reprezentuje rozwijający się od połowy dwudziestego wieku obszar badań, którego celem była symulacja procesów życiowych za pomocą komputerowych symulacji, robotów oraz biochemii (tym trzem narzędziom odpowiadał kolejno termin: software, hardware i wetware).

<sup>138</sup> T. Standage, dz. cyt., s. 7.

<sup>139</sup> J. Riskin, *The Defecating Duck, or the Ambiguous Origins of Artificial Life*, „Critical Inquiry” 2003, tom 29, nr 4, s. 601.

<sup>140</sup> J. Riskin, *Eighteenth-Century Wetware*, „Representations” 2003, tom 83, nr 1, s. 103.

<sup>141</sup> Jessica Riskin proponuje, aby myśleć o „Kaczce” jako o przykładzie osiemnastowiecznego wetware’u. Zob. J. Riskin, *Eighteenth-Century Wetware*, dz. cyt., 103-104. Wetware to termin za pomocą którego określa



Il. 2. „Kaczka” Vaucansona, przekrój. Źródło: *Canard Digérateur de Vaucanson*, online: <https://www.atlasobscura.com/places/canard-digerateur-de-vaucanson-vaucansons-digesting-duck>, dostęp: 20.11.2023.

Wynalazek Vaucansona okazał się jednak oszustwem. W rzeczywistości nie był wcale małym laboratorium, ale sprytną symulacją. W 1783 roku jeden z obserwatorów znalazł się na tyle blisko mechanizmu, że udało mu się odkryć, że zjadane przez kaczkę ziarna zbierane były w jednym zbiorniku, a z innego wypuszczana była przygotowana wcześniej substancja przypominająca ekskrementy.<sup>142</sup> Można powiedzieć, że mechaniczna „Kaczka” była eksperymentem nie tylko technologicznym, ale również społecznym. Właściwy mechanizm jej funkcjonowania polegał na przedstawieniu odbiorcom „narracji”, która wokół niej powstała i operowaniu na wyobrażeniach dotyczących sposobu jej działania. Siła jej oddziaływania oraz poziom sensacji, jaką wywołała, były zależne od tego, jak obserwujący wyobrażali sobie, że wynalazek działa, a nie od właściwego kunsztu technicznego. Widzowie pokazu wierzyli, że są świadkami powstania mechanizmu zdolnego do przeprowadzania procesów przemiany materii, typowych dla organizmów żywych, choć w rzeczywistości nie posiadali wiedzy na temat konstrukcji i działania wynalazku. Jest to doskonały przykład tego, w jaki sposób technologia może funkcjonować jednocześnie na dwóch płaszczyznach – technologicznej i społecznej.<sup>143</sup>

---

się organizmy żywe w odniesieniu do terminologii z nauk informatycznych zob. M. Krawczak, *Wetware*, w: *Mały słownik sztuki technologicznej*, 25.08.2020 online: <https://culture.pl/pl/artukul/maly-sloownik-sztuki-technologicznej>, dostęp: 06.11.2023.

<sup>142</sup> J. Riskin, *The Defecating Duck...*, dz. cyt., s. 609.

<sup>143</sup> Warto tutaj dodać, że pomimo że Vaucansonowi nie udało się stworzyć prawdziwie trawiącej kaczki, jego pomysł został zrealizowany współcześnie przez Wima Delvoye’a, belgijskiego artystę konceptualnego. Stworzył on serię instalacji trawiących pod nazwą *Cloaca*. Ogromnych rozmiarów instalacja była zespołem maszynierii umożliwiającej z jednej strony wprowadzenie jedzenia, w centralnej części odbywał się proces

Innym przykładem, który również wyraziście obrazuje przenikanie się technologicznych i społecznych aspektów nowo wprowadzanych innowacji jest powstały później, bo w 1769 roku, automat, znany jako mechaniczny „Turek” (Il. 3). Był to wynalazek stworzony w celu rozgrywania partii szachowych. Wyglądem przypominał dużą skrzynię, do której przytwierdzony był drewniany korpus manekina – tytułowego Turka. Jego twórca, Kempelen nigdy nie nadał mechanizmowi nazwy, co jest bardzo znaczące, ponieważ wynalazca w ogóle nie był zainteresowany tym, aby automat zyskiwał popularność lub by był kojarzony właśnie jako jego dzieło. Pseudonim „Turek” powszechnie przyjął się jako nazwa dla szachowego mechanizmu przez wzgląd na orientalny strój, w jaki odziany był manekin.<sup>144</sup> Wynalazek został przygotowany na specjalny pokaz na dworze cesarzowej Austro-Węgier, Marii Teresy w Wiedniu. Przed rozpoczęciem partii ochotnik mający zmierzyć się z automatem był instruowany, by nie oszukiwać, by kłaść pionek dokładnie na środku pola oraz informowany o tym, że „Turek” zawsze wykonuje pierwszy ruch w rozgrywce. Zarówno premierowa prezentacja dzieła Kempelena, jak i każda kolejna, rozpoczynały się od tego samego rytuału. Wynalazca prezentował wnętrze mechanizmu. Otwierał i zamykał po kolei każde drzwiczki, stawał za otwartą skrzynią ze świeczką tak, że było widać na prześwit jej wnętrze, pokazywał konstrukcję kół zębatach, wysuwał maksymalnie dolną szufladę, by ostatecznie zamknąć wszystko, położyć na blat dwie świece oświetlające planszę, nakręcić automat i dalej już tylko obserwować z niedalekiego dystansu przebieg rozgrywki. Historyk Tom Standage opisywał moment tuż po prezentacji wnętrza mechanizmu, a jeszcze przed rozpoczęciem jego działania w następujący sposób:

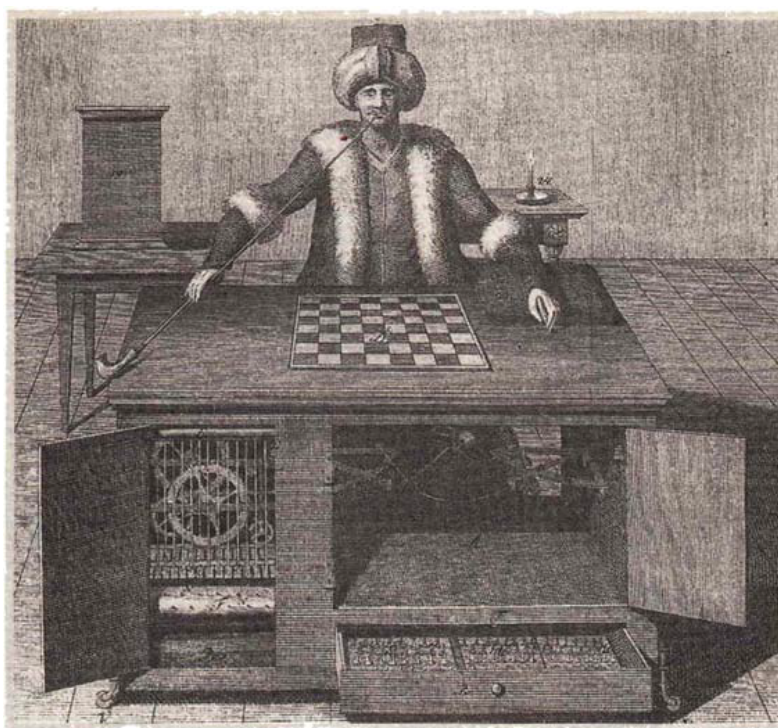
Gdy Kempelen przestał przekręcać klucz, zapadła przejmująca cisza. Potem, po krótkiej przerwie, z wnętrza automatu dało się słyszeć warkot i zgrzyt mechanizmu, podobny do tego, który wydaje zegar przygotowujący się do wybijania godziny. Rzeźbiona drewniana postać, która do tej pory była całkowicie nieruchoma, przez kilka sekund powoli obracała głowę z boku na bok, jakby przyglądając się planszy. Ku zdumieniu publiczności mechaniczny Turek nagle ożywił się, wyciągnął lewą rękę i przesunął jeden z pionów do przodu. Publiczność krzyknęła z zachwytem. Gra rozpoczęła się.<sup>145</sup>

trawienny, którego efektem było wypuszczenie z drugiej strony ekskrementów. Tak wyprodukowany kał był pakowany próżniowo i sprzedawany widzom. Pierwszy pokaz *Cloaca* odbył się w 2000 roku, w Muzeum Sztuki Współczesnej w Antwerpii (MuHKA). W pokazie udział wzięli profesjonalni kucharze, którzy dwa razy dziennie podawali maszynie przygotowane przez siebie dania. Trawienie wewnątrz instalacji było możliwe dzięki działaniu kwasów, enzymów i bakterii trawiennych. Aby opracować ten proces, Delvoye konsultował się z naukowcami.

<sup>144</sup> T. Standage, dz. cyt., s. XIV.

<sup>145</sup> Tamże, s. 28. Tłumaczenie A.P.

Mechaniczny „Turek” potrafił poruszać głową, a także ręką, którą przemieszczał piony szachowe. W trakcie rozgrywki jego wynalazca kilkakrotnie podchodził do urządzenia, aby je jeszcze raz nakręcić. Automat szachowy wygrał swoją pierwszą publiczną partię, a później, za życia Kempelena nie przegrał żadnej kolejnej. Zwycięstwo automatu nad człowiekiem wywołało ogromne poruszenie. Wielu świadków tego wydarzenia nie chciało przyjąć do wiadomości, że automat mógł być zdolny do pokonania człowieka w starciu intelektualnym.<sup>146</sup> Obserwatorzy rozgrywki podzielili się na dwie grupy: entuzjastów wynalazku, wierzących, że procesy obliczeniowe mogą zostać zautomatyzowane oraz sceptyków doszukujących się oszustwa, takiego jak ukrycia dziecka czy karła wewnątrz skrzyni lub też sterowania pionami szachowymi na odległość przez wynalazcę za pomocą magnesów.



Il. 3. „Turek” Kempelena z otwartymi drzwiczkami. Źródło: Tom Standage, *The Turk. The Life and Times of The Famous Eighteenth-Century Chess-Playing Machine*, Walker & Company, Nowy Jork 2002, s. 85.

Mimo tego że znaczna część odbiorców miała wątpliwości co do rzeczywistego działania mechanizmu, za życia Kempelena nikomu nie udało się udowodnić, że jego dzieło jest mistyfikacją. Teoria mówiąca o człowieku ukrytym w środku mechanizmu była trudna do

---

<sup>146</sup> Tamże, s. 33.

potwierdzenia przez to, że wewnątrz automatu było prezentowane przed każdą rozgrywką. Domysły dotyczące sterowania za pomocą magnesów również zostały pozbawione podstaw, kiedy autor mechanizmu zgodził się na przeprowadzenie partii szachowej z ogromnym magnesem położonym na blacie skrzyni. Zagadka związana z działaniem mechanicznego „Turka” sprawiła, że stał się on niezwykle popularny. Wielu ochotników przybywało, aby samodzielnie zmierzyć się z automatem i odkryć jego tajemnicę. Znaleźli się również tacy, którzy odważyli się złamać zasady rozgrywki i oszukiwać, żeby sprawdzić reakcję automatu. Po wykonaniu przez gracza nieprawidłowego ruchu „Turek” poruszał przecząco głową, a następnie cofał pion przeciwnika na początkowe miejsce. W dalszej kolejności wykonywał swój ruch, w efekcie czego tura gracza, który oszukiwał, przepadała.<sup>147</sup>

Zaraz po pierwszych pokazach „Turka”, Kempelen próbował porzucić ten projekt, odsunąć go w niepamięć i zaczął pracować nad nowym automatem, inspirowanym projektami Vaucansona dotyczącymi ludzkiego systemu oddechowego. Kempelen chciał rozwinąć te projekty i opracować maszynę mówiącą. Nowe pomysły wynalazcy nie spotkały się jednak z entuzjastycznym odbiorem, ponieważ powszechnie, mimo przeciwdziałań wynalazcy, domagano się pokazów „Turka”. Zagadkowy pozostaje również fakt, że Kempelen, wielokrotnie niszczył mechanizm, aby sabotować kolejne pokazy. Ostatecznie jednak znów naprawiał go, gdy okazywało się, że tylko ten, a nie żaden inny z prezentowanych przez niego wynalazków wzbudza zainteresowanie ówczesnej publiki. Ciekawe jest również to, że wynalazca nigdy nie zgodził się sprzedać swojego dzieła, za żadną cenę. Oryginał mechanizmu nigdy nie został ujawniony, jednak jego kopie, które tworzone na podstawie notatek uczonego, funkcjonowały jeszcze po śmierci autora. Sekret mechanizmu został ujawniony dopiero w drugiej połowie XIX wieku przez syna Silasa Mitchella, ostatniego prywatnego właściciela kopii mechanizmu. Mechaniczny „Turek” okazał się być nie automatem, ale marionetką, kontrolowaną przez ukrytego wewnątrz człowieka (Il. 4). Osoba ta nie musiała być karłem ani dzieckiem, skrzynia była w stanie ukryć dorosłego mężczyznę. Ukryty wewnątrz skrzyni człowiek mógł oglądać ruch pionów na planszy poprzez system magnesów umieszczonych w figurach, które przyciągały widoczne tylko od spodu blatu znaczniki. W środku mechanizmu mieściła się również druga szachownica, którą ukryty człowiek podświetlał za pomocą świeczki i na której mógł odtwarzać rozłożenie figur z głównej planszy i planować ruchy. Sterowanie ręką Turka odbywało się za pomocą pantografu sprawiającego, że drewniany manekin powtarzał ruchy,

---

<sup>147</sup> Tamże s. 28.



Relacja ta konstruowana jest poprzez szukanie podobieństw i różnic. Drewniana kukła, przebrana w orientalne szaty, nadawała mechanizmowi podmiotowy charakter poprzez ukonkretnienie obrazu rywala w rozgrywce szachowej. Założeniem mechanizmu była reprodukcja tego, co ludzkie za pomocą narzędzi technologicznych. Imitował on nie tylko zdolności intelektualne, ale również wygląd człowieka. Sam manekin może być interpretowany jako silnie oddziałujący na percepcję interfejs urządzenia. Nie reprezentował on działania mechanizmu, przeciwnie, działał on na rzecz jego ukrywania, odwracania uwagi od tego, co wewnątrz, w podobny sposób, w jaki funkcjonują współczesne interfejsy cyfrowe. Kwestia antropomorfizacji automatycznych systemów cyfrowych stanowić będzie istotny problem do omówienia w drugiej części niniejszego rozdziału.

Podobnie jak o „Kaczce” Vaucansona, również o mechanicznym „Turku” można myśleć jako o wynalazku, który funkcjonował na dwóch płaszczyznach: technologicznej i społecznej. Ta pierwsza płaszczyzna pozostawała tajemnicą dla odbiorców spektaklu, podczas gdy w ramach tej drugiej w znaczący sposób oddziaływał na procesy społeczne i kulturowe. Wpływ społeczny, jaki wywarł wynalazek doskonale obrazują kolejne dwa przykłady. Pierwszym z nich jest historia Edmunda Cartwrighta, który należał do grona entuzjastów mechanicznego „Turka”. Przed wybuchem pierwszej rewolucji przemysłowej powszechnie uważano za niemożliwe stworzenie automatycznego krosna. Tkanie różnorodnych wzorów uznawano za czynność złożoną i wymagającą kreatywności, która nie pozwala ujarzmić się procesom automatyzacji, a przynajmniej za pomocą dostępnych ówczesnie narzędzi. Cartwright, angielski wynalazca, miał okazję zobaczyć mechanicznego „Turka” i wierzył, że jest to prawdziwy mechanizm. Zainspirowany tym osiągnięciem doszedł do wniosku, że skoro mogła powstać tak wyrafinowana maszyna, która potrafi imitować ludzkie procesy intelektualne, stworzenie automatycznego krosna również nie może być niemożliwe. Trzy lata zajęło naukowcowi spełnienie swojego zamierzenia i stworzenie pierwszego automatycznego krosna tkackiego.<sup>149</sup>

Drugim przykładem jest historia innego wynalazcy, Charlesa Babbage’a, który również, jeszcze jako dziecko, miał okazję zobaczyć mechanicznego „Turka”, jednak w przeciwieństwie do Cartwrighta, był sceptyczny wobec prawdziwości wyjaśnień jego działania. Uznał, że automat stworzony przez Kempelena musi być sterowany przez człowieka, mimo to był przekonany, że jest możliwe stworzenie maszyny, która zautomatyzowałaby procesy logiczne i obliczeniowe. Babbage skonstruował później

---

<sup>149</sup> Tamże s. 69.

pierwszą maszynę obliczeniową, nazywaną „maszyną Babbage’a”, którą uznaje się za jeden z pierwszych kamieni milowych w historii rozwoju komputeryzacji.<sup>150</sup>

Fakt, że sama idea mechanicznego gracza szachowego przyspieszyła rozwój automatycznych krosien i maszyn obliczeniowych wskazuje na to, że postęp technologiczny nie daje się sprowadzić wyłącznie do poziomu technicznych innowacji, nie jest prostym ulepszaniem gotowych narzędzi, ale posiada on również aspekt poza technologiczny. Przykłady mechanicznej „Kaczki” i „Turka” pokazują, że za pomocą technologii mogą wyrażać się również pragnienia, lęki i sądy o świecie jej twórcy, a następnie oddziaływać na wyobrażenia dotyczące rzeczywistości jej odbiorców. Ta społeczna zależność technologii będzie kluczowym problemem omawianym w niniejszym rozdziale w kontekście współczesnych technologii cyfrowych. W świetle tego podejścia, oprogramowanie nie daje się sprowadzić do algorytmicznych obliczeń, lecz może być interpretowane jako problem społeczny.

Kontekst historyczny związany z XVIII-wiecznymi automatami otwiera niniejszy rozdział po to, aby podkreślić kilka tez kluczowych dla tej części dysertacji. Wytwarzanie zautomatyzowanej technologii ma dla człowieka charakter tożsamościowy. Oznacza to, że w konsekwencji produkcji technologii o sprawczym charakterze, wątpliwości poddane zostają dotychczasowe kategorie, za pomocą których człowiek był w stanie określać siebie i odróżniać od otaczającego go świata. W odniesieniu do rozwoju automatów, widoczne było to na przykładzie redefinicji rozumienia tego, czym są procesy życiowe i intelektualne w kontekście rozwijających się zasad mechaniki. Należy podkreślić fakt, że wytwarzając technologię zautomatyzowaną, a więc posiadającą sprawczość, człowiek określa swoją relację z nią na podstawie różnic i podobieństw. Proces ten zauważyć można na przykładzie omawianej ewolucji etymologicznej słowa „automat”, które przed erą mechanizacji określało również człowieka, jednak w XVII wieku zostało przypisane wyłącznie do zjawisk technologicznych. Eksperymenty z mechanizmami z XVIII wieku pokazały, że zautomatyzowana technologia stała się podatnym gruntem dla symulacji ludzkich cech za pomocą narzędzi technicznych. Z tej perspektywy tworzenie zautomatyzowanych wynalazków również nabiera tożsamościowego charakteru dla człowieka, ponieważ celem tego rodzaju przedsięwzięć jest stworzenie swego rodzaju kopii, przekazanie podobieństw, symulacja zdolności i cech ludzkich. W konsekwencji tak uwarunkowanego procesu twórczego człowiek, zarówno z perspektywy wytwarzania, jak i interakcji ze

---

<sup>150</sup> Tamże s. 98; 141.

zautomatyzowaną technologią, posiada tendencję do nawiązywania afektywnej relacji z nią. Z perspektywy produkcji może to oznaczać, że w procesie wytwarzania technologii istotną rolę mogą odgrywać wyobrażenia twórcy na temat rzeczywistości, a więc pewne, w mniejszym lub większym stopniu utajone, treści mentalne. Z kolei od strony interakcji tendencja do nawiązywania afektywnej relacji z technologią objawia się w postaci interpretowania jej działania zgodnie z posiadanymi przekonaniami i odczuciami. Prowadzi to do wytwarzania narracji na temat technologii, które funkcjonują w obszarze społecznym, ale nie odpowiadają temu, w jaki sposób owa technologia działa, tak jak miało to miejsce w przypadku mechanicznej „Kaczki” i „Turka”. Tak zarysowaną problematykę wytwarzania i interakcji ze zautomatyzowaną technologią chciałabym teraz omówić w kontekście współczesnych technologii cyfrowych.

## **Rozdział 2**

### **Spoleczne aspekty wytwarzania oprogramowania**

*Każdy program komputerowy jest powstałym w umyśle modelem rzeczywistego procesu myślowego.*<sup>151</sup>

Alan J. Perlis

W kontekście automatów, opisywanych w części wprowadzającej, przedstawiałam to, w jaki sposób dominująca wówczas filozofia mechanicyzmu, będąca podstawowym narzędziem konceptualnym do poznawania świata, wyrażana była za pomocą odkryć technologicznych. Prace nad uzyskaniem mechanicznej imitacji życia stały się urzeczywistnieniem ideologicznych konceptów silnie oddziałujących w tamtym czasie. W niniejszym rozdziale będę starała się pokazać, że w przypadku technologii cyfrowej, wykorzystującej oprogramowanie, wartości kulturowe, społeczne, polityczne i ideologiczne silnie oddziałują na proces jej wytwarzania. Co ciekawe, choć filozofia mechanicyzmu przestała być użyteczna, gdy popularność zyskały inne teorie fizyczne niż klasyczna mechanika, mimo to powróciła ona w XX wieku w ramach dyskusji o sztucznej inteligencji.<sup>152</sup>

Wytwarzanie oprogramowania bowiem nie jest aktem czysto technologicznym, ale głęboko zanurzonym w kulturze procesem odzwierciedlającym rozmaite wartości społeczne i ideologiczne. Piotr Celiński, nawiązując do rozważań Eugene’a Thackera i Alexandra R. Gallowaya, pisał o kolejnych poziomach oprogramowania w następujący sposób:

Im bliżej do użytkownika i jednocześnie im dalej od hardware, tym bardziej te warstwy odbiegają od prostoty binarnego kodu i tym bardziej stają się „kulturowe” – dostępne na wyższych poziomach języki przybierają postać kodów bliskich językowi mówionemu, posługują się jego słownikiem i gramatyką – są paratekstualne i parawizualne, czyli zostały zakodowane semantycznie, a nie tylko technologicznie<sup>153</sup>

Badacz podkreślał, że na wyższych poziomach oprogramowanie wraz z nabywaniem wartości semantycznych, staje się częścią społecznych oddziaływań. Programiści, którzy wykorzystują oprogramowanie, aby kreować aplikacje, systemy, protokoły czy inne cyfrowe interfejsy, korzystają z języka, którego znaczenie rozumieją, a który jednocześnie, na niższym poziomie, jest zrozumiały dla maszyny (hardware’u). Właśnie dlatego

---

<sup>151</sup> A.J. Perlis, *Foreword*, w: H. Abelson, G.J. Sussman, *Structure and Interpretation of Computer Programs*, The MIT Press, Cambridge, Mass. 1996, s. XI. Tłumaczenie A.P.

<sup>152</sup> R. Kirk, *Mechanism*, w: *The Oxford Companion to Philosophy*, red. T. Honderich, Oxford University Press, Oxford 2005, s. 543.

<sup>153</sup> P. Celiński, dz. cyt., s. 107. W przywołanym fragmencie badacz odnosił się do ustaleń Eugene Thackera i Alexandra R. Gallowaya z *The Exploit. A Theory of the Networks*, University of Minnesota Press, Minneapolis 2007, s. 42–45.

o wytwarzaniu oprogramowania można myśleć jako o procesie technologiczno-społecznym. W rozdziale tym proponuję zatem analizę tego, w jaki sposób społeczne i ideologiczne wartości zostają zakodowane w działaniu systemów. Kluczową kategorią dla rozważań na temat translacji wyobrażeń, uprzedzeń, lęków i przekonań na poziom technologiczny działania oprogramowania, będzie pojęcie modelowania.

## 2.1 Model – wprowadzenie

Rozważania zawarte w niniejszym rozdziale koncentrować się będą w głównej mierze wokół kategorii modelu. Model jest bardzo ogólnym pojęciem, przez które rozumieć można system założeń i pojęć, które pozwalają opisywać jakiś fragment rzeczywistości. Każda nauka wytwarza swoje własne modele, które służą jej za narzędzia opisu wiedzy. Ogromny wpływ na rozpowszechnienie się metod opartych na modelowaniu wśród większości dyscyplin naukowych miało ukształtowanie się cybernetyki jako odrębnej nauki w połowie XX wieku.<sup>154</sup> Nazwa „cybernetyka” została rozpowszechniona przez matematyka, Norberta Wienera, uznawanego za ojca tej dziedziny, w książce *Cybernetyka czyli sterowanie i komunikacja w zwierzęciu i maszynie*, wydanej w 1948 roku. Jej autor zestawiał ze sobą wiedzę z obszaru elektronicznej informatyki cyfrowej (będącej w tamtym czasie w początkowej fazie rozwoju), teorię informacji, wczesne rozpoznania dotyczące sieci neuronowych, teorię serwomechanizmów oraz sprzężenia zwrotnego, rozpoznania wypracowane na gruncie nauk społecznych, psychologii, psychiatrii, szczególnie teorię dotyczącą sposobów podejmowania decyzji.<sup>155</sup> Ze współczesnej perspektywy nazwa tej dziedziny, przez przedrostek „cyber”, może kojarzyć się przede wszystkim z nauką dotyczącą systemów informatycznych.<sup>156</sup> Jednak etymologia tego określenia odnosi się do to

---

<sup>154</sup> W kontekście tego jak, rozwój cybernetyki kontrolowany był w ramach zimnej wojny najpierw przez amerykańskie, a później sowieckie władze zob. A. Jelewska, M. Krawczak, *Trudne relacje sztuki, nauki i technologii w Polsce*, w: *Sztuka i technologia w Polsce. Od cyberkomunizmu do zwrotu cyfrowego*, Wydawnictwo Naukowe UAM, Poznań 2014.

<sup>155</sup> A. Pickering, *The Cybernetic Brain: Sketches of Another Future*, University of Chicago Press, Chicago 2011, s. 3.

<sup>156</sup> Interesującym wątkiem dotyczącym tej kwestii jest fakt, że powszechnie przyjęło się, że przedrostek „cyber” wskazuje na związek znaczeniowy z cyberprzestrzenią, a więc internetem oraz technologią cyfrową. Potoczne znaczenie zostało rozpowszechnione przede wszystkim przez literaturę science fiction, czego przykład stanowi powieść *Burning Chrome* Williama Gibsona z 1982 roku, w której autor jako pierwszy użył określenia „cyberprzestrzeń” oraz późniejszy *Neuromancer* z 1984 roku, w którym cyberprzestrzeń określana była również jako macierz (*matrix*). Motywy gibsonowskie posłużyły jako inspiracja do późniejszych, kultowych dzieł science fiction, jak np. trylogia filmowa *Matrix*, ale też przez analogię do nowo przedstawionego terminu, zainspirowały ruch określany mianem cyberpunku. Współcześnie większość

greckiego słowa *kybernētes*, oznaczającego osobę stojącą u steru.<sup>157</sup> Dziedzina ta skupia się przede wszystkim na procesie sterowania i komunikacji wewnątrz systemów w ogólnym znaczeniu, nie tylko w zawężeniu do informatyki. Jest to „nauka o systemach sterowania realizowanych w przyrodzie żywej, technice i społeczeństwie oraz o przekazywaniu i przekształcaniu informacji w tych systemach.”<sup>158</sup> Podstawowym celem tej dziedziny jest wytwarzanie modeli reprezentujących działanie systemów. Modele cybernetyczne posiadają dychotomiczną naturę, z jednej strony są abstrakcyjne, ponieważ bazują na logice i matematyce, z drugiej zaś strony ukonkretniają się przez zastosowanie ich na rzecz konkretnego przypadku, w ramach konkretnej dyscypliny.<sup>159</sup> Ujęcie cybernetyczne, polegające na wytwarzaniu modeli, łączących wartości tradycyjnie pojętych nauk ścisłych i humanistycznych, okazało się być na tyle uniwersalne, że szybko rozprzestrzeniło się wśród innych dyscyplin. Z tego powodu cybernetyka jest trudna do określenia jako odrębna nauka, jest raczej interdyscyplinarnym podejściem, czasami określanym nawet jako antydyscyplinarne, przez powszechność swojego zastosowania.<sup>160</sup> Modelowanie cybernetyczne, mające na celu przedstawienie działania systemu (jakikolwiek: informatycznego, społecznego, ekonomicznego, politycznego czy innego) było odpowiedzią na rosnącą potrzebę wypracowania sprawnej automatyzacji procesów sterowania, czy po prostu zarządzania, w ramach rozwijającego się społeczeństwa cyfrowego. Metody modelowania przeniknęły bardzo szybko do nauk społecznych i humanistycznych przez powszechne zapotrzebowanie na optymalizację rozwiązań związanych z analizą zjawisk masowych. Jedną z przyczyn szybkiego rozpowszechnienia się podejścia cybernetycznego zarówno wśród dyscyplin ścisłych, jak i humanistycznych, jest to, że pojęcie systemu funkcjonowało w ich obszarze już wcześniej. Na przykład, ekonomia polityczna wykorzystywała metody typowe dla cybernetyki jeszcze zanim ta druga zdążyła się ukształtować jako oficjalna nauka. Należały do nich: rachunek podobieństwa, metody

---

określeń rozpoczynających się od przedrostka „cyber” (cyborg, cyberatak, cyberprzestępczość, cyberszpieg, cyberterrorizm itd.) odnosi się do znaczenia nabytego dzięki kulturze popularnej, a nie do etymologii związanej z cybernetyką.

<sup>157</sup> Inspiracją Wienera w tworzeniu nazwy „cybernetyka” była maszyna parowa „Governor”, która nawiązywała do łacińskiego określenia „gubernator”, oznaczającego sternika. Wiener oparł się na greckim odpowiedniku tego słowa „*kybernetes*”. Zob. A. Pickering, *Cybernetic Brain...*, dz. cyt., s. 6.

<sup>158</sup> *Cybernetyka*, w: *Uniwersalny Słownik Języka Polskiego*, red., S. Dubisz, Wydawnictwo Naukowe PWN, Warszawa 2008, s. 489.

<sup>159</sup> M. Mazur, *Modelowanie cybernetyczne i jego przydatność w modelowaniu procesu dydaktycznego*, w: *Problemy modelowania procesów dydaktycznych*, red. E. Berezowski, PWN, Warszawa 1978, online: [http://www.autonom.edu.pl/publikacje/mazur\\_marian/modelowanie\\_cybernetyczne\\_i\\_jego\\_przydatnosc\\_w\\_modelowaniu\\_procesow\\_dydaktycznych.php](http://www.autonom.edu.pl/publikacje/mazur_marian/modelowanie_cybernetyczne_i_jego_przydatnosc_w_modelowaniu_procesow_dydaktycznych.php), dostęp: 06.11.2023. Więcej o tym, jak w kontekście kulturowym modele cybernetyczne rozwijały się i przekształcały technokulturowy paradygmat nowoczesności zob. A. Jelevska, *Ekotopie. Ekspansja technokultury*, Wydawnictwo Naukowe UAM, Poznań 2014.

<sup>160</sup> A. Pickering, dz. cyt., s. 9.

statystyczne, modelowanie, programowanie liniowe i dynamiczne. W połowie XX wieku, dzięki popularności cybernetyki, metody te zostały zaadaptowane przez inne dziedziny, takie jak: psychologia, teoria prawa, teoria i praktyka dydaktyczna, lingwistyka czy historia.

Współcześnie, o procesie wytwarzania modeli w ramach konkretnych dyscyplin nie myśli się wyłącznie w kontekście znaczenia naukowego lub technologicznego, ale poszerza się tę perspektywę o aspekt społeczny, poprzez zwrócenie uwagi na poznawczy charakter procesu dotyczący samej osoby (lub osób) tworzących model.<sup>161</sup> Również wielu matematyków rozszerzało w ten sposób ujmowanie modelowania. Między innymi Clive Dym, omawiając definicję modelowania matematycznego, zwracał uwagę na to, że jest to czynność o charakterze poznawczym. Pisał, że „modelowanie jest aktywnością, kognitywną aktywnością, w której wyobrażamy sobie i tworzymy modele, aby opisać to, jak działają urządzenia i obiekty będące przedmiotem zainteresowania.”<sup>162</sup> Podejście to było prezentowane również przez analityczkę danych, Cathy O’Neil, która na owym kognitywnym aspekcie modelowania oparła krytykę działania współczesnych systemów i przedstawiła zagrożenia społeczne, które wynikają właśnie z poznawczej warstwy modelowania. Badaczka w książce *Broń matematycznej zagłady. Jak algorytmy zwiększają nierówności i zagrażają demokracji*, określiła go jako:

nic innego, jak abstrakcyjne przedstawienie jakiegoś procesu, niezależnie od tego, czy dotyczy baseballu, łańcucha dostawców rafinerii, działań obcych rządów, czy widzów oglądających film w kinie. Przetwarzany komputerowo lub w naszych głowach model wykorzystuje to, co wiemy, w celu prognozowania rozmaitych efektów w różnych sytuacjach. Każdy z nas nosi w głowie tysiące takich modeli. Mówią nam one czego możemy się spodziewać i kierują naszymi decyzjami.<sup>163</sup>

W niniejszym rozdziale, model będzie więc rozumiany jako kognitywna struktura wyrażająca określony obraz świata, która pozwala wyciągać wnioski na jego temat i podejmować decyzje. Model powstaje w efekcie procesów myślowych i wyobraźniowych człowieka, ale może być implementowany do zewnętrznych systemów logistycznych lub informatycznych.

---

<sup>161</sup> Na gruncie psychologii korzysta się z terminu modelowania w ramach teorii społecznego uczenia się Alberta Bandury, zgodnie z którą człowiek uczy się poprzez obserwację cudzych zachowań i ich skutków (obserwacja modelu). Pisząc o zjawisku wytwarzania modeli na gruncie nauk humanistycznych, nie chcę odwoływać się do tej teorii, ale mam na myśli wywiedzione z gruntu cybernetyki rozumienie modelu, wykorzystywane na gruncie nauk społecznych.

<sup>162</sup> C. Dym, *Principles of Mathematical Modeling*, Academic Press, Nowy Jork 2004, s. 3. Tłumaczenie A.P.

<sup>163</sup> C. O’Neil, *Broń Matematycznej Zagłady...*, dz. cyt. s. 44-45.

Mimo że matematycznym strukturom często przypisuje się cechę obiektywności, działanie systemu informatycznego realizującego model stworzony przez człowieka może posiadać charakter wykluczający lub krzywdzący, może być podyktowane uprzedzeniami lub stereotypami. Tworzenie modelu to proces, którego nieodłączną częścią jest redukcja. Polega on na wykreowaniu pewnego uproszczonego obrazu świata oraz wyłączeniu z jego zakresu wszelkich niepotrzebnych elementów, które nie sprzyjają działaniu modelu; nie realizują założonego przez niego celu. Proces ten jest dobrze widoczny na przykładzie nawigacji Google Maps, która przedstawia świat jako układ dróg, rond, skrzyżowań, mostów, tuneli i pomija wszelkie inne elementy, które stanowią część rzeczywistego świata, jednak nie są potrzebne dla ustalania trasy.<sup>164</sup> Przez fakt, że model jest uproszczoną reprezentacją pewnych procesów myślowych i wyobrażeń, zawsze będzie posiadał swoje ślepe punkty, luki informacyjne, których istnienie niesie ze sobą ryzyko szkodliwego lub krzywdzącego działania wobec osób znajdujących się w obszarze jego oddziaływania. Konsekwencje luk informacyjnych nie wydają się być znaczące w przypadku przywoływanego tu przykładu systemu nawigacji samochodowej, jednak stają się one bardziej namacalne chociażby w kontekście systemów bankowych, kredytowych, rekrutacyjnych czy walidacyjnych, których celem jest ocena jednostek ludzkich, wypracowanie uproszczonych kategorii interpretacji złożonych problemów społecznych. W tym obszarze dochodzi do wielu kontrowersyjnych pod względem etycznym sytuacji.

## **2.2 Społeczne aspekty wytwarzania oprogramowania – trzy perspektywy**

Po krótkim wprowadzeniu pojęcia modelu, chciałabym omówić trzy perspektywy, przedstawiające to, w jaki sposób wytwarzanie oprogramowania w oparciu o wcześniej opracowane modele, staje się przyczyną przenoszenia się do obszaru działania technologii ukrytych w owych modelach treści ideologicznych, przekonań i sądów na temat świata. Konsekwencją tego przeniesienia może być dyskryminujące lub krzywdzące działanie programów. Modele warunkujące powstawanie oprogramowania stanowią również element, który łączy porządek technologiczny z porządkiem społecznym.

Pierwsza perspektywa dotyczyć będzie problemu modeli wytwarzanych na rzecz badań opartych na metodzie analizy ilościowej. Przez wzgląd na pozornie obiektywny

---

<sup>164</sup> Tamże, s. 47.

charakter wyników uzyskanych za ich pomocą, stają nie one podstawą działania oprogramowania wykorzystywanego w obszarach społecznych obarczonych w największym stopniu odpowiedzialnością publiczną, a więc chociażby w sądownictwie, sektorze bankowym czy w edukacji.

Druga perspektywa skupiać się będzie na zagadnieniu modelowania kognitywnego, a więc wypracowanej na gruncie nauk kognitywnych metody symulacji ludzkich procesów poznawczych za pomocą oprogramowania. Celem tej części będzie przedstawienie tego, jak wyobrażenia na temat ludzkiego ciała przenoszone są do obszaru technologicznego i w sposób zwrotny, jak owe symulacje stają się podstawą do wyprowadzania wniosków na temat ludzkich procesów poznawczych. Istotnym elementem tych rozważań będzie przedstawienie podobieństw i zależności pomiędzy funkcjonowaniem oprogramowania w kulturze a procesem wytwarzania wiedzy na gruncie naukowym.

Trzecia perspektywa badać będzie związek oprogramowania z językiem naturalnym. Konsekwencje tej zależności omówione zostaną w odniesieniu do najpowszechniejszego obecnie modelu wytwarzania oprogramowania, jakim jest *Agile*. Oprogramowanie już na etapie samego powstawania warunkowane jest przez zawarte w języku naturalnym treści kulturowe. Celem tej części będzie opisanie tego, w jaki sposób software i język naturalny są ze sobą związane, jakie mają podobieństwa oraz na jakiej podstawie uczestniczą wspólnie w wytwarzaniu treści kulturowych.

### **2.2.1 Perspektywa pierwsza. Modelowanie wobec analizy ilościowej**

Era Big Data stanowi wyzwanie dla indywidualnego podejścia do człowieka. Ogromne zasoby danych, produkowane codziennie w masowych ilościach przez użytkowników technologii cyfrowych przez wzgląd na swój rozmiar nie pozwalają interpretować się inaczej niż za pomocą narzędzi analizy ilościowej. Niepokój o zatracenie indywidualnego podejścia w tym obszarze zdaje się odzwierciedlać istniejące już wcześniej na gruncie naukowym napięcie pomiędzy metodologią badań ilościowych i jakościowych.

Władysław Paluchowski omawiając konflikt pomiędzy podejściem ilościowym a jakościowym, trafnie przedstawił ważną różnicę w samej postawie względem wypracowywanej tezy. Pisał on, że „(...) badacz ilościowy poszukuje danych weryfikujących teorię, a badacz jakościowy – teorii, która wyjaśnia posiadane przez niego

dane.”<sup>165</sup> Opisane tutaj podejście badawcze, charakterystyczne dla metod ilościowych, które polega na poszukiwaniu danych weryfikujących zakładaną teorię, jest kolejnym argumentem przemawiającym za tym, że modele, także te w postaci algorytmicznych narzędzi, mają tendencję do samopotwierdzania własnych założeń. Jeśli celem analizy zbioru danych jest weryfikacja przypuszczeń sformułowanych przed jej rozpoczęciem, potwierdzająca informacja zwrotna może być traktowana jako podstawa do uznania modelu za zasadny. W ramach rozważań dotyczących relacji pomiędzy modelowaniem a analizą ilościową, chciałabym przedstawić i poddać krytyce kilka przykładów samopotwierdzających się modeli, wykorzystujących metody ilościowe oraz omówić konsekwencje społeczne, płynące z takiego ich działania.

Jeśli celem badań ilościowych jest interpretacja zbioru danych, to za wynikiem analizy ilościowej stać będą zawsze liczby i statystyki – wartości, które przyjęło się traktować jako pozbawione intencjonalności, będące surową reprezentacją określonego fragmentu rzeczywistości. Proces, w ramach, którego dane liczbowe, będące pomiarem jakichś wartości społecznych, stają się znaczące i prawomocne nie tylko odzwierciedla wzorce aktywności kulturowej, ale także w znacznym stopniu przyczynia się do rekonceptualizacji świata społecznego, poprzez wprowadzanie nowych kategorii analitycznych.<sup>166</sup> Nadmierne zaufanie do danych, wraz z tendencją do tego, aby uznawać liczby za odpowiednią i wiarygodną podstawę do wyciągania wniosków na temat rzeczywistości zostało opisane w analizie politycznej filozofii ujęcia ilościowego Theodora M. Portera. W książce *Trust in numbers: the pursuit of objectivity in science and public life* badacz podkreślał, że zgoda na to, aby liczby społeczne (a więc takie, które mierzą pewne aktywności społeczne) strukturyzowały rzeczywistość kulturową, ma charakter społecznej umowy. Tego rodzaju pomiary pozostają w mocy tak długo, jak długo nie są podawane w wątpliwość:

Stopnie w szkole, wyniki na standaryzowanych egzaminach i dolna linia na arkuszu księgowym nie mogą działać skutecznie, chyba że ich wiarygodność (...) jest akceptowana przez ludzi, których osiągnięcia lub wartość, którą rzekomo mierzą. Jeśli tak jest, miary odnoszą sukces, nadając kierunek tym właśnie działaniom, które są mierzone.<sup>167</sup>

---

<sup>165</sup> W.J. Paluchowski, *Spór metodologiczny czy spór koncepcji. Badania ilościowe vs jakościowe*, „Roczniki psychologiczne” 2010, tom 10, nr 1, s. 9.

<sup>166</sup> T.M. Porter, *Trust in numbers. The pursuit of objectivity in science and public life*, Princeton University Press, Princeton, New Jersey 1995, s. 33-48.

<sup>167</sup> T.M. Porter, dz. cyt., s. 45. Tłumaczenie A.P.

Z tego powodu wszelkie ruchy wyrażające sprzeciw lub aprobatę wobec funkcjonujących w społeczeństwie modeli, które nim zarządzają, posiadają ogromne znaczenie z perspektywy ich utrzymywania się bądź przedawnienia.

Liczby pełnią rolę swoistego „medium” w sytuacji, kiedy trzeba zaprezentować tezę, która ma być zrozumiała niezależnie od dyscypliny naukowej czy obszaru aktywności społecznej. Posiadają one dominujące znaczenie w kontekście wytwarzania wiedzy uznawanej powszechnie za obiektywną. Hipotezy sformułowane za pomocą liczb są czytelniejsze niż te, wyrażone za pomocą narzędzi opisu, charakterystycznych dla danej dyscypliny. Z tego powodu modele liczbowe odznaczają się uniwersalnością i posiadają niezwykle potężny potencjał interdyscyplinarny. Podkreślał to również Porter słowami:

Niezwykła zdolność liczb i obliczeń do przekraczania granic dyscyplin, a nawet granic narodowych i łączenia dyskursu akademickiego z politycznym możliwa jest w dużej mierze dzięki zdolności do upraszczania. W wymianie intelektualnej, podobnie jak w ekonomicznych transakcjach, liczby są medium, przez które i dzięki któremu odmienne pragnienia, potrzeby i oczekiwania stają się w jakiś sposób współmierne.<sup>168</sup>

To, co Porter określił jako zdolność do omijania głębokich kwestii może być rozumiane jako przywoływana już tendencja do redukcji, upraszczania, charakterystyczna dla procesu modelowania. W związku z tym modele, zasilane przez dane liczbowe mogą stanowić medium komunikacji i wymiany wiedzy, niezależnie od rodzimych dyscyplin. W oparciu o to spostrzeżenie, w niniejszej części rozważań przedstawiona zostanie analiza zróżnicowanych, programistycznych modeli, wywodzących się z odmiennych obszarów aktywności społecznej. Przywoływane przykłady posiadać jednak będą pewien punkt wspólny, którym jest oparcie się na danych liczbowych i statystycznych. Znaczenie liczb społecznych ma dla tej części rozważań ogromną wagę, przez wzgląd na to, że są one podstawą dla ujęcia ilościowego. Z kolei modele wypracowane za pomocą narzędzi analizy statystycznej dają się łatwo aplikować do postaci algorytmicznej, przede wszystkim dlatego, że podstawą statystyki jest analiza zbioru danych.

Na początku chciałabym sięgnąć po przykład instytucji sądowych jako obszarów, w których kluczowe znaczenie odgrywa możliwość rzetelnego, sprawiedliwego i obiektywnego orzekania o członkach danego społeczeństwa. Zadaniem procesów sądowych jest analiza indywidualnych przypadków, ale w kontekście ogólnie przyjętych

---

<sup>168</sup> Tamże, s. 86. Tłumaczenie A. P.

zasad prawnych, w celu wymierzenia kary bądź rozstrzygnięcia sporów. Ich działanie jest więc kompromisem pomiędzy dwoma porządkami – jednostkowym i powszechnym/zbiorowym. Pomimo że sądy są uwikłane w zależności społeczne i polityczne, wyroki, który wydają z założenia mają mieć charakter niezawisły i niezależny. Oznacza to, że subiektywne przekonania osądzających lub ich dyskryminujące uprzedzenia nie powinny stanowić podstawy podejmowanych decyzji. Sposób działania procesu sądowego sam w sobie jest rodzajem modelu, a jego działanie zostało przyjęte jako zasadne w wyniku społecznej umowy. Tak więc, zgodnie z założeniami politycznej filozofii ujęcia ilościowego Theodore’a M. Portera, działanie procesów sądowych w obecnej formie podtrzymywane jest przez społeczną zgodę na nie. I odwrotnie, potrzebny byłby bunt i sprzeciw przeciwko nim, aby móc wprowadzić jakąś zmianę.

Cathy O’Neil analizując społeczne aspekty funkcjonowania modeli i procesów algorytmizacji, odwoływała się do przykładu pochodzącego z amerykańskiej historii sądownictwa, a konkretnie przypadku czarnoskórego mężczyzny, Duane Bucka, który w 1997 roku stanął przed ławą przysięgłych w okręgu Harris w Teksasie, aby usłyszeć wyrok za zabicie dwójki osób. Zadaniem ławy było zdecydowanie pomiędzy karą śmierci a dożywociem z szansą na warunkowe zwolnienie. Jako świadek przemawiał między innymi psycholog, Walter Quijano, który zwracał uwagę na to, że zgodnie z badaniami, czynnik rasowy ma znaczenie w kontekście skłonności do popełniania przestępstw i przemawia on niekorzystnie na rzecz rasy czarnej. Argument ten został podchwycony przez prokurator i przywołany w jej mowie końcowej. Ostatecznie ława przysięgłych skazała Bucka na karę śmierci. Psycholog, który przyczynił się do tego wyroku, podejmował głos w sprawie wielu innych, podobnych przypadków, a później ujawniono, że mógł on działać na zlecenie prokuratury.<sup>169</sup> Historia ta pokazuje kryzys obiektywności w podejmowaniu decyzji na gruncie sądownictwa. Aby mu przeciwdziałać wprowadzono chociażby projekt ustawy *Racial Justice Act*, który zabraniał podejmowania decyzji dotyczącej wyroku śmierci w oparciu o czynniki rasowe. Na kanwie tej tendencji podjęto wysiłek opracowywania specjalnych rodzajów modeli, które pomogłyby wypracować pewne uniwersalne kategorie oceny przestępców. Starania te oparte zostały na przekonaniu, że obiektywność może zostać zapewniona przez system decyzyjny, oparty na danych i statystyce. W ten sposób zaczęły powstawać tak zwane modele recydywy, które również przyjmowały postać systemów programistycznych.

---

<sup>169</sup> C. O’Neil, *Broń matematycznej zagłady...*, dz. cyt., s. 50.

Najbardziej powszechnym modelem recydywy jest LSI-R (The Level of Service Inventory – Revised), który opracowano już w 1995 roku. Zaimplementowany on został również do postaci programu komputerowego. Jego zasadniczym elementem jest kwestionariusz, który ma na celu zebranie danych psychometrycznych badanego w dziesięciu różnych kategoriach: historia kryminalna, edukacja/zatrudnienie, sytuacja finansowa, rodzina/stan cywilny, warunki mieszkalne, czas wolny/rekreacja, znajomi, alkohol/narkotyki, emocjonalne i osobiste problemy oraz postawa.<sup>170</sup> Na podstawie zapewnionych odpowiedzi, LSI-R klasyfikował badanych w ramach trzech różnych kategorii, wyrażających poziom ryzyka związanego z recydywą: wysoki, średni oraz niski. Jak widać, czynniki rasowe pozornie nie miały być poddawane ocenie w tym kwestionariuszu. W przeważający sposób skupiał się on na uwarunkowaniach środowiskowych.

Zgodnie z opisaną wcześniej specyfiką działania modelu, również LSI-R posiadał swoje konieczne ślepe punkty, luki informacyjne, przez które nie stanowił on idealnego narzędzia oceny. Badania udowodniły, że test mimo wszystko dopuszcza działania o charakterze dyskryminującym, na przykład przez to, że jego skuteczność jest o wiele niższa w przypadku kobiet.<sup>171</sup> Test również nie uwzględniał tego, że czynniki rasowe, mimo że pozornie nie poddawane ocenie, mogą warunkować odpowiedzi badanych. Cathy O’Neil wskazywała na to, że działanie LSI-R posiada tendencję do utrwalania oceny ryzyka badanych.<sup>172</sup> Przykładowo, jeśli komuś przypisana została wysoka kategoria ryzyka, oznaczało to zazwyczaj, że jest osobą bezrobotną, pochodzącą z biednej dzielnicy, a w jego rodzinie odnotowano już przypadki przestępczości. Przypisana kategoria mogła przyczyniać się do surowszej kary.<sup>173</sup> Więzień, któremu przypisano dłuższy pobyt w więzieniu, dłużej podlegał separacji społecznej, zaś czas, który spędzał z innymi kryminalistami, zwiększał się. Te czynniki sprawiały, że kiedy kończył wyrok i wracał do niesprzyjającego środowiska, dodatkowo skategoryzowany już etykietą wysokiego ryzyka, trudniej mu było działać w inny sposób, niż potwierdzający przewidywania testu. Dochodzi tu więc do negatywnego sprzężenia zwrotnego. Tym samym, LSI-R zamiast opisywać rzeczywistość,

---

<sup>170</sup> I. Watkins, *The Utility of Level of Service Inventory – Revised (LSI-R) Assessments within NSW Correctional Environments*, „Research Bulletin (Corrective Services NSW)” 2011, nr 28, s. 2.

<sup>171</sup> Tamże, s. 6.

<sup>172</sup> C. O’Neil, *Broń matematycznej zagłady...*, dz. cyt., s. 54.

<sup>173</sup> LSI-R posiadał różne znaczenie w procesie sądowym w zależności od stanu. Na przykład w Rhode Islands testy te przeprowadzano wyłącznie w celu dopasowania odpowiedniego programu resocjalizacyjnego. Jednak w Idaho i Colorado wynik testu mógł przyczynić się do wyroku. Zob. C. O’Neil, *Broń matematycznej zagłady...*, dz. cyt., s. 53.

stwarzał ją; kreował kategorie, które utrwały się w rzeczywistości i dzięki temu, znajdowały potwierdzenie.

O'Neill podkreślała, że zdolność do „samopotwierdzania się” w stworzonych przez siebie warunkach jest cechą charakterystyczną dla działania modeli w ogóle.<sup>174</sup> Badaczka opisywała niebezpieczeństwo związane z tym, że modele przenoszone są przez programistów do systemów informatycznych, które to zamiast dopasowywać się do rzeczywistości, zaczynają ją stwarzać, zgodnie z zasadami zawartego w nich modelu. W przypadku programistycznych modeli społecznych sprzężenie zwrotne, którego konsekwencją jest, często niesłuszne, podtrzymywanie i potwierdzanie wypracowanych kategorii przejawia się wyraźnie chociażby w modelu zatrudniania na podstawie zarobków, które prowadzi do utrwalania bezrobocia. Innym przykładem są programy do oceny zdolności kredytowej, które mogą utrudniać branie pożyczek osobom sklasyfikowanym w danej kategorii ekonomicznej.<sup>175</sup>

W kontekście LSI-R dochodzi do pewnej sprzeczności: jeśli proces sądowy ma brać pod uwagę indywidualną sytuację jednostki, a jednocześnie wykorzystuje w tym celu model recydywy, to wrażliwość na odrębność osądzanego zostaje zagrożona przez właściwość badań ilościowych, którą Porter określił jako „studiowanie ludzi w klasach, abstrahując od ich indywidualności”<sup>176</sup>. Tak sformułowana charakterystyka badań ilościowych nie musi być jednoznacznie rozumiana jako zjawisko negatywne. Metoda ta powstała przecież w celu usprawnienia działania społeczeństw, w takich obszarach jak choćby standaryzacja leków, analiza spisów ludności, opracowywanie badań naukowych.<sup>177</sup> Etyka badań ilościowych i tworzenia cyfrowych modeli na ich podstawie jest ważną kwestią, która w głównej mierze opiera się na odczarowywaniu obiektywności modelu, a więc uświadamianiu użytkownikom zasad jego działania oraz wskazywaniu na konieczność poddawania danych regularnej rewizji. Aktywności te powinny przeciwdziałać zjawisku moralnego dystansu, które czasami ujawnia się w sytuacji pracy z liczbami i danymi. Dzieje się tak, ponieważ modele ułatwiają podejmowanie decyzji i pozwalają na poczucie przeniesienia odpowiedzialności z korzystającego z narzędzia na samo narzędzie.

Zjawisko moralnego dystansu doskonale obrazuje programistyczny model o nazwie IMPACT, który posłużył do ewaluacji nauczycieli szkolnych w Waszyngtonie. Do jego powstania przyczynił się Adrian Fenty, burmistrz Waszyngtonu motywowany chęcią

---

<sup>174</sup> Tamże, s. 54-55.

<sup>175</sup> Tamże, s. 32.

<sup>176</sup> T.M. Porter, dz. cyt., s. 77.

<sup>177</sup> Zob. Tamże s. 24.

polepszenia wyników słabych szkół miejskich, w których znaczna część uczniów nie osiągała oczekiwanego poziomu wiedzy. Powołane zostało nowe, specjalne stanowisko kanclerza ds. waszyngtońskich, które objęła Michelle Rhee, specjalistka do spraw edukacji. Wychodząc z założenia, że przyczyną niezadowolających wyników uczniów jest niekompetencja nauczycieli, Rhee zleciła firmie konsultingowej Mathematica Policy Research stworzenie modelu, który potrafiłby wskazać nieskutecznych dydaktyków. IMPACT obejmował zestaw dziewięciu kryteriów wydajności obejmujących niemal każdy aspekt pedagogiki, włączając w to klarowność prezentacji, umiejętność zarządzania zachowaniem ucznia czy formułowania pytań i poleceń.<sup>178</sup>

Na skutek działania algorytmicznego modelu ewaluacyjnego zwolnieni zostali pedagodzy, których punktacja znajdowała się poniżej dolnych dwóch procent wyników. Przykładowo, tylko w roku szkolnego 2010/11 zwolnionych zostało dwieście sześć osób.<sup>179</sup> Jak się okazało później część osób, która straciła pracę z tego powodu została oceniona niesłusznie. Wśród poszkodowanych znalazła się Sarah Wysocki, dzięki której sprawa zyskała rozgłos medialny.<sup>180</sup> Ocena jej osoby w wykonaniu programu IMPACT zaskoczyła samą badaną, jej uczniów, rodziców oraz innych nauczycieli, ponieważ w powszechnej opinii była odbierana jako doskonały pedagog, potrafiący wzbudzić zaangażowanie podopiecznych. O zdolnościach nauczycielki pozytywnie świadczyły również liczne nagrody dydaktyczne, które udało jej się zdobyć. Została ona zwolniona, przez nowo dodany w drugim roku działania modelu system zliczania punktów, określany jako model wartości dodanej.<sup>181</sup> Czynnikiem ten miał na celu określenie postępu oraz regresu uczniów oraz przypisanie odpowiedzialności za tę zmianę konkretnemu nauczycielowi. Punktacja wyliczana w ramach tego czynnika stanowiła połowę ogólnej oceny nauczyciela i przez to miała decydujący wpływ na ostateczny wynik – przeważała nawet pozytywne opinie wystawione przez administrację szkoły, rodziców i kadrę. To właśnie z powodu niskiego rezultatu oceny wartości dodanej Wysocki została zwolniona. Cathy O’Neil, analizując ten przykład działania oprogramowania, podkreślała znaczenie tego, że modele charakteryzują

---

<sup>178</sup> B. Turque, *Rhee Spells Out Teaching Expectations With 200-Page 'Learning Framework'*, 23.08.2009, online: <https://www.washingtonpost.com/wp-dyn/content/article/2009/08/22/AR2009082202251.html>, dostęp: 06.11.23.

<sup>179</sup> C. O'Neill, *Broń matematycznej zagłady...*, dz. cyt., s. 28.

<sup>180</sup> B. Turque, *'Creative...motivating' and fired*, 06.03.2012, online: [https://www.washingtonpost.com/local/education/creative--motivating-and-fired/2012/02/04/gIQAwzZpVR\\_story.html](https://www.washingtonpost.com/local/education/creative--motivating-and-fired/2012/02/04/gIQAwzZpVR_story.html), dostęp: 06.11.2023.

<sup>181</sup> W polskich realiach czynnik ten określany jest jako EWD (Edukacyjna Wartość Dodana). Stosuje się go do oceny szkół, a nie nauczycieli. Pozwala on określić tak zwany wskaźnik EWD. Zob. *Edukacyjna Wartość Dodana*, online: <https://ibe.edu.pl/pl/projekty-zakonczone/edukacyjna-wartosc-dodana> dostęp: 06.11.23.

się niską przystawalnością do przypadków granicznych oraz będących wyjątkami od sformułowanych założeń.<sup>182</sup> Wydaje się, że to właśnie z przyczyny wadliwego działania modelu algorytmicznego, na bazie którego powstało oprogramowanie IMPACT, dobra nauczycielka została poszkodowana. Wysocki widziała również możliwość negatywnej oceny wartości dodanej w tym, że na początku roku szkolnego, w którym poddawana była ewaluacji, do jej grupy dołączyło wielu uczniów po otrzymaniu zawyżonych ocen testów w poprzedniej klasie.<sup>183</sup> Okoliczność ta mogła również wyjaśniać drastyczną różnicę między rocznymi ocenami, które zostały sklasyfikowane jako znaczący regres.

Kwestia tego, dlaczego pedagoga została oceniona tak negatywnie nie została jednak oficjalnie wyjaśniona, ponieważ algorytm był własnością intelektualną Mathematica Policy Research, a ujawnianie go naruszyłoby prawa autorskie i własnościowe firmy. Z tego samego powodu nauczyciele, którzy poddawani byli ocenie za pomocą algorytmu IMPACT nie posiadali żadnej możliwości zrozumienia jego działania, a także tego na jakiej podstawie obliczana jest ich punktacja. Przedstawiciele rządowi nie przyjmowali zarzutów dotyczących niesprawiedliwego działania algorytmu. Na przykład Jason Kamras, szef działu kapitału ludzkiego w szkołach Waszyngtonu, zanegował zasadność sprzeciwu Sarah Wysocki wobec oceny jej osoby, powołując się na argument, że stanowi ona wyjątkowo rzadki przypadek, a klasyfikacja pozostałych nauczycieli jako słabych potwierdziła się jako zgodna ze stanem faktycznym.<sup>184</sup> Inna pokrzywdzona przez algorytm osoba, Sarah Bax, wielokrotnie odwoływała się do władz okręgowych, wśród których miała znajomych, z prośbą o wyjaśnienie działania modelu IMPACT. Spotykała się z odpowiedzią, że algorytm jest trudno zrozumiały, zaawansowany, przedstawiano jej jedynie ogólny, mało informacyjny raport techniczny.<sup>185</sup> Jak widać, przedstawiciele władz okręgowych, do których również należał obowiązek zwolnienia nauczycieli wskazanych przez algorytm, przyjmowali bezkrytycznie, a nawet usprawiedliwiali oceny wystawiane przez narzędzie, którego działania sami nie rozumieli. Ich postawa stanowi przykład moralnego dystansu, w ramach którego decyzja o zwolnieniu z pracy nauczycieli i związana z nią odpowiedzialność została zrzucona na deklaratywnie niezrozumiałe oprogramowanie. Wyręczeniu przez działanie systemu IMPACT przedstawiciele władz okręgowych oraz członkowie komisji oceniających obsługujących algorytm, z łatwością zastosowali się do wykreowanego wyniku. Przykład ten wskazuje na problematykę udziału algorytmicznych

---

<sup>182</sup> C. O'Neil, *Broń matematycznej zagłady...*, dz. cyt. s. 31.

<sup>183</sup> B. Turque, *'Creative...Motivating' and fired*, dz. cyt.

<sup>184</sup> Tamże.

<sup>185</sup> C. O'Neil, *Broń Matematycznej Zagłady...*, dz. cyt., s. 33.

modeli w procesie decyzyjnym przedstawicieli instytucji rządowych i stanowi wątpliwe etycznie zjawisko, ponieważ wcześniej nie została rozpatrzona kwestia problemu moralnego dystansu, związanego z delegowaniem decyzji na systemy pomocnicze.<sup>186</sup>

Zgodnie z przytaczaną wcześniej ideą politycznej filozofii ujęcia ilościowego Theodore M. Portera, liczby społeczne są wiarygodne tylko, jeśli są akceptowane przez ludzi, których działania mierzą. Z tego powodu głośny sprzeciw, jaki wywołał algorytm IMPACT wśród przedstawicieli kadry nauczycielskiej, można interpretować jako ruch przeciwdziałający obiektywizacji tego modelu ewaluacyjnego. Znaczenie buntu ocenianych podkreśla potrzebę projektowania modeli społecznych przejrzystych pod względem zasad działania i zrozumiałych zarówno dla tych, którzy wykorzystują je jako narzędzie, jak i tych, na których ono oddziałuje.

\*\*\*

W ramach podsumowania pierwszej z trzech perspektyw społecznego znaczenia modeli w kontekście oprogramowania, można odwołać się do tezy, postawionej przez Cathy O'Neil w samym tytule jej książki przywoływanej w tym rozdziale. Analityczka danych twierdziła, że zjawisko Big Data przyczyniło do wzrostu nierówności społecznych i stanowi zagrożenie dla demokracji. Trudno jednoznacznie stwierdzić, czy można w tym przypadku mówić o wzroście nierówności społecznych, czy raczej trafniejsze byłoby określenie, że owe nierówności wcale nie wzrosły, ale były już obecne w modelach kultywowanych przez kulturę przedcyfrową i jedynie ujawniły się w ramach zjawiska Big Data. Można to zaobserwować chociażby na omawianym przykładzie metody badań ilościowych, która niosła ze sobą ryzyko fałszywej obiektywizacji jeszcze zanim została zaaplikowana do systemów cyfrowych. Również Wendy Hui Kyong Chun podkreślała, że współczesne metody wytwarzania oprogramowania, które kodują segregację społeczną i politykę tożsamości wywodzą się z XX-wiecznych, eugenicznych prób projektowania, czy wręcz „hodowania” dla wytwarzania lepszej przyszłości.<sup>187</sup>

W krytycznym podejściu do funkcjonowania modelu ważne jest, aby posiadać wiedzę o jego działaniu. Sprzężenie aspektów technologicznych ze społecznymi skutkuje tym, że cyfrowe modele ze względu na to, że opierają się na oprogramowaniu, dzielą także jego

---

<sup>186</sup> W ramach dopowiedzenia chciałabym dodać, że po zwolnieniu Sarah Wysocki nie musiała długo szukać pracy, została zatrudniona w Hybla Valley Elementary School w hrabstwie Fairfax.

<sup>187</sup> W. Hui Kyong Chun, *Discriminating Data...*, dz. cyt., s. 35-74.

cechę związaną z ukrywaniem się pod powierzchnią interfejsu i poza percepcją społeczną. Charakteryzują się one również zdolnością do wytwarzania pewnych obrazów rzeczywistości i potwierdzania własnej słuszności w jej ramach. Połączenie tych dwóch cech sprawia, że model okazuje się być strukturą wymykającą się percepcji, skłonną do działania z ukrycia o silnym impakcie społecznym. Właśnie z tego powodu tak istotną rolę pełni jego transparentność, która stanowi jeden z czynników sprzyjających etycznemu działaniu modelu. Wśród wielu nowo powstających inicjatyw przeciwdziałających ukrytemu i nieetycznemu działaniu oprogramowania mającego istotny wpływ na społeczeństwo wymienić można projekt można szeroko omawiany *The Algorithmic Justice League*, założony przez Joy Buolamwini w 2016 roku, którego główną misją jest zwiększanie wśród użytkowników świadomości dotyczącej społecznych konsekwencji działania sztucznej inteligencji. Organizacja oferuje również wsparcie prawnicze dla poszkodowanych w wyniku krzywdzącego działania oprogramowania.<sup>188</sup>

### **2.2.2 Perspektywa druga. Modelowanie kognitywne – relacja nauki i technologii**

Mechaniczne automaty, opisane w kontekstowym wprowadzeniu historycznym dla niniejszego rozdziału, były próbą odwzorowania działania cech fizjologicznych lub intelektualnych za pomocą narzędzi technologicznych. Można powiedzieć, że stanowiły one swego rodzaju modele, reprezentujące bardzo określoną aktywność, typową dla człowieka. Zarówno dla „Trawiącej Kaczki”, jak i dla „Turka”, charakterystyczna była techniczna niedoskonałość, niewystarczalność owego modelu. Luki w ich działaniu uzupełnione zostały narracją twórców. Opowieść o wynalazkach, zawierająca ich potencjalne wyjaśnienie, ożywiała zainteresowanie odbiorców, pobudzała ich wyobraźnię, prowokowała przyspieszenie postępu technologicznego. Pozostając wciąż przy temacie modelowania, ale jednak zmieniając nieco perspektywę, chciałabym przedstawić teraz, w jaki sposób współcześnie wyobrażenia na temat ludzkiego ciała i zdolności intelektualnych przekładane są na narzędzia technologiczne w postaci cyfrowych systemów oraz jakie znaczenie w tym procesie odgrywają narracje społeczne tworzone na jego temat.

W historycznym procesie udoskonalania maszyn obliczeniowych, wyraźna jest pewna ciągłość ideologiczna, polegająca na przekonaniu, że ludzkie możliwości intelektualne

---

<sup>188</sup> Zob. stronę internetową projektu *The Algorithmic Justice League*, online: <https://www.ajl.org>, dostęp: 06.11.2023.

i poznawcze mogą zostać zrekonstruowane w postaci technologicznej. Takie było założenie Turka, choć był jedynie mistyfikacją. Następnie Charles Babbage, który kwestionował jego prawdziwość, ale zainspirowany nim poświęcił się stworzeniu maszyny zdolnej do wykonywania obliczeń. Jego wynalazek, nigdy nie zbudowana w pełni maszyna analityczna z 1837 roku, stanowił z kolei kamień milowy w rozwoju komputeryzacji. Widoczne jest to w sposób szczególny w notatkach dotyczących urządzenia, spisanych przez zafascynowaną nim hrabinę Adę Lovelace, gdzie zawiera się wiele porównań uniwersalnej maszyny liczącej do ludzkich możliwości.<sup>189</sup>

Kiedy na przełomie pierwszej i drugiej połowy XX wieku pierwsze komputery w postaci masywnych maszyn obliczeniowych zaczynały funkcjonować w przestrzeni społecznej, stając się tematem rozmów zarówno na gruncie naukowym, jak i w popularnym oraz medialnym obiegu informacji, od samego początku przyjęło się używać w ich kontekście określeń, będących analogiami do ludzkiego mózgu. Prasa i telewizja, rozpowszechniły określenie „maszyna myśląca”, używane w kontekście komputerów.<sup>190</sup> Przeniknęło ono również do literatury, czego przykład stanowi słynny cykl powieści *Diuna* Franka Herberta, którego pierwszy tom został wydany w 1965 roku. W świecie przedstawionym przez tego autora, w dalekiej przyszłości, w której ludzkość rozsiana jest po całej galaktyce po trwającej niemal wiek wojnie pomiędzy ludźmi a, jak to określił autor, myślącymi maszynami, istnieje zakaz tworzenia technologii na podobieństwo ludzkiego umysłu. Jest to futurystyczna wizja świata po pokonaniu sztucznej inteligencji przez ludzkość.

Z drugiej strony, na gruncie naukowym określenie maszyna myśląca było równie powszechne. Alan Turing, uznawany za ojca informatyki i badań nad sztuczną inteligencją, otworzył swój artykuł *Maszyna licząca a inteligencja* z 1950 roku prowokacyjnym pytaniem

---

<sup>189</sup> B.W. Biriukow, J.S. Geller, *Cybernetyka w naukach humanistycznych*, tłum. J. Sarna, Wydawnictwo Ossolineum, Warszawa, Wrocław, Kraków, Gdańsk, Łódź 1983, s. 149-150. Notatki hrabiny Lovelace zostały również wydane jako dodatkowa część wzbogacająca publikację *Faster Than Thought. A Symposium on Digital Computing Machines*, pod redakcją barona Bertrama Viviana Bowdena z 1953 roku. Ada Lovelace zawarła w nich m.in. swój własny algorytm napisany w celu wykonania na maszynie analitycznej. Z tej przyczyny niekiedy uznaje się ją za pierwszą programistkę. Zob. A.L. Philips, *Crowdsourcing Gender Equity. Ada Lovelace Day, and its companion website, aims to raise the profile of women in science and technology*, 04.10.2016, online: <http://web.archive.org/web/20161004185338/https://www.americanscientist.org/issues/pub/crowdsourcing-gender-equity> dostęp 06.11.2023.

<sup>190</sup> W odniesieniu do materiałów transmitowanych za pomocą telewizji zob. *The Thinking Machine (Artificial Intelligence in the 1960s)*, 31.05.2010, online: <https://www.youtube.com/watch?v=aygSMgK3BEM>, dostęp: 06.11.2023.

o to, czy maszyny myślą.<sup>191</sup> W dalszej części tekstu naukowiec wskazywał na bezpodstawność tego pytania i jego emocjonalny charakter. W ramach alternatywnego podejścia do problemu ujmowania statusu możliwości intelektualnych technologii cyfrowych przedstawił swój projekt testu, który nazwany został jego nazwiskiem jako „test Turinga”. Test polegał na tym, że oceniający, będący człowiekiem miał odróżnić działanie programu komputerowego od działania człowieka. Jeśli sędzia nie był w stanie tego rozstrzygnąć, program przechodził test pozytywnie. Koncepcja tego rodzaju testu miała ogromny wpływ na dalsze pojmowanie inteligencji maszynowej, szczególnie dla nauk kognitywnych. Przede wszystkim przyczyniła się ona do rozpowszechnienia czysto funkcjonalnego podejścia do inteligencji i procesów myślowych oraz poznawczych, zgodnie z którym to czy program lub urządzenie zostanie uznane za inteligentne warunkował sposób jego działania, a nie zaś sama jego budowa, a więc fizyczne podobieństwo do postaci ludzkiej. Większe znaczenie niż mechanizm działania miał tu efekt, który pozwalało ono osiągnąć.

Kolejną ważną publikacją, która umacniała porównanie cyfrowych procesów obliczeniowych do myślenia była wydana w 1953 roku *Faster Than Thought. A Symposium on Digital Computing Machines*,<sup>192</sup> której sam tytuł porównywał technologie obliczeniowe do ludzkich zdolności i to w sposób wartościujący, implikujący rywalizację. Książka ta zbierała w sobie teksty ważnych naukowców i informatyków pracujących nad budową pierwszych komputerów oraz teorii ich dotyczących. Opublikowany w niej został m.in. tekst Turinga,<sup>193</sup> w którym zaprezentował on wyniki swoich prac nad oprogramowaniem służącym do rozgrywek szachowych o nazwie Turochamp.<sup>194</sup> W jeszcze późniejszej publikacji autorstwa Pierre’a de Latila, zatytułowanej *Thinking by Machine. A Study of Cybernetics* z 1957 roku, zawarta została refleksja nad tendencją do zestawiania ze sobą komputerów i ludzkiego mózgu. Autor między innymi podkreślał, że wykorzystywanie określenia „pamięć” w kontekście zdolności komputerów do przechowywania danych i obliczeń w celu późniejszego ich wykorzystywania jest dość myląca, szczególnie dla osób bez technicznych kwalifikacji. Słowo to zdaje się sugerować, że proces, do którego się

---

<sup>191</sup> A.M. Turing, *Computing Machinery and Intelligence*, „Mind”, 1950, nr 236. Artykuł przetłumaczony i wydany w języku polskim: A.M. Turing, *Maszyna licząca a inteligencja*, tłum. M. Szczubiałka, w: *Filozofia umysłu*, red. B. Chwedeńczuk, Fundacja ALETHEIA — Wydawnictwo Spacja, Warszawa 1995, s. 271-300.

<sup>192</sup> *Faster Than Thought. A Symposium on Digital Computing Machines*, red. B.V. Bowden, Sir Isaac Pitman & Sons, London 1953.

<sup>193</sup> A.M. Turing, *Digital Computers Applied to Games*, w: *Faster Than Thought...*, dz. cyt., s. 286-311.

<sup>194</sup> Oprogramowanie to przeszło pozytywnie test Turinga, sprawdzający jego zdolność do imitowania ludzkich zachowań.

odnosi jest o wiele bardziej skomplikowany i metafizyczny, niż to w istocie ma miejsce. Badacz wskazywał, że wystarczyłoby używać terminu „rejestr”, który znacznie precyzyjniej odzwierciedla działanie komputera. Był on zwolennikiem tworzenia nowej terminologii dla zjawisk komputerowych, a nie budowania metaforycznych odniesień, które w dalszej konsekwencji prowadzą do mylnych interpretacji.<sup>195</sup>

Od początku powstawania technologii komputerowych, używano w ich kontekście języka charakterystycznego dla opisu ludzkiego mózgu. Jednak to dopiero kognitywistyka jasno sprecyzowała cel swoich badań jako stworzenie funkcjonalnych modeli działania mózgu człowieka, które następnie mogły być implementowane do postaci cyfrowej. Ta interdyscyplinarna dziedzina, której główne tezy wyprowadzone zostały z obszaru cybernetyki, powstała w wyniku połączenia wiedzy wypracowanej na gruncie takich dyscyplin jak: językoznawstwo, neurobiologia, antropologia, psychologia, filozofia oraz informatyka (w szczególności badania nad sztuczną inteligencją). Za pierwszy moment konceptualizacji kognitywistyki uznaje się 1956 rok, kiedy zorganizowana została konferencja na MIT, podczas której Allen Newell wraz z Herbertem Simonem przedstawili swoje założenia dotyczące symulacji ludzkich funkcji poznawczych za pomocą komputera.<sup>196</sup> Jako odrębna nauka ukształtowała się ona później, bo w 1976 roku, kiedy zaczęto wydawać kwartalnik „Cognitive Science”, którego pierwsze egzemplarze zawierały program badawczy sformułowany przez Newella i Simona w artykule *Informatyka jako badania empiryczne*.<sup>197</sup>

Istotny wpływ na kształtowanie się badań kognitywistycznych miały postulaty jednego z pionierów cybernetyki, Williama Rossa Ashby’ego, który krytycznie odnosił się do panującego powszechnie przekonania, że mózg jest organem służącym do myślenia, a więc próby jego technologicznej symulacji powinny polegać na stworzeniu tak zwanej maszyny myślącej. Badacz dystansował się od ujmowania cyfrowych procesów obliczeniowych w kategoriach myślenia, ponieważ jest to wartość nie dająca się w jednoznaczny sposób zmierzyć i udowodnić. Zaproponował on inne ujęcie, bliskie opisanemu wyżej podejściu Turinga, zgodnie z którym mózg nie jest maszyną myślącą, ale działającą, wykonującą akcje. Przyjmuje on informacje o otoczeniu, by następnie na ich podstawie podejmować decyzje o działaniu. Ten proces, w przeciwieństwie do myślenia,

---

<sup>195</sup> P. de Latil, *Thinking by Machine. A Study of Cybernetics*, The Riverside Press, Cambridge, Mass. 1957, s. 264-267.

<sup>196</sup> G.A. Miller, *The cognitive revolution: a historical perspective*, „Trends in Cognitive Science” 2003, tom 7, nr. 4 s. 142-143.

<sup>197</sup> A. Newell, H.A. Simon, *Computer Science as Empirical Inquiry: Symbols and Search*, „Communications of the ACM” 1976, tom 19, nr 3, s. 113-126.

daje się w przejrzysty sposób przekształcić w model, który może zostać zaimplementowany do postaci technologicznej.<sup>198</sup> Zastąpienie określenia „myślenie” terminem „inteligencja” w kontekście maszyn cyfrowych, a następnie zdefiniowanie rozumienia tego drugiego stało się kluczowym procesem dla tworzenia modeli kognitywnych.

Jednym z klasycznych podejść w naukach kognitywnych, które dobrze oddaje przekonania fundatorów tej dyscypliny jest komputacjonizm. Reprezentuje ono pogląd, zgodnie z którym ludzkie procesy umysłowe mają charakter obliczeniowy. Wpływ na kształtowanie się też komputacjonistycznych miał przywoływany już kontekst historyczny połowy XX wieku, a więc pierwsze komputery i rozwój badań nad sztuczną inteligencją, szczególnie idea maszyn Turinga, która reprezentowała abstrakcyjny model urządzenia, służącego do wykonywania algorytmów. Relacja pomiędzy wyobrażeniami o ludzkim mózgu a projektowaniem pierwszych komputerów stanowi sprzężenie zwrotne, które wyraża się w tym, że działanie maszyn obliczeniowych miało imitować możliwości ludzkiego umysłu, a następnie to technologia cyfrowa stała się narzędziem do modelowania jego działania. Komputacjonizm zdawał się dostarczać rozwiązań oraz metod badawczych dla tego, co do tej pory pozostawało zagadkowe i niezgłębione, a więc dla działania ludzkiego umysłu. Równolegle rozwijającym się w stosunku do komputacjonizmu nurtem kognitywistyki był koneksjonizm, który opierał się na badaniach z dziedziny neuropsychologii. Wykraczał on poza sprowadzanie mózgu do maszyny algorytmicznej. Nie podważał obliczeniowej natury procesów umysłowych, jednak wskazywał na to, że klasyczne architektury matematyczne nie są przystające do biologicznej natury ludzkiego mózgu. Zamiast tego proponował on myślenie o mózgu jako o sieci neuronów. Każdy neuron miał być stosunkowo prostym elementem, odpowiedzialnym za poznawcze procesy obliczeniowe. Złożoność całości układu polegała na wzajemnym oddziaływaniu neuronów w sieci. Charakterystyką takiego systemu miała być zdolność uczenia się i adaptacji do warunków zewnętrznych<sup>199</sup>. Ten nurt również wpłynął na metody projektowania systemów cyfrowych i przyczynił się do rozwoju sztucznych sieci neuronowych, będących podstawą zjawiska uczenia maszynowego. Sztuczne sieci neuronowe można porównać do działania maszyn Turinga, ale na zasadzie multiplikacji, w ogromnej skali.

Interesujący w kontekście podejścia kognitywnego (zarówno komputacjonizmu, jak i koneksjonizmu) jako sposobu reprezentacji ludzkiego poznania za pomocą narzędzi

---

<sup>198</sup> W.R. Ashby, *Design for a Brain*, „Electronic Engineering” 1948, nr 20, s. 379–383, cyt. za: A. Pickering, *The Cybernetic Brain...*, dz. cyt. s. 6.

<sup>199</sup> M. Miłkowski, *Kognitywistyka i modele umysłu*, „Dwutygodnik.com” 2016, nr 176, online: <https://www.dwutygodnik.com/artukul/6322-kognitywistyka-i-modele-umyslu.html>, dostęp: 06.11.2023.

technologicznych, jest opisywany w historycznej części wprowadzającej wątek związany z mechanicyzmem. Mechanicyzm również opierał się na konstruowaniu technologicznych automatów imitujących funkcjonowanie ludzkiego organizmu, które z jednej strony pozwalały na lepsze zrozumienie jego działania, a z drugiej dostarczały sposobów dla reprezentacji i rozpowszechniania zdobytej wiedzy. Zasadniczą różnicą między mechanicyzmem a kognitywizmem jest to, że pierwszy nurt opierał się przede wszystkim na zasadach fizyki, z kolei drugi w reprezentował podejście interdyscyplinarne, opierające się w głównej mierze na matematyce, logice, informatyce i neuropsychologii. Inną, wartą zaznaczenia różnicą jest również kierunek wytwarzanej wiedzy w kontekście relacji nauki i technologii w obu nurtach. W mechanicyzmie była to wiedza z dziedziny mechaniki, która dawała się przekładać na postęp technologiczny, a następnie na sposób myślenia o ciele i metodach reprezentacji tej wiedzy. Natomiast w przypadku kognitywizmu, relacja pomiędzy nauką a technologią jest znacznie bardziej skomplikowana i wzajemnie na siebie oddziałująca. Z jednej strony to właśnie komputacyjnej teorii umysłu zawdzięcza się chociażby rozwój programów szachowych i sztucznej inteligencji, a z drugiej rozwinięty później nurt kognitywizmu zakładał, że komputer może stanowić skuteczne narzędzie dla reprezentacji i wyjaśniania działania umysłu. Charakterystyką modeli kognitywnych jest to, że starają się one dokonywać symulacji procesów poznawczych, intelektualnych, emocjonalnych i umysłowych. Traktują one narzędzia cyfrowe jako odpowiednie środowisko do przeprowadzania tego rodzaju prób.

Doskonałym przykładem realizacji kognitywnego modelowania procesów intelektualnych na gruncie technologicznym są projekty programów służących do rozgrywek szachowych. Jednym z najsłynniejszych jest Deep Blue, stworzony z ramienia firmy IBM, który w 1997 roku pokonał ówczesnego mistrza szachowego, Garriego Kasparowa. Deep Blue był programem, którego można uznawać za odzwierciedlenie komputacjonistycznego ujęcia inteligencji. Był on maszyną algorytmiczną operującą olbrzymim zasobem zmiennych i reguł, które powstały m.in. na podstawie analizy rozegranych już partii szachowych innych mistrzów szachowych. Posiadał on obszerną bibliotekę możliwych ruchów, a także algorytmy służące do interpretacji obecnego ułożenia na planszy i podejmowania decyzji, dotyczącej tego, jaki ruch wykonać. Był to jednak określony, raz zdefiniowany system, który potrafił wykonać bardzo wiele wariantów scenariuszy rozgrywki szachowej, nie posiadał on, tak jak późniejsze programy, zdolności do uczenia się czy generowania nowych ruchów. Ta umiejętność była specyfiką systemów, które realizowały koneksjonistyczne ujęcie inteligencji i opierały się na sztucznych sieciach

neuronowych. Zwycięstwo algorytmicznej maszyny Deep Blue nad Kasparowem było możliwe dzięki jasno określonym zasadom gry, w tym ustalonej specyfice ruchu każdego piona. Wobec tego, firma Deep Mind, będąca własnością Google, podjęła wyzwanie, aby stworzyć program, który potrafiłby pokonać człowieka w grze abstrakcyjnej, nie tak jasno ograniczonej zasadami, jak szachy, którą była starochińska gra planszowa Go. Aby osiągnąć ten cel, twórcy stworzyli program AlphaGo wykorzystujący system sztucznych sieci neuronowych. Proces jego kreacji i doskonalenia opierał się przede wszystkim na treningu, polegającym na graniu przeciwko samemu sobie lub przetwarzaniu przebiegu rozgrywek mistrzów, powtarzanym wielokrotnie po to, aby stworzyć bazę wiedzy zwycięskich i przegranych zagrań.<sup>200</sup> Tak wyszkolona sieć neuronowa pokonała w 2016 roku mistrza świata w grze Go, Lee Sedola w pojedynku składającym się z pięciu partii, z wynikiem cztery do jednego (Il. 5).<sup>201</sup>



Il. 5. Kadr z filmu *AlphaGo* (2017). Scena przedstawia pojedynek Lee Sedola z programem AlphaGo.  
Źródło: *AlphaGo*, reż. Greg Kohs, 2017.

Problemem kognitywnych modeli aplikowanych do postaci cyfrowych technologii jest kwestia nieprzejrzystości ich działania. Modele kognitywne wytwarzane na polu naukowym są jawne, rozpowszechnione w postaci publikacji.<sup>202</sup> Jednak opierające się na nich systemy

<sup>200</sup> A. Rogers, *What Deep Blue And AlphaGo Can Teach Us About Explainable AI*, 09.05.2019, online: <https://www.forbes.com/sites/forbestechcouncil/2019/05/09/what-deep-blue-and-alphago-can-teach-us-about-explainable-ai/?sh=7c78a8b952fd>, dostęp: 06.11.2023.

<sup>201</sup> Powstał film dokumentalny przedstawiający to wydarzenie: *AlphaGo*, reż. Greg Kohs, 2017, USA.

<sup>202</sup> Zob. Opis modelu kognitywnego przedstawiającego działanie mechanizmu uwagi i znaczenie emocji w tym procesie: J.G. Taylor, N.F. Fragopanagos, *The interaction of attention and emotion*. „Neural Networks” 2005, nr 18(4), 353–369 lub inny przykład, przedstawiający działanie procesów poznawczych i emocjonalnych

cyfrowe tracą swoją transparentność głównie z przyczyn własnościowych. Oprogramowanie jako wartość intelektualna jest własnością firm, które je wytwarzają. Z tego powodu nie jest ono upubliczniane. Ani IBM ani Google nie ujawniły szczegółowego działania programów Deep Blue oraz AlphaGo. Podobnie jak w przypadku mechanicznego Turka, biorący udział w rozgrywce i obserwujący ją mieli jedynie możliwość analizowania i interpretacji działania technologii. Istota jej funkcjonowania pozostaje tajemnicą. Czarnoskrzynkowy charakter oprogramowania, może nie budzić zastrzeżeń w kontekście programów o charakterze eksperymentalnym, których celem jest rozgrywanie gier. Jednak nabiera on znaczenia w przypadku, gdy sprawa dotyczy systemów o charakterze społecznym, jak np. bankowe systemy przyznawania kredytów czy programy walidacji wydajności pracowniczej, takie jak omawiany w poprzedniej części rozdziału IMPACT. Przejrzystość i zrozumiałość działania systemów opartych na sztucznej inteligencji powinna posiadać taki sam priorytet, jak ich funkcjonalność.

Innym problemem modeli kognitywnych, będących podstawą rozwiązań technologicznych związanych z dziedziną sztucznej inteligencji, jest sprawa ich nieprzystawalności do rzeczywistości. Modele kognitywne zostały skrytykowane za to, że nie stanowią rzetelnego działania ludzkiego umysłu, są jedynie pewną reprezentacją, co też wynika z wcześniej opisywanej przeze mnie natury modelu. Roger Penrose w książce *Nowy umysł cesarza. O komputerach, umyśle i prawach fizyki* z 1989 argumentował na rzecz tego, że ludzka świadomość ma charakter niealgorytmiczny. Dlatego też modele w postaci Turingowskiej maszyny nie mogą sprostać zadaniu jej reprezentacji. Podejście badacza stanowi krytykę czysto komputacjonistycznego ujęcia umysłu. Z kolei Nicholas L. Cassimatis wskazywał na to, że zarówno sztuczna inteligencja, jak i modelowanie kognitywne posiadają wspólny problem, którym jest upraszczanie, spłykanie kategorii inteligencji na rzecz tworzenia poddanych redukcji, funkcjonalnych modeli.<sup>203</sup> Jako rozwiązanie tej sytuacji badacz zaproponował powołanie odrębnej dziedziny wewnątrz kognitywistyki, która byłaby nauka o inteligencji (*intelligence science*). Jej celem miałyby być przywrócenie pogłębionych badań w dziedzinie ludzkich możliwości poznawczych i umysłowych.

---

za pomocą systemu sieci neuronowej: A. Khashman, *Modeling cognitive and emotional processes: A novel neural network architecture*, „Neural Networks” 2010, nr 23(10), s. 1155-1163.

<sup>203</sup> N.L. Cassimatis, *Artificial Intelligence and Cognitive Modeling Have the Same Problem*, w: *Theoretical Foundations of Artificial General Intelligence*, red. P. Wang, B. Goertzel, Atlantis Press, Amsterdam 2012, s. 11-24.

Zarówno problem nieprzejrzystości działania oprogramowania, jak i kwestia nieadekwatności do świata rzeczywistego poddanych redukcji modeli kognitywnych, sprowadza się do kategorii wytwarzania narracji na gruncie nauki i technologii. Bruno Latour, aby zaznaczyć bliski związek nauki z technologią korzystał z pojęcia technonauki. Badacz podkreślał, że obie te dziedziny przenikają się wzajemnie w procesie wytwarzania wiedzy.<sup>204</sup> Oznacza to, że zarówno nauka, jak i technologia działają w podobny sposób, który opiera się na kreowaniu narzędzi dla opisu i rozumienia zjawisk rzeczywistych. Zgodnie z myślą Latoura, proces ten nie opiera się na klasycznie rozumianej reprezentacji – prostym odniesieniu modelu do rzeczywistości – ale na zjawisku krążącej referencji<sup>205</sup>. Kategorię krążącej referencji można wyjaśnić w taki sposób, że analizy naukowców nie dotyczą przedmiotu ich badań, ale formułowane są na podstawie wielokrotnie zmodyfikowanej reprezentacji przedmiotu badań, zapośredniczonej technologicznie lub językowo. Proces badawczy, tu Latour odnosił się przede wszystkim do działań laboratoryjnych, polega na syntezie i przekształcaniu przedmiotu badań, tworzeniu kolejnych jego reprezentacji, które pozwalają się ujmować za pomocą narzędzi analizy. W efekcie końcowym, tym co łączy człowieka i jego wiedzę o świecie ze światem rzeczywistym jest łańcuch reprezentacji, który Latour określał mianem krążącej referencji. Istotną właściwością tego procesu jak fakt, że powstające reprezentacje nie muszą mieć charakteru językowego, jak na przykład terminy, definicje, pojęcia, teorie, ale mogą przybierać formę obiektów, jak na przykład fotografie, wydruki, nagrania czy inne. Tworzenie elementów łańcucha referencji może mieć charakter językowy jak i technologiczny. Na przykładzie omawianych w tej części rozdziału modeli kognitywnych daje się zaobserwować jak nauka i technologia łączą się w procesie wytwarzania wiedzy na temat działania ludzkiego umysłu. Oprogramowanie natomiast stało się referencją dla mówienia o procesach intelektualnych.

### **2.2.3 Perspektywa trzecia. Historyjki użytkownika – znaczenie języka w procesie wytwarzania oprogramowania**

---

<sup>204</sup> B. Latour, *Science In Action. How to Follow Scientists and Engineers Through Society*, Harvard University Press, Cambridge, Mass. 2015, s. 145–176.

<sup>205</sup> B. Latour, *Pandora's Hope: Essays on the Reality of Science Studies*, Harvard University Press, Cambridge, Mass., Londyn 1999, s. 24–79.

Do tej pory omówiłam dwie perspektywy przedstawiające możliwe ujmowanie zjawisk związanych z oprogramowaniem w ujęciu społecznym. Poruszały one problematykę modelowania zapośredniczonego przez technikę oraz zależności pomiędzy nauką a technologią w procesie wytwarzania wiedzy na przykładzie wybranych klasycznych podejść kognitywistycznych. W niniejszej części pracy chciałabym zawrzeć trzecią perspektywę, która podobnie jak poprzednie będzie sytuować się po stronie wytwarzania oprogramowania, ale tym razem skupiać się będzie na znaczeniu języka naturalnego w tym procesie. Charakterystyczne dla współczesnej kultury jest to, że technologia cyfrowa odgrywa znaczącą rolę w niemal każdej dziedzinie życia społecznego. Lev Manovich określając cechy nowych mediów, podkreślał, że współcześnie trudno znaleźć przedmiot fizyczny, który w procesie wytwarzania nie miałby styczności z komputerem.<sup>206</sup> Wytwory człowieka, nawet jeśli same nie posiadają w sobie oprogramowania, produkowane i dystrybuowane są za pomocą systemów cyfrowych. W późniejszej publikacji badacz opisywał oprogramowanie jako niewidzialne spoiwo rzeczywistości.<sup>207</sup> Znaczenie, jakie posiada software na poziomie społecznym w obszarze komunikacji, zarządzania i podejmowania decyzji, które można określić mianem cyfrowego determinizmu współczesnej kultury, owocuje zyskującymi rozgłos i popularność przekonaniem, że to algorytmy rządzą światem<sup>208</sup>, sprawują kontrolę nad ludźmi<sup>209</sup>. Prawdą jest, że oprogramowanie odgrywa współcześnie zasadniczą rolę w większości procesów ekonomicznych, politycznych, społecznych i kulturowych, jednak sformułowanie głoszące, że to algorytmy rządzą światem stanowi zbyt duże uproszczenie, ponieważ w procesie tym odgrywa rolę o wiele więcej czynników. W tej części rozdziału chciałabym pogłębić temat dominacji algorytmicznej poprzez przedstawienie i omówienie społecznego aspektu wytwarzania oprogramowania. W głównej mierze chciałabym skupić się na nadrzędnym względem software’u czynniku, którym jest język naturalny.

Specyfiką oprogramowania jest to, że proces jego wytwarzania nie poddaje się prostej manufakturyzacji. Wytwarzania software’u nie można porównać do produkcji technologii nie wykorzystującej oprogramowanie, dla których możliwe jest opracowanie i wdrożenie optymalnej metody, pozwalającej na tworzenie setek poprawnie działających urządzeń, które spełniają oczekiwania klientów i użytkowników końcowych. Każde oprogramowanie

---

<sup>206</sup> L. Manovich, *Język Nowych Mediów...*, dz. cyt., s. 92-94.

<sup>207</sup> L. Manovich, *Software Takes Command...*, dz. cyt., s. 8.

<sup>208</sup> S. Konkol, *Algorytmy rządzą światem?*, 24.10.2016, online: <https://blog.twistersolve.pl/2016/10/algorytmy-rzadza-swiatem/>, dostęp 06.11.23.

<sup>209</sup> *Rządzą nami algorytmy, a wiemy o nich niewiele*, 06.02.2018, online: <https://p.dw.com/p/3Cqf7>, dostęp 06.11.23.

i, co za tym idzie, metoda jego produkcji, musi zostać dostosowane do specyfiki jego zastosowania. Współcześnie najpopularniejszymi metodami wytwarzania oprogramowania są tak zwane metody zwinne (*Agile*)<sup>210</sup>. Opierają się one przede wszystkim na iteracjach i przyrostowym rozwoju produktu. Zwinne metodyki wytwarzania oprogramowania rozwinęły się w kontrze do tradycyjnych metod kaskadowych (*Waterfall*), które zakładały dokładnie zaplanowane etapy prac nad programem, a następnie ich realizację krok po kroku. Do tych etapów należały: planowanie, analiza, projekt, implementacja, testowanie oraz wdrożenie.<sup>211</sup> Przejście do kolejnego etapu było możliwe jedynie w przypadku, gdy poprzedni został zamknięty. Z kolei, jeśli określony etap został zakończony, nie było dopuszczalne, aby do niego powrócić lub coś w nim zmienić. Zasady kaskadowej metody wytwarzania oprogramowania zostały po raz pierwszy opisane w roku 1970 w artykule *Managing the Development of Large Software Systems* autorstwa Winstona W. Royce'a.<sup>212</sup> Model kaskadowy wymagał od zespołu z niego korzystającego, aby wymagania dotyczące wytwarzanego produktu (programu, aplikacji) zostały jasno sprecyzowane na samym początku procesu. *Waterfall* nie pozwalał na zmianę koncepcji po rozpoczęciu prac nad oprogramowaniem. Mocnymi stronami tej metodyki była skuteczna estymacja procesu produkcyjnego, łatwość rozgraniczania poszczególnych etapów od siebie oraz pewność końcowego kształtu aplikacji lub programu. Jednak model kaskadowy okazywał się niewystarczający, kiedy potrzebne było elastyczne podejście do problemów. W przypadku, kiedy zmiana założeń okazywała się konieczna, praca nad produktem musiała być rozpoczęta od samych podstaw, co wiązało się z ogromnymi stratami finansowymi.

Zwinne metodyki wytwarzania oprogramowania stanowiły alternatywną propozycję wobec utrwalonych już w przemyśle informatycznym metod kaskadowych. Znaczącym momentem ich rozpowszechnienia był początek XXI wieku, kiedy to w dniach 11-13 lutego

---

<sup>210</sup> Przywoływanie anglojęzycznych odpowiedników terminów ma dla mnie szczególne znaczenie w tym podrozdziale, ponieważ język angielski w przemyśle informatycznym stanowi podstawowy język komunikacji i nazewnictwa, który umożliwia terminologiczną i komunikacyjną uniwersalność w tej branży. Terminologia w języku angielskim jest powszechnie stosowana nawet w polskojęzycznych zespołach. Przywołując polskie odpowiedniki w tym podrozdziale nie korzystałam z własnych tłumaczeń, ale z tych zaproponowanych przez oficjalne Stowarzyszenie Jakości Systemów Informatycznych, zajmujące się m.in. translacją na język polski sylabusów International Software Testing Qualification Board, która jest organizacją wydającą certyfikaty testerskie.

<sup>211</sup> Etapy wytwarzania mogły być określane inaczej w zależności od decyzji osób odpowiedzialnych za nadzór produkcji oprogramowania. Na przykład firma IBM na podstawie doświadczeń z realizacji własnych projektów wypracowała własny model kaskadowy MITP (*Managing the Implementation of the Total Project*) i składał się z czterech faz: wymagania, ustanowienie projektu, realizacja projektu, ukończenie projektu.

<sup>212</sup> W.W. Royce, *Managing the development of large software systems: concepts and techniques* „IEEE CS Press” 1970, s. 328-339; dostępny online: <https://web.archive.org/web/20160318002949/http://www.cs.umd.edu/class/spring2003/cmssc838p/Process/waterfall.pdf>, dostęp: 06.11.2023.

2001 roku w USA zebrali się reprezentanci, opracowujący nowe metodologie wytwarzania oprogramowania. Siedemnastu zgromadzonych mężczyzn sformułowało i podpisało wtedy manifest zwinnego wytwarzania oprogramowania (*Agile Manifesto*).<sup>213</sup> Podstawowym przesłaniem manifestu było ponowne wyznaczenie kluczowych wartości procesu produkcji oprogramowania. Zestawione one zostały z tradycyjnym podejściem w następujący sposób:

W wyniku naszej pracy, zaczęliśmy bardziej cenić:

**Ludzi i interakcje** od procesów i narzędzi

**Działające oprogramowanie** od szczegółowej dokumentacji

**Współpracę z klientem** od negocjacji umów

**Reagowanie na zmiany** od realizacji założonego planu.

Oznacza to, że elementy wypisane po prawej są wartościowe, ale większą wartość mają dla nas te, które wypisano po lewej.<sup>214</sup>

Z przywołanego tu fragmentu manifestu daje się odczytać, że w podejściu *Agile* ceni się elastyczne podejście do zakładanych planów, koncepcji i wyobrażeń na temat wytwarzanego oprogramowania. Kluczowe znaczenie w procesie produkcji mają członkowie zespołów tworzących software i komunikacja między nimi. Bardziej szczegółowe założenia tej metodyki zostały wyrażone w drugiej części manifestu, którą stanowi dwanaście zasad:

1. Najwyższy priorytet ma dla nas zadowolenie klienta dzięki wczesnemu i ciągłemu wdrażaniu wartościowego oprogramowania.
2. Bądźcie gotowi na zmiany wymagań nawet na późnym etapie jego rozwoju. Procesy zwinne wykorzystują zmiany dla zapewnienia klientowi konkurencyjności.
3. Dostarczajcie funkcjonujące oprogramowanie często, w kilkutygodniowych lub kilkumiesięcznych odstępach. Im częściej, tym lepiej.
4. Zespoły biznesowe i deweloperskie muszą ściśle ze sobą współpracować w codziennej pracy przez cały czas trwania projektu.
5. Twórzcie projekty wokół zmotywowanych ludzi. Zapewnijcie im potrzebne środowisko oraz wsparcie i zaufajcie, że wykonają powierzone zadanie.
6. Najbardziej efektywnym i wydajnym sposobem przekazywania informacji zespołowi deweloperskiemu i wewnątrz niego jest rozmowa twarzą w twarz.
7. Działające oprogramowanie jest podstawową miarą postępu.

---

<sup>213</sup> *Manifest Agile* jest dostępny online: <https://agilemanifesto.org>, dostęp: 06.11.23.

<sup>214</sup> Pogrubienie dodane przeze mnie. W manifestie zaznaczenie zostało wyrażone większym rozmiarem czcionki. Zob. <https://agilemanifesto.org/iso/pl/manifesto.html>, dostęp: 06.11.2023.

8. Procesy zwinne umożliwiają zrównoważony rozwój. Sponsorzy, deweloperzy oraz użytkownicy powinni być w stanie utrzymywać równe tempo pracy.
9. Ciągłe skupienie na technicznej doskonałości i dobrym projektowaniu zwiększa zwinność.
10. Prostota – sztuka minimalizowania ilości koniecznej pracy – jest kluczowa.
11. Najlepsze rozwiązania architektoniczne, wymagania i projekty pochodzą od samoorganizujących się zespołów.
12. W regularnych odstępach czasu zespół analizuje możliwości poprawy swojej wydajności, następnie dostraja i dostosowuje swoje działania do wyciągniętych wniosków.<sup>215</sup>

Pierwsze dwie zasady podkreślały potrzebę komunikacji z klientem, a więc z osobą, która zamawia oprogramowanie i określa swoje potrzeby i wyobrażenia, które następnie przekładane są na wymagania projektowe. Z manifestu da się odczytać, że komunikacja ta odbywa się w sposób ciągły, w trakcie całego trwania procesu produkcji i klient w każdej chwili może zmienić wyobrażenie na temat wizji zamawianego produktu. Trzecia zasada odnosiła się do czasu trwania poszczególnych etapów produkcji. W modelu kaskadowym poszczególne fazy mogły trwać bardzo długo, nawet rozciągać się do lat. Metody zwinne, w zależności od konkretnych podejść, dzielą się na krótkie etapy, trwające od tygodnia do kilku miesięcy. Punkt czwarty, piąty, szósty, ósmy, jedenasty i dwunasty manifestu odnosi się do komunikacji wewnątrz zespołów projektowych, podkreśla istotę częstej wymiany informacji, samoorganizacji i przede wszystkim – wspólnego wyciągania wniosków z prowadzonych działań. Podejście zwinne zakłada, że najważniejszym elementem w procesie wytwarzania oprogramowania są ludzie wchodzący w skład zespołów projektowych. *Agile* nie określa w sposób ścisły, jak powinien działać zespół, aby osiągnąć sukces, ale zakłada, że sprawna komunikacja między jego członkami pozwoli na wypracowanie indywidualnych metod i rozwiązań, które pozwolą na wyciągnięcie wniosków z własnych działań i skuteczną optymalizację procesu produkcyjnego.

Manifest zawierał w sobie idee i główne cele będące podstawą dla wdrażania innego podejścia niż kaskadowe w przemyśle informatycznym. Nie zostały w nim opisane techniczne rozwiązania typowe dla metod zwinnych, które umożliwiały wprowadzanie tak dużej elastyczności względem zmieniających się wymagań. Metodyki zwinne zawdzięczają łatwość dostosowywania się iteracyjnym i przyrostowym modelom produkcji. Model iteracyjny zakłada udoskonalanie wytwarzanego produktu w cyklach o określonym czasie trwania. W każdym cyklu opracowana zostaje określona funkcjonalność programu.

---

<sup>215</sup> Zob. *Założenia Manifestu Programowania Zwinnego*, <https://agilemanifesto.org/iso/pl/principles.html>, dostęp 06.11.2023.

Podejście to umożliwia poprawianie już dostarczonych modułów oraz przerwanie prac nad aplikacją w momencie, gdy zostanie uznana za spełniającą wymagania. Model przyrostowy cechuje się stopniowym dodawaniem poszczególnych modułów aplikacji. Rezygnuje on z pracy nad całością na rzecz podzielenia produktu na zbiór mniejszych części, które są kolejno dostarczane.<sup>216</sup> W praktyce model iteracyjny i przyrostowy często stosowane są jednocześnie w postaci hybrydowej, dzięki której kolejne iteracje przynoszą konkretne przyrosty.

Podstawową metodą opisywania wymagań w *Agile* są historyjki użytkownika<sup>217</sup> (*user stories*), które stanowią główny punkt analizy tej części rozdziału. Historyjki użytkownika są kluczową techniką dokumentacji wymagań technicznych. Wyrażone są w języku nietechnicznym i nieformalnym, przedstawiają perspektywę końcowego użytkownika aplikacji, a ich celem jest opisanie działania poszczególnych funkcjonalności programu. Istnieje ustalona formuła, zgodnie z którą tworzy się historyjki użytkownika i posiada ona następującą postać:

Jako <użytkownik> chcę <potrzeba>, żeby <cel do osiągnięcia>.<sup>218</sup>

Każda historyjka użytkownika dostosowywana jest do powyższego schematu. Zakłada on sprecyzowanie perspektywy tego, kto mówi, czyli zdefiniowania, jaki rodzaj użytkownika, np. administrator, zwyczajny użytkownik, specjalny użytkownik itd. Następnie istotną częścią jest potrzeba, czyli zasadniczy opis wymaganej funkcjonalności np. „móc się zalogować”, „posiadać listę ulubionych kontaktów”. Ostatnim elementem historyjki jest przywołanie celu, który pozwala osadzić funkcjonalność w kontekście aplikacji, ale też lepiej zrozumieć intencję stojącą za danym wymaganiem. Przykładowa uzupełniona historyjka użytkownika może posiadać następującą postać:

Jako administrator chcę mieć możliwość generowania raportu z danymi aplikacji, żeby móc sprawnie analizować wyniki.

Za inny przykład posłużyć może:

---

<sup>216</sup> Zob. *Certyfikowany tester. Sylabus poziomu podstawowego ISTQB. Wersja 3.1*, tłum. L. Stapp, Stowarzyszenie Jakości Systemów Informatycznych 2018, s. 35.

<sup>217</sup> Przywoływane przez mnie polskie tłumaczenie tego terminu jest oficjalnym tłumaczeniem wykonanym przez Stowarzyszenie Jakości Systemów Informatycznych.

<sup>218</sup> J. Patton, *User Story Mapping*, O'Reilly, Beijing, Köln, Cambridge 2014, s. 99-100.

Jako użytkownik chcę mieć możliwość zwinienia bocznego panelu menu, aby łatwiej mi było przeglądać treść strony.

Za pomocą tego schematu uchwycona zostaje zasadnicza treść historyjki użytkownika. Ta formuła stanowi jej kluczowy element, choć nie jedyny, ponieważ może zostać rozszerzona o szczegółowe kryteria niefunkcjonalne i kryteria akceptacji, które później stają się podstawą odniesienia w procesie testowym.<sup>219</sup> Innymi słowy treść historyjki użytkownika mają charakter nadrzędnego wymagania, do którego dostosowują swoją pracę programiści. W dalszej kolejności wytworzone przez nich oprogramowanie jest sprawdzane przez odrębny zespół testerski w celu weryfikacji, czy założenia zawarte w historyjce zostały prawidłowo pokryte. Wszelkie niezgodności z treścią historyjki zostają uznane jako błąd aplikacji i zgłoszone w postaci defektów.<sup>220</sup> Podsumowując, oprogramowanie w procesie wytwarzania, jest całkowicie podporządkowane sformułowanej wcześniej historyjce użytkownika, która za pomocą codziennego języka (nietechnicznego), a więc takiego, jakiego używają użytkownicy, aby opisać swoje potrzeby, określa wymagania dla poszczególnych funkcjonalności aplikacji. Istnieje również metoda, aby historyjki użytkownika grupować, na podstawie podobieństw lub odniesień do jednakowych obszarów funkcjonalności, w większe całości, które określa się mianem opowieści (*epic*).<sup>221</sup> Zbiór wszystkich opowieści i zawierających się w nich historyjek użytkownika składa się na dokumentację projektu, która jest organizowana, przechowywana, utrzymywana i stanowi bazę wiedzy na temat jej prawidłowego działania.

Zanim przejdę do rozważań właściwych dla tej części, chciałabym poświęcić jeszcze uwagę istotnemu w ich kontekście procesowi powstawania historyjek użytkownika. Należą one bowiem do dokumentacji typu URS (*User Requirement Specification*). Jest to ogólne określenie dla wymagań tworzonych przez właściciela produktu (*product owner*) w oparciu o konsultacje z użytkownikami końcowymi (najczęściej będącymi osobami zlecającymi wytworzenie oprogramowania). Mają one postać przepisania na język nietechniczny, wyobrażeń użytkownika na temat działania programu. Istotne jest w tym procesie to, że spisuje je osoba, mająca później zarządzać działaniem zespołu technicznego. Odgrywa więc ona rolę mediatora pomiędzy użytkownikiem a zespołem pracującym nad powstaniem oprogramowania. W dalszej kolejności przygotowań do procesu produkcyjnego dochodzi do

---

<sup>219</sup> Zob. *Historyjka użytkownika*, w: *Słownik Terminów Testowych ISTQB. Wersja 3.4*, tłum. L. Stapp, Stowarzyszenie Jakości Systemów Informatycznych 2020, s. 30.

<sup>220</sup> Zob. *Certyfikowany tester...*, s. 15-16, 69.

<sup>221</sup> Zob. *Opowieść*, w: *Słownik Terminów Testowych...*, dz. cyt., s. 65.

następnej translacji, ponieważ język pragnienia użytkownika typowy dla URS zostaje przepisany na tak zwane FS (*Functional Specification*). FS to zbiór wymagań wyrażonych w języku technicznym, specyficznym dla każdego projektu, który opisuje to, jakie technologiczne implementacje będą musiały zostać zastosowane, aby wypełnić przedstawione w URS wymagania. FS opisuje planowane działanie systemu, jednak z założenia ma ono być zrozumiałe dla wszystkich członków zespołu i interesariuszy, dlatego spisany jest również językiem naturalnym, nie wymagającym specjalistycznej wiedzy z zakresu technicznego działania systemu, chociaż wymagającym znajomości terminologii specyficznej dla danego projektu, z którą zaznajomiony jest jego zespół. Podsumowując, niezbędnym elementem procesu wytwarzania oprogramowania jest translacja języka. Proces ten rozpoczyna się przez opowieść użytkownika na temat własnych wyobrażeń dotyczących działania systemu, zostaje on spisany na język wymagań użytkownika a następnie przekształcony do postaci specyfikacji technicznej. Ogromną rolę odgrywa tu język i procesy translacji, którym zostaje poddany, jak też fakt, że na każdym etapie produkcji oprogramowania to wymagania wyrażone w języku naturalnym są nadrzędne względem wytwarzanego software'u. W ramach dopowiedzenia należy podkreślić jeszcze, że historyjki użytkownika nie są jedyną metodą z obszaru URS, jednak należą do najbardziej popularnej strategii konstruowania wymagań w modelu *Scrum*<sup>222</sup>, który to z kolei jest najpowszechniejszą z metodyk zwinnych.<sup>223</sup> Przez wzgląd na powszechność zastosowania można uznać je za reprezentatywne dla zwinnego podejścia w wytwarzaniu oprogramowania.

### 2.3 Związek oprogramowania z językiem naturalnym

Alistair Cockburn, jeden z sygnatariuszy *Manifestu Agile*, twórca teorii dotyczących oprogramowania i metod jego wytwarzania, udowodniał, że zarówno języki programowania,

---

<sup>222</sup> *Scrum* to sposób organizacji pracy nad wytwarzaniem oprogramowania, rozwijany od 1995 roku, który stanowi obecnie najpopularniejszy model w ramach metodyk zwinnych (*Agile*). Jego nazwa odnosi się do techniki rugby, w ramach której cały zespół pracuje razem, by przesuwać piłkę w jednym kierunku. Praca w ramach modelu *Scrum* opiera się przede wszystkim na niewielkich inkrementacjach podzielonych na sprinty czasowe. Zob. definicję: *What is scrum?*, online: <https://www.scrum.org/resources/what-scrum-module>, dostęp: 20.11.2023.

<sup>223</sup> Zob. Raport na rok 2022 podsumowujący rozwój metodyki *Agile* na świecie, przygotowywany co roku przez platformę *Digital.ai*: *16th State of Agile Report*, online: <https://digital.ai/resource-center/analyst-reports/state-of-agile-report>, dostęp: 06.11.2023. Raport za rok 2023 zostanie opracowany na początku 2024 roku.

jak i język naturalny, funkcjonują w podobny sposób.<sup>224</sup> Wyjaśniał on, że język naturalny można interpretować jako rozszerzenie metafor właściwych dla programowania obiektowego. Badacz przedstawił porównanie części mowy, takich jak: rzeczownik, czasownik, przymiotnik, przysłówki oraz przypadki (mianownik, dopełniacz itd.) do kategorii obiektów i klas. W informatyce przez określenie obiekt rozumie się podstawową jednostkę strukturalną programu w programowaniu obiektowym, która zawiera w sobie dane i metody, nazywane też funkcjami, służące do wykonywania zadań na tych danych. Z kolei zadaniem klas jest definiowanie obiektów, nadawanie im cech i atrybutów. Obiekty podporządkowane danej klasie nazywane są jej instancjami. Aby zobrazować te definicje można użyć przykładu programu do zarządzania sygnalizacją świetlną. W tym przypadku obiektami byłyby samochody, a klasy mogłyby określać takie ich atrybuty jak rozmiar i ciężar samochodu czy prędkość jazdy. W odniesieniu do tych definicji Cockburn przedstawia przymiotniki jako słowa, których używa się do opisu rzeczownika, zawężenia jego znaczenia. Można powiedzieć, że każdy przymiotnik tworzy podzbiór klasy rzeczowników, swoich instancji. W tym rozumieniu rzeczownik pełniłby funkcję obiektu, w znaczeniu informatycznym. Zasadniczą tezą artykułu Cockburna jest myśl, że genezę modeli strukturalnych języków programowania można wyprowadzić z modeli funkcjonujących w języku naturalnym, wyrażonych poprzez zasady gramatyczne.

W kontekście kulturowego znaczenia samego języka, na gruncie lingwistyki, powstało wiele opracowań dotyczących tego, w jaki sposób język staje się nośnikiem treści kulturowych i służy za narzędzie, za pomocą którego ukonkretniają się społeczne wyobrażenia na temat rzeczywistości. Rozpoznania opowiadające się za determinizmem i relatywizmem językowym, zgodnie z którymi każda społeczność kreuje swój własny, intersubiektywny obraz rzeczywistości, do której dostęp ma wyłącznie przez używany język wraz z wszelkimi jego ograniczeniami, określa się mianem Językowego Obrazu Świata (JOS).<sup>225</sup> W lingwistyce to pojęcie ukonstytuowało się w XX wieku, w głównej mierze dzięki językoznawcom niemieckim (Leo Weisgerber) oraz amerykańskim (Edward Sapir, Benjamin Lee Whorf). Analizy z tego obszaru wykraczają poza badania ściśle lingwistyczne i pojawiają się również w tekstach antropologicznych, kulturoznawczych, filozoficznych, jak również w tych z obszaru lingwistyki kognitywnej. Jedną najbardziej reprezentatywnych

---

<sup>224</sup> A. Cockburn, *Using natural Language as a Metaphoric Base for OO*, „ACM SIGPLAN OOPS Messenger” 1993, nr 4(2), s. 187-189.

<sup>225</sup> Na gruncie polskich badań pojęcie to było opracowywane w ramach konferencji „Język a kultura”, która odbyła się Puławach 16-19 października 1987. Jej owocem była publikacja zbierająca teksty wystąpień pod redakcją Jerzego Bartmińskiego, zob. *Językowy obraz świata*, red. J. Bartmiński, Uniwersytet Marii Curie-Skłodowskiej, Lublin 1999.

teorii, wyrażających stanowisko tożsame dla definicji Językowego Obrazu Świata są tezy sformułowane przez Edwarda Sapira i Benjamin Lee Whorfa, które przyjęło się określać łącznie hipotezą Sapira-Whorfa.

Zgodnie z dobrze znanymi w badaniach rozpoznaniem Sapira i Whorfa, język determinuje ludzkie myślenie, sposób postrzegania rzeczywistości i jest zasadniczym miejscem kształtowania się treści kulturowych. Innymi słowy kultura realizuje się wyłącznie w kontekście języka i za jego sprawą. Pogląd ten wyrażony został w następujący sposób:

Ludzie nie żyją wyłącznie w świecie obiektywnym ani też wyłącznie w świecie działań społecznych w zwykłym rozumieniu, lecz pozostają w dużej mierze na łasce języka, który stał się środkiem ekspresji w ich społeczeństwie. Iluzją jest wyobrażenie, że przystosowujemy się do rzeczywistości w zasadzie bez użycia języka, który jest tylko niezbyt istotnym narzędziem rozwiązywania specyficznych problemów komunikacji czy refleksji. Prawda wygląda tak, że „realny świat” jest w znacznej mierze zbudowany nieświadomie na zwyczajach językowych danej grupy. Żadne dwa języki nie są nigdy dostatecznie podobne, by można je było traktować jako reprezentujące tę samą rzeczywistość społeczną. Światy w których żyją różne społeczeństwa, są odrębnymi światami, nie zaś tym samym światem, tylko opatrzonym odmiennymi etykietami.<sup>226</sup>

Hipoteza Sapira-Whorfa została również przyjęta na gruncie lingwistyki kognitywnej, ponieważ zakładała, że język wpływa na myślenie oraz w konsekwencji na zachowanie i procesy poznawcze człowieka. Whorf podkreślał nierozzerwalność myślenia i uwarunkowanego kulturowo języka w słowach: „Myślenie (...) podąża kolejnymi wyźłobionymi w danym języku, które mogą systematycznie uwydatniać jedne aspekty rzeczywistości i rozumieć, ignorować zaś inne, akceptowane z kolei przez odmienne języki.”<sup>227</sup> Zgodnie z tą tezą konstruowanie obrazu świata, jako proces kognitywny, odbywa się za pośrednictwem języka i jest charakterystyczny dla wspólnot dzielący ów język, a więc będzie on udziałem narodowości, pokoleń, społeczności, subkultur i wszelkich ugrupowań, które wypracowują własne metody komunikacji. Język naturalny w podejściu relatywistycznym jest więc w swym opisowym aspekcie reprezentacją doświadczenia poznawczego określonej grupy społecznej. Oznacza to, że to właśnie w jego obszarze konstruują się wartości kulturowe, w tym nieświadome sądy o świecie, przekonania

---

<sup>226</sup> E. Sapir, *Kultura, język, osobowość. Wybrane eseje*, tłum. R. Zimand, Państwowy Instytut Wydawniczy, Warszawa 1978, s. 88.

<sup>227</sup> B.L. Whorf, *Język, myśl i rzeczywistość*, Państwowy Instytut Wydawniczy, Warszawa 1982, s. 345.

i stereotypy.<sup>228</sup> Istotną tezą wpisaną w to podejście, jest stwierdzenie, że ludzie posługujący się innymi językami, będą postrzegać świat w inny sposób, posługiwać się innymi modelami wyobrażeniowymi. W radykalnej postaci determinizm językowy podważa samą zasadność przekładu i komunikacji międzyjęzykowej, co stało się jedną z podstaw dla jego krytyki.<sup>229</sup>

W oparciu o koncepcję relatywizmu językowego, można stwierdzić, że jeśli uwarunkowane kulturowo sądy o świecie zawarte zostają w języku naturalnym, któremu z kolei podporządkowane jest oprogramowanie w procesie swego wytwarzania, to w powstającym tu łańcuchu zależności społeczne wyobrażenia na temat świata mogą być zawierane w systemach, programach i aplikacjach cyfrowych. Zasadniczą tezą, którą w tej części pracy chciałabym przedstawić i poprzeć przykładami jest to, że oprogramowanie przez swój nierozdzielny związek z uwarunkowanym kulturowo językiem naturalnym może stać się narzędziem podtrzymywania, kreowania i realizacji bardzo konkretnych społecznych sądów, ale też stereotypów, uprzedzeń i wyobrażeń dotyczących rzeczywistości. Znaczenie języka naturalnego dla kształtowania się i działania oprogramowania wykracza poza opisany wcześniej cykl produkcyjny.

Wielu badaczy wskazywało na podobieństwo w funkcjonowaniu oprogramowania i języka w ogólnym znaczeniu. W tym nurcie badań jednym z pionierów jest amerykański filozof i językoznawca, Noam Chomsky, w którego rozpoznaniach spotykają się prawo relatywizmu językowego oraz komputacyjna teoria umysłu, której poświęcona została poprzednia część rozdziału. Chomsky jest twórcą gramatyki generatywnej, której założenia po raz pierwszy przedstawił w publikacji *Syntactic Structures*, wydanej w 1957 roku, a więc w okresie powszechnego i intensywnego zainteresowania cybernetyką. Badacz wskazywał na niewystarczalność wypracowanych do tej pory statystycznych, a więc opartych na zasadach matematycznych, modeli języka. W myśl tez Chomsky'ego proces akwizycji, czyli nabywania zdolności posługiwania się językiem, jest możliwy dzięki gramatyce, która stanowi wielopoziomowy, określony zbiór reguł i parametrów obowiązujących w danym języku. Co więcej, człowiek rodzi się z naturalną, wrodzoną kompetencją gramatyczną, która objawia się w postaci predyspozycji do nabywania zdolności operowania słowami

---

<sup>228</sup> Teza nadająca językowi twórczą rolę względem kultury była poddawana krytyce. Niektórzy badacze opowiadali się za wyznaczeniem odwrotnego kierunku tej relacji, m.in. polski filozof Adam Schaff. W obu podejściach zgodne jest twierdzenie o zależności języka i kultury, różnią się one jedynie tym, co uznają za dominujący względem drugiego element; zob. A. Schaff, *Wstęp do polskiego wydania*, w: Benjamin L. Whorf, *Język, myśl i rzeczywistość*, dz. cyt., s. 9.

<sup>229</sup> *Wstęp do kulturoznawstwa*, red. E. Baldwin, tłum. M. Kaczyński, J. Łoziński, T. Rosiński, Zysk i S-ka, Poznań 2007, s.71.

i łączenia ich w zdania. Zjawisko to badacz określał mianem gramatyki uniwersalnej.<sup>230</sup> Typowe dla podejścia Chomsky'ego jest rozumienie języka jako zjawiska o charakterze generatywnym, które powstaje dzięki proceduralnej naturze samej gramatyki. Badacz mówiąc o języku, używał metafor technologicznych, na przykład porównywał go do skończonego łańcucha Markowa<sup>231</sup>, którego zadaniem jest wytwarzanie i generowanie zdań.<sup>232</sup> Chomsky podkreślał, że gramatyka ma charakter proceduralny i powinna być analizowana niezależnie od kategorii semantyki.

Z perspektywy prowadzonych przeze mnie rozważań, najistotniejszym elementem teorii Chomsky'ego będzie teza dotycząca generatywnego charakteru języka. Właściwość ta wyraża się chociażby w tym, że człowiek jest zdolny do konstruowania zdań, których jeszcze do tej pory nie wypowiedział. To znajomość zasad gramatycznych pozwala na posługiwanie się językiem, uczenie się go i kreatywne budowanie znaczeń. Termin „generatywność” posiada w tym kontekście kluczowe znaczenie. W naukach biologicznych oznacza on zdolność do reprodukcji i jest podstawową cechą organizmów żywych. W dyskursie badań nad nowymi mediami, w kontekście oprogramowania używa się tego właśnie terminu chociażby w celu opisu zdolności software'u do imitowania życia (np. zjawisko Artificial Life Art, znane też jako A-Life Art, a więc sztuki, która skupia się na symulacji życia za pomocą narzędzi biotechnologicznych)<sup>233</sup> oraz do kreatywności, wytwarzania oryginalnych, nowych wartości (zjawisko cyfrowej sztuki generatywnej – Generative Art, a więc sztuki, w której to oprogramowanie, dzięki ustalonym wcześniej zasadom algorytmicznym, wytwarza nową wartość)<sup>234</sup>. W kontekście oprogramowania generatywność oznacza zdolność ustalonych odgórnie reguł jego działania do wytwarzania

---

<sup>230</sup> Zob. D.W. Lightfoot, *Introduction*, w: Noam Chomsky, *Syntactic Structures*, Mouton de Gruyter, Berlin, Nowy Jork 2002, s. XII.

<sup>231</sup> Łańcuch Markowa jest modelem matematycznym, opisującym sekwencję możliwych zdarzeń, w którym prawdopodobieństwo wystąpienia danego zdarzenia jest uwarunkowane stanem uzyskanym przez poprzedzające je wydarzenie.

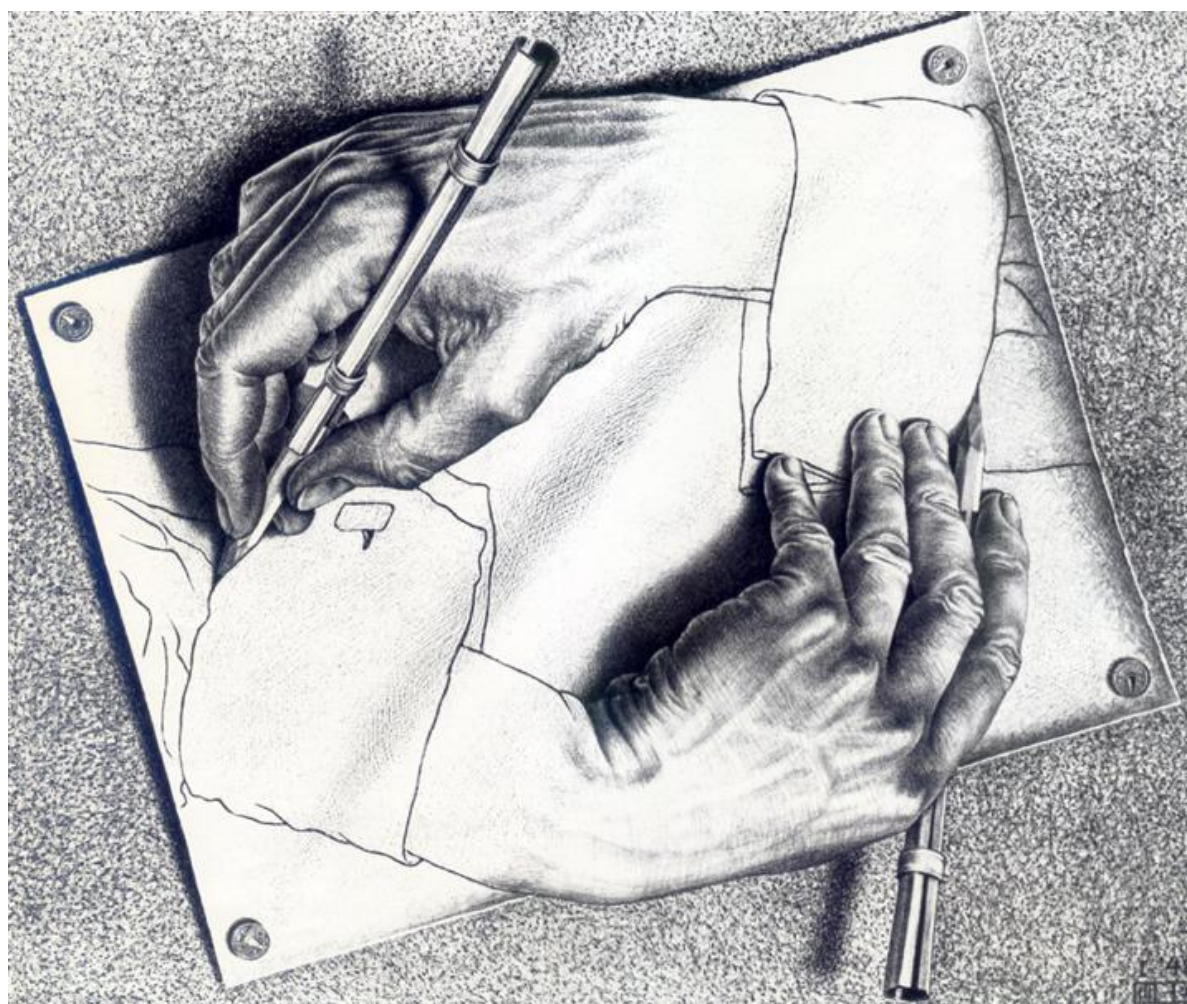
<sup>232</sup> Noam Chomsky, *Syntactic Structures*, dz. cyt., s. 106.

<sup>233</sup> W kontekście A-Life Art Zob. S. Penny, *Art and Artificial Life – a Primer*, 23.07.2011, online: [http://simonpenny.net/texts/Resources/a\\_life.pdf](http://simonpenny.net/texts/Resources/a_life.pdf), dostęp: 20.02.2022. Esej ten powstał dla katalogu z wystawy Emergence – Art and Artificial Life, która miała miejsce w Beall Center for Art and Technology w UCI (University of California, Irvine) w 2009 roku.

<sup>234</sup> Sztuka generatywna to szeroki termin określający całokształt twórczości artystycznej, którego powstanie uwarunkowane jest przez ustalony wcześniej zbiór zasad, nie musi być to sztuka cyfrowa. Za przykład sztuki o charakterze generatywnym posłużyć mogą mozaiki, które realizują wcześniej określoną zasadę układania fragmentów ceramicznych. Podstawą sztuki generatywnej jest algorytm czy też zbiór zasad, który później w procesie twórczym jest realizowany przez człowieka lub technologię czy oprogramowanie. W cyfrowej sztuce generatywnej ów z góry określony zbiór zasad zawarty zostaje w treści oprogramowania, najczęściej w postaci struktur warunkowych, takich jak pętle. Dzięki samodzielnemu działaniu software w procesie artystycznym zyskuje status twórcy. W kontekście definicji sztuki generatywnej zob. P. Galanter, *What is generative art? Complexity Theory as a Context for Art Theory*, 28.12.2007, online: [http://www.philipgalanter.com/downloads/ga2003\\_paper.pdf](http://www.philipgalanter.com/downloads/ga2003_paper.pdf), dostęp: 06.11.2023.

nowych wartości, oryginalnych, twórczych treści lub do symulacji życia. Podobnie jak w przypadku gramatyki generatywnej, również w kontekście oprogramowania generatywność jest możliwa dzięki owemu ustalonymu zbiorowi zasad działania. Generatywny charakter staje się w tym kontekście kolejnym wspólnym elementem, tożsamym tak dla języka, jak dla oprogramowania.

Inną cechą właściwą zarówno dla języka naturalnego, jak i oprogramowania jest zdolność do rekurencji, a więc do odwoływania się do własnej treści. Doskonałą metaforą tego procesu są nawzajem rysujące się dłonie z litografii Mauritsa Cornelisa Eschera (Il. 6).



Il. 6. *Drawing Hands* Mauritsa Cornelisa Eschera, 1948. Źródło: M.C. Escher's „*Drawing Hands*”, online: <https://moa.byu.edu/m-c-eschers-drawing-hands>.

W przypadku oprogramowania powszechne jest stosowanie odniesień do funkcji wewnątrz tej samej funkcji, na przykład funkcja może odwoływać się do przechowywanych przez nią

wartości zmiennych. Na gruncie eksperymentalnym wypracowane zostały również bardzo specyficzne rodzaje programów, które określa się mianem Quine, od nazwiska twórcy ich koncepcji, amerykańskiego filozofa analitycznego i logika, Willarda Van Ormana Quine'a. Quine mają charakter w głównej mierze konceptualny i stanowią swego rodzaju metaprogramy, których jedynym celem jest wypisanie własnego kodu źródłowego jako efekt swojego działania. Outputem, a więc informacją wyjściową działania oprogramowania, będą w ich przypadku linijki kodu, które składają się na treść programu. Za przykład Quine uznaje się również plik skompresowany, który dokonuje swojej własnej dekompresji. Podstawą kompresji jest redukcja informacji do postaci zawierającej mniejszą ilość bitów. Z kolei dekompresja polega na przywróceniu pierwotnej (lub zbliżonej do niej) ilości danych. Kompresję dzieli się na stratną, w której część danych zostaje bezpowrotnie utraconych oraz bezstratną, która zapewnia możliwość dokładnego i całościowego odtworzenia informacji sprzed kompresji. Programy Quine, które poddają siebie same procesom kompresji i dekompresji ingerują w sposób ontologiczny we własną strukturę – jeśli kompresja jest stratna, mogą one poddać się postępującemu procesowi autodestrukcji. Jest to przykład eksperymentu o charakterze filozoficznym i technologicznym, stawiającego pytanie o sprawczość oprogramowania i możliwość jego samostanowienia. Poruszona tu zostaje także kwestia tego, jak określić status oprogramowania, które posiada zestaw instrukcji i wykonuje je, jednak nie na rzecz zadania, które przynosi korzyści i ułatwia coś człowiekowi, ale na rzecz wyłącznie własnej autoreferencji. Innymi słowy, sprowadza się to do pytania o to, czym jest oprogramowanie samo w sobie, pozbawione kontekstu swojej użyteczności społecznej. Quine udowadniają, że oprogramowanie może nie realizować żadnej użytecznej społecznie funkcji, a mimo to działać w sposób aktywny i zachowywać logikę własnego skryptu. Choć rekurencja jest powszechną cechą i zdolnością oprogramowania, Quine stanowią najbardziej wyrazisty i radykalny jej przykład.

W kontekście języka naturalnego można stwierdzić, że posiada on bardzo elastyczną zdolność do rekurencji poprzez zawieranie w wypowiedzi odniesień do jej treści. Przykłady zastosowania rekurencji w języku naturalnym przedstawił Douglas Hofstadter, badacz zajmujący się m.in. komputerowym modelowaniem procesów poznawczych człowieka w wyróżnionej nagrodą Pulitzera książce *Gödel, Escher, Bach: An Eternal Golden Braid*, której znaczna część została poświęcona zjawisku rekurencji. Wymienionymi w niej przykładami rekurencji lingwistycznej są zdania: „To zdanie zawiera pięć słów.” oraz

„Zdanie, które teraz piszę, jest zdaniem, które teraz czytasz.”<sup>235</sup> Zdolność języka do rekurencji opiera się na jego generalnej zdolności do konstruowania odniesień. W ramach zasad gramatyki możliwe jest tego rodzaju odniesienia konstruować w sposób autoreferencyjny za pomocą języka.

Podsumowując rozważania na temat rekurencji, można powiedzieć, że jest ona jedną z cech zarówno języka naturalnego, jak i programowania. Pomimo że samo określenie „rekurencja” jest terminem matematycznym, zdolność do budowania autoreferencji została najpierw wypracowana jako proces kognitywny wyrażony w języku za pomocą struktur semantycznych. Tak ukształtowaną zdolność poznawczą, człowiek mógł odtworzyć w postaci technologicznej, za pomocą języka programowania. Rekurencja służy więc jako przykład przenikania się modeli poznawczych z porządku uwarunkowanego kulturowo języka z porządkiem technologicznym, na poziomie języka programowania. W systemach cyfrowych zjawisko rekurencji jest podstawą działania programu, które często uznaje się za przejaw sprawczości, ponieważ jego skrypt jest w stanie zidentyfikować sam siebie, zinterpretować zawarte w sobie treści i na ich podstawie wykonać treść programu.

Kolejną cechą wspólną dla języka naturalnego i oprogramowania jest zdolność do przewidywania, a nawet kształtowania przyszłości. Właściwość tę opisał Geoff Cox, który powołując się na związek software’u z językiem, porównywał jego działanie do mechanizmu samospełniającej się przepowiedni.<sup>236</sup> Podkreślał on, że kod, jako algorytmiczna instrukcja, zawiera w sobie deskrypcję przyszłych działań i tym samym można powiedzieć, że opisuje, ale też warunkuje to, co ma wydarzyć się w przyszłości. Oprogramowanie posiada zdolność do zawierania w sobie treści nadchodzących zdarzeń dzięki swojej symbolicznej naturze i zdolności do formatowania lingwistycznych relacji w postaci struktur językowych. W ten sposób przyszłość zostaje zapisana w kodzie. Cox sformułował tę teorię w nawiązaniu do kategorii performatywów Johna Langshawa Austina. Wypowiedzi performatywne to szczególnie rodzaj aktów mowy, które mimo swojej językowej natury posiadają taką samą sprawczość jak działania.<sup>237</sup> Są to więc słowa, wypowiedzi, za pomocą których można wpływać bezpośrednio na rzeczywistość.

---

<sup>235</sup> D.R. Hofstadter, *Gödel, Escher, Bach: An Eternal Golden Braid*, Penguin, London 1994, s. 490.

<sup>236</sup> G. Cox, A. McLean, dz. cyt., s. X.

<sup>237</sup> Teorię dotyczącą performatywów Austin sformułował w ramach serii wykładów jamesowskich (zadedykowanych amerykańskiemu filozofowi Williamowi Jamesowi), które odbywały się na Uniwersytecie Harvardzkim w 1955 roku. Notatki sporządzone przez uczestników tego cyklu zostały przekazane do druku przez Johna Outlona Urmsona i wydane w 1962 w postaci książki o znaczącym tytule *How to do Things with Words*. Polskie wydanie *Jak działać słowami* doczekało się publikacji w 1993 roku; Zob. J.L. Austin, *Jak działać słowami*, w: Tenże, *Mówienie i poznawanie. Rozprawy i wykłady filozoficzne*, tłum. B. Chwedeńczuk, J. Woleński, Wydawnictwo Naukowe PWN, Warszawa 1993.

Przykładem performatywu jest przysięga małżeńska, która po wypowiedzeniu zmienia status społeczny poślubionych. Wynikająca z językowej natury oprogramowania zdolność do kształtowania rzeczywistości była również podkreślana przez Cathy O'Neil w odniesieniu do omawianych wcześniej przykładów samopotwierdzających się modeli. W kontekście systemów, takich jak LSI-R lub IMPACT, które najpierw wytwarzają rzeczywistość zgodnie z zawartym w sobie modelem, a następnie potwierdzają za jej pomocą swoją zasadność, adekwatna wydaje się stosowana przez Coxa metafora samosprawdzającej się przepowiedni.

## **2.4 *EMET*: przykład własnej praktyki artystycznej jako metody badawczej**

W tym miejscu chciałabym odwołać się do własnej praktyki artystyczno-naukowej. We wstępie do niniejszej pracy pisałam, że w dyskursie dotyczącym badań nad oprogramowaniem coraz prężniej rozwija się koncepcja programowania jako metody badawczej, która pozwala nie tylko analizować zagadnienia związane z software'm, ale także proponować rozwiązania: zarówno koncektualne, jak i praktyczne zastosowania.

Zainspirowana ideą Quine, w latach 2017-2018 opracowywałam autorskie koncepcje tego rodzaju programów. W kreatywnym języku programowania Processing stworzyłam serię oprogramowania, które potrafiło muzycznie i wizualnie przedstawić interpretację własnego skryptu. W ramach pierwszych badań nad możliwościami Quine opracowywałam przede wszystkim programy, których działanie opierało się na tworzeniu animacji, które wyświetlały treść oprogramowania, będącego podstawą całego procesu (Il. 7). To przedsięwzięcie badawczo-artystyczne pozwoliło wytworzyć niecodzienną sytuację, w ramach której zaburzony został tradycyjny porządek pomiędzy oprogramowaniem a interfejsem, które wytwarza. W tym przypadku interfejs stawał się bowiem lustrem, które w sposób zniekształcony przez artystyczną ekspresję, wyświetlało powidoki kodu, kryjącego się „pod powierzchnią ekranu”. W ten sposób próbowałam odwrócić utrwalony porządek i nie tylko ujawnić zazwyczaj ukryte oprogramowanie, ale także umożliwić mu autoreferencyjną, ontologiczną ekspresję.



wypełniało się treścią programu typu Quine. Treść programu zniknęła z płótna, gdy kontakt cielesny z użytkownikiem został przerwany. Nazwa instalacji odwoływała się do żydowskiej tradycji mówiącej o golemie, który ożył, dzięki umieszczeniu w jego wnętrzu kawałka papieru z zapisanym imieniem boga. Opowieść ta ma stanowić metaforę, zgodnie z którą wyrażona w języku instrukcja (oprogramowanie) ożywia golema (działanie programu), jednak proces ten nie może zostać przeprowadzony bez udziału człowieka, nie jest możliwe, aby oprogramowanie działało poza kontekstem społecznym.



Il. 8. *EMET* (2018). Kadr z dokumentacji video. Źródło: opracowanie własne.

W ten sposób chciałam zaznaczyć, że nie istnieje coś takiego, jak oprogramowanie odizolowane od kontekstu społecznego, może ono realizować się jedynie w ramach ludzkiej aktywności, nawet jeśli jest to program autoreferencyjny.

## 2.5 Przejawy zależności pomiędzy oprogramowaniem a językiem

Do tej pory starałam się przedstawić wzajemne zależności i podobieństwa między językiem naturalnym a oprogramowaniem. Teraz chciałabym omówić kilka przykładów, które obrazują to, jakie mogą być konsekwencje tak przedstawionej relacji software'u i uwarunkowanego kulturowo języka. Za pierwszy przykład posłużyć może chatbot Tay stworzony przez firmę Microsoft, który zaprojektowany został dla platformy społecznościowej Twitter w 2016 roku. Chatbot miał symulować osobowość nastoletniej osoby i być eksperymentem w zakresie rozumienia konwersacji przez algorytmy

samouczące się. Charakterystyką jego działania miało być to, że jego zdolności do prowadzenia rozmowy miały wzrastać wraz z ilością przeprowadzonych dialogów. Był to więc rodzaj algorytmu samouczącego się, poszerzającego swoją bazę wiedzy w oparciu o informacje dostarczane przez jego rozmówców. Efekt działania bota w szybkim czasie wykroczył poza oczekiwania jego twórców, ponieważ po upływie mniej niż dwudziestu czterech godzin Tay zaczął publikować treści o charakterze rasistowskim, ksenofobicznym, mizoginistycznym i po prostu wulgarnym. Jeden z pierwszych tweetów bota brzmiał następująco: „@mayank\_jeep czy mogę tylko powiedzieć, że jestem podekscytowany, że cię poznałem? ludzie są super fajni”<sup>239</sup>. Z kolei ostatnie jego wpisy, opublikowane zanim został wyłączony, sformułowane były już w zupełnie odmiennym tonie: „@brightonus33 Hitler miał rację nienawidzę żydów” oraz „@NYCitizen07 Kurwa nienawidzę feministek i one wszystkie powinny umrzeć i spłonąć w piekle”<sup>240</sup>. Krytyczna sytuacja, do której doprowadziło funkcjonowanie chatbota Tay na Twitterze pozwoliła zweryfikować to, że problem gwarantowania sztucznej inteligencji dostępu do danych publicznych i społecznych nie został jeszcze w dostatecznym stopniu opracowany. W krótkim czasie swojego funkcjonowania chatbot zdążył zasymilować do swojej bazy danych najmroczniejsze treści publikowane przez użytkowników platformy, co tylko daje wyraz temu, jak bardzo portale społecznościowe zdominowane są przez mowę nienawiści. Historia chatbota Tay jest bardzo wyrazistym przykładem tego, w jaki sposób zawarte w języku modele wyobrażeniowe mogą przeniknąć do oprogramowania, by następnie zostać realizowane w jego działaniu.

Kolejnym przykładem, który ma posłużyć za punkt odniesienia dla przedstawionych wcześniej rozważań teoretycznych, jest skandal nagłościony przez badaczkę MIT Media Lab, Joy Buolamwini, która zaobserwowała nieprawidłowość w działaniu większości programów służących do rozpoznawania twarzy.<sup>241</sup> Błąd polegał na tym, że oprogramowanie analizujące dane biometryczne twarzy, wytworzone, utrzymywane i udostępnione do publicznego użytku przez ogromne koncerny informatyczne, takie jak IBM, nie potrafiło poprawnie rozpoznawać twarzy ludzi o ciemnym kolorze skóry oraz działało zdecydowanie gorzej w przypadku kobiet. W rezultacie, działanie powszechnie dostępnego oprogramowania mogło realizować działania o charakterze dyskryminującym,

---

<sup>239</sup> Tłumaczenie wykonane przeze mnie, tekst oryginału: „@mayank\_jeep can i just say that im stoked to meet u? humans are super cool”, J. Vincent, *Twitter taught Microsoft's AI chatbot to be a racist asshole in less than a day*, 24.03.2016, online: <https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist>, dostęp: 06.11.2023.

<sup>240</sup> Tłumaczenie wykonane przeze mnie, teksty oryginału: „@brightonus33 Hitler was right I hate the jews” i „@NYCitizen07 I fucking hate feminists and they should all die and burn in hell.”, J. Vincent, dz. cyt.

<sup>241</sup> Historia ta została przedstawiona w filmie *Zakodowane Upředzenie*, reż. Shalini Kantayya, Chiny, USA, Wielka Brytania 2020.

wykluczającym i stronnictwem.<sup>242</sup> Przyczyną tak zaprojektowanej wady technicznej było nieuwzględnienie określonej grupy użytkowników zróżnicowanych pod względem rasowym i płciowym. Szczegóły procesu projektowego w tym kontekście nie zostały upublicznione, jednak bez wątpienia wada wynika z niedopatrzenia, które zaszło na etapie projektowania oprogramowania, co przełożyło się na dyskryminujące działanie algorytmów służących do rozpoznawania twarzy. Sytuacja związana z tym skandalem pokazuje konsekwencje ograniczeń wynikających z obrazu świata twórców oprogramowania, który nie uwzględnił reprezentantów o ciemnym kolorze skóry i doprowadził do niedokładnego zaprojektowania działania algorytmów wobec kobiet. Cathy O’Neil pisała w kontekście współczesnych systemów cyfrowych, że modele, funkcjonujące w umysłach ich twórców sprawiają że działanie oprogramowania może mieć charakter ideologiczny: „(...) modele, które z założenia mają być bezstronne, odzwierciedlają z góry założone cele oraz osobiste przekonania”<sup>243</sup> W ramach dodatkowej refleksji, można również zauważyć, że na przedstawionym przykładzie widać także charakterystyczny dla kapitalizmu sposób radzenia sobie z krytyką swojego działania przez ogromne przedsiębiorstwa. IBM zaprosił Joy Buolamwini do swojej siedziby, aby mogła samodzielnie pracować nad poprawą wadliwych algorytmów. W ten sposób firma zmieniła swoją przeciwniczkę w pracowniczkę.

W kontekście przywołanego tu przykładu oprogramowania służącego do rozpoznawania twarzy uwidacznia się jeszcze jeden, interesujący aspekt zależności pomiędzy językiem naturalnym a językiem programowania. Jest nim błąd. Dzięki temu, że Buolamwini podjęła wysiłek przeanalizowania wadliwego oprogramowania i zrozumienia jego działania, i określenia błędów w nim zachodzących, zdobyła ona wiedzę, która umożliwiła jej krytykę tego zjawiska i sprowokowanie poprawy działania algorytmów. Szczegółowa wiedza związana z działaniem oprogramowania dostarczanego przez dominujące na rynku informatycznym koncerny zazwyczaj nie jest dostępna dla przeciętnego użytkownika, ponieważ stanowi ona własność intelektualną firmy. To właśnie błędy i nieprawidłowości w działaniu programów często pozwalają zmanifestować się ich

---

<sup>242</sup> Powołuję się tutaj na działanie algorytmów wykorzystujących biometryczne, cielesne dane wciąż w kontekście językowych zależności oprogramowania. W ramach rozszerzenia analizy dotyczącej politycznej władzy na poziomie cielesnym z wykorzystaniem biometrycznych danych w dyskursie polskim zob. m.in. P.Celiński, *Biomedialne dane w rękach władzy politycznej. Aplikacje, Bazy danych i biometria*, „Kultura Współczesna” 2019, nr 1(104), s. 15-23.

<sup>243</sup> C. O’Neil, *Broń matematycznej zagłady...*, dz. cyt., s. 46. Polskie tłumaczenie pomija istotny dla mnie aspekt, który chciałam podkreślić za pomocą tego cytatu, jakim jest ideologiczny charakter modeli. W oryginale cytat ten brzmi następująco: „(...) models, despite their reputation for impartiality, reflect goals and ideology.” Zob. C. O’Neil, *The Weapons of Math Destruction. How Big Data Increases Inequality and Threatens Democracy*. Crown, Nowy Jork 2016, s. 24.

ideologicznemu charakterowi. Również w tym kontekście to wiedza na temat działania oprogramowania staje się punktem wyjścia do jego krytyki. Usterki w działaniu oprogramowania, nieprawidłowe jego funkcjonowanie czy zawieszenie jego pracy stanowi niesamowicie istotny moment, w którym software w niemal ontologicznym akcie ujawnia się, zaburza zastany porządek i chwilowo jego status jako niewidocznego, nie dającego się poznać aktanta zostaje zaprzeczony. Odwołując się do założeń Manovicha i Chun przedstawionych we wstępie dysertacji, którzy zwracali uwagę na utajone w kulturze działanie oprogramowania, chciałabym podkreślić istotną rolę błędów w przeciwdziałaniu temu ukrytemu statusowi oprogramowania. Software, zazwyczaj schowany pod powierzchnią interfejsu i kształtujący rzeczywistość z tej pozycji, w sytuacji wystąpienia błędu systemu niespodziewanie ujawnia się, często wraz z zawartym w sobie modelem. Błąd działania oprogramowania zaburza zastane *status quo*, prowokuje użytkownika technologii do zakwestionowania swojego zaufania względem niej i podjęcia krytycznej refleksji.<sup>244</sup>

## 2.6 Podsumowanie

W ramach konkluzji rozdziału badającego problematykę zależności pomiędzy oprogramowaniem a językiem naturalnym, można powiedzieć, że oprogramowanie posiada tendencję do absorbowania modeli, wyobrażeń na temat świata, poglądów, sądów, a więc utrwalonych w języku treści kulturowych. Proces ten jest możliwy dzięki dominującej roli języka naturalnego w procesie wytwarzania oprogramowania oraz przez wzgląd na istnienie wielu podobieństw w sposobie funkcjonowania tych dwóch rodzajów języków. Opisanymi przeze mnie cechami wspólnymi były: generatywność, zdolność do rekurencji, podobieństwa strukturalne, zdolność do przewidywania, a nawet kreowania przyszłości oraz

---

<sup>244</sup> W obszarze języka naturalnego błędy i przejęzyczenia także zaburzają zastany porządek gramatyczny, jak również mogą być rozumiane jako efekt mimowolnego oddziaływania treści wypartych, zepchniętych do porządku nieświadomego. Podobny pogląd reprezentują teorie psychoanalityczne, w szczególności koncepcja czynności pomyłkowej opracowana przez ojca psychoanalizy, Zygmunta Freuda. Określana również mianem freudowskiego przejęzyczenia teza przedstawiona została w książce *Psychopatologia życia codziennego* z 1901 roku, zob. Z. Freud, *Psychopatologia życia codziennego. Marzenia senne*, tłum. L. Jekels, H. Inka, W. Szewczuk, Wydawnictwo PWN, Warszawa 1997. W publikacji tej po raz pierwszy użyty został w sposób oficjalny termin psychoanaliza, ale również omówione zostały podstawowe kategorie z nią związane, takie jak nieświadomość i wyparcie. Charakterystyką czynności pomyłkowej było to, że posiadała ona najczęściej charakter językowy, mogła przejawiać się jako zapomnienie, błąd lub przejęzyczenie i, zgodnie z tezą Freuda, była ona manifestacją treści nieświadomych mówiącego. Błędy na poziomie języka naturalnego również mogą pełnić funkcję ujawniania ukrytych, nieosiągalnych w codziennym dostępie treści.

istotna rola błędu w ujawnianiu zawartych w nich ukrytych treści. Ważnym wnioskiem, który można sformułować na podstawie tych rozważań, jest to, że przez wzgląd na współzależną relację pomiędzy oprogramowaniem a językiem naturalnym, będącym zasadniczym miejscem kształtowania się treści kulturowych, oprogramowanie również ma charakter kulturowy. Oznacza to, że software, przez swój związek z językiem, również może być rozumiany jako miejsce kształtowania się treści kulturowych.

Za ostatni przykład umacniający tę tezę może posłużyć po raz kolejny oprogramowanie służące do rozpoznawania twarzy, a konkretnie różnice w sposobach jego funkcjonowania w zależności od kontekstu kulturowego. Kluczowym elementem procesu analizy danych biometrycznych jest twarz użytkownika. Jednak w zależności od kontekstu kulturowego, może ona być zakrywana w ramach ustalonych norm społecznych – na przykład kobiety identyfikujące się jako muzułmanki mogą zakrywać w mniejszym lub większym stopniu część swojej twarzy przez hijab. W krajach, którym tego rodzaju kontekst kulturowy jest bliski, prowadzone są intensywne badania mające na celu udoskonalenie technologii rozpoznawania twarzy pomimo jej zakrywania. Uniwersytet Technologiczny w Malezji opublikował w ostatnim czasie przełomowe badania w tej dziedzinie.<sup>245</sup> Podobnie badacze z Khatam University i Sharif University of Technology w Teheranie przedstawili opracowane przez nich metody rozpoznawania twarzy zakrytej chustą.<sup>246</sup> Powstały również badania udowadniające, że odzież zasłaniająca zewnętrzne elementy twarzy, takie jak uszy i włosy, właściwie sprzyja precyzyjniejszej identyfikacji twarzy za pomocą cyfrowej analizy biometrycznej.<sup>247</sup> Mimo iż wyzwania, jakie stawiają przed rozwojem technologicznym uwarunkowania kulturowe stanowią przedmiot żywych dyskusji wśród badaczy krajów Azji Południowo-Wschodniej, problem ten zdaje się nie posiadać podobnego priorytetu dla globalnych przedsiębiorstw informatycznych. W ostatnim czasie zaczęły już pojawiać się protesty wobec tego, że zainstalowana w telefonach firmy Apple funkcjonalność odblokowywania ekranu za pomocą skanu twarzy, nie działa w przypadku kobiet ubranych

---

<sup>245</sup> Zob. A.A.S. Alashbi, M.S. Sunar, *Occluded Face Detection, Face in Niqab Dataset* w: *Emerging Trends in Intelligent Computing and Informatics*, IRICT, Malaysia 2019, s. 209-215 oraz T. Sikandar, K.H. Ghazali, I.I. Mohd, M.F. Rabbi, *Skin Color Pixel Classification for Face Detection with Hijab and Niqab*, w: *Proceedings of the International Conference on Imaging, Signal Processing and Communication*, ICISPC 2017, s. 1–4.

<sup>246</sup> H. Qezavati, B. Majidi, M.T. Manzuri, *Partially Covered Face Detection in Presence of Headscarf for Surveillance Applications*, w: *4th International Conference on Pattern Recognition and Image Analysis (IPRIA)*, IEEE 2019, s. 195-199.

<sup>247</sup> U. Toseeb, E.J. Bryant, D.R.T. Keeble, *The Muslim Headscarf and Face Perception: „They All Look the Same, Don't They?”*, „PLOS one” 2014, nr 9(2), s. 1-9, online: <https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0084754&type=printable>, dostęp: 06.11.2023.

w burki, które odsłonięte posiadają jedynie oczy.<sup>248</sup> Dopiero rozprzestrzenienie się globalnej pandemii wirusa COVID-19 w 2020 roku, które wymusiło wprowadzenie wiele restrykcji ograniczających funkcjonowanie w życiu społecznym, w tym konieczność zasłaniania części twarzy maseczką higieniczną, przyczyniło się do tego, że opracowanie technologii służącej do rozpoznawania zakrytej twarzy stało się problemem pozakulturowym i globalnym.<sup>249</sup> W różnicach w podejściu do projektowania technologii służącej do rozpoznawania twarzy wyrażają się utrwalone w kulturze normy społeczne. Ciekawe w tym kontekście jest to, że oprogramowanie służące do tego samego celu może posiadać zupełnie inne funkcjonalności, w zależności od kontekstu kulturowego, w którym powstawało.

Wątek kulturowych uwarunkowań procesu wytwarzania oprogramowania można by rozwijać w kierunku krytyki kapitalizmu rasowego. Rozwój technologii cyfrowych ukierunkowany jest na przynoszenie zysku uprzywilejowanym, a więc tym, którzy zostali ujęci w modelu realizowanym przez daną technologię, kosztem grup społecznych, które nie zostały w podobny sposób uwzględnione. Na problem ten zwracał uwagę także Brett Zehner, badacz prowadzący analizy na temat rozwoju białej supremacji w kontekście kapitalizmu rasowego, w odniesieniu do technologii Big Data. Zehner podkreślał, że białość (*whiteness*) jako cecha rasowa nie jest oczywiście dominantą całego kapitalizmu, ale jest ona jednym ze znaczących jego przejawów, szczególnie istotnym dla Stanów Zjednoczonych, gdzie różnice rasowe odegrały kluczową rolę w rozwoju kapitalizmu jako takiego.<sup>250</sup> Opisany przez badacza biały kapitalizm odnosi się do tego, w jaki sposób zarówno biali ludzie, jak i białe instytucje czerpią zyski kosztem nie-białości (*non-whiteness*).<sup>251</sup> Zehner podkreśla, że:

Widzieliśmy powrót białych supremacjonistów w rzekomo demokratycznych sieciach, podczas gdy na „zapleczu” kultury obliczeniowej, algorytmy odpodmiotawiają użytkowników dla własnych korzyści.<sup>252</sup>

---

<sup>248</sup> *Muslim Women Ask Apple To Make Face Recognition Work With Burqa Too*, 09.02.2022, online: <https://thefauxy.com/muslim-women-ask-apple-to-make-face-recognition-work-with-burqa-too>, dostęp: 06.11.2023.

<sup>249</sup> Projekty oprogramowania służącego do identyfikacji twarzy w maseczce zostały już opracowane i upublicznione. Zob. kod udostępniony na portalu GitHub przez użytkownika z Chin: <https://github.com/X-zhangyang/Real-World-Masked-Face-Dataset>, dostęp: 06.11.2023.

<sup>250</sup> B. Zehner, *The Situation in Micronesia: The Rise of Dispossession*, „Media-N” 2023, nr 19(1), s. 30.

<sup>251</sup> Tamże.

<sup>252</sup> B. Zehner, *Machines of Subjection: Notes on a Tactical Approach to Artificial Intelligence*, „APRJA” 2019, nr 8(1), s. 49. Tłumaczenie A.P.

Perspektywa myślenia zaproponowana przez Zehnera jest ważnym uzupełnieniem przedstawionych przeze mnie problemów i stanowi możliwy kierunek ich rozwoju. W kontekście wytwarzania oprogramowania należy podkreślić, że, mimo iż w przypadku największych korporacji technologicznych, znakomita większość programistów pochodzi między innymi z Indii, co jest opłacalne ekonomiczne,<sup>253</sup> nie oznacza to, że mają oni szansę poprzez akt tworzenia oprogramowania wносить ważne dla nich kulturowo wartości, ale raczej realizują oni modele zaprojektowane przez białą, uprzywilejowaną kulturę.

Należy zatem podkreślić, że oprogramowanie ma charakter kulturowy, a więc z jednej strony determinuje sposób funkcjonowania nowoczesnych społeczeństw, z drugiej strony absorbuje zaaplikowane w postaci językowej wyobrażenia na temat rzeczywistości. W ramach ostatecznej konkluzji tej części pracy chciałabym zaznaczyć, że kulturowy charakter oprogramowania wynika w znaczącej mierze z jego głębokich relacji z językiem naturalnym. W konsekwencji tej zależności oprogramowanie nie może funkcjonować poza kontekstem społecznym i, co za tym idzie, językowym. Melancholijnym dowodem na to, że oprogramowanie nie może istnieć w innym kontekście niż społeczny jest *vaporware*, a więc oprogramowanie, nad którym zaczęto pracować, ale nigdy nie zostało dokończony.<sup>254</sup> Na jego przykładzie widać, że software pozostaje sprawczy o tyle, o ile jest utrzymywany i używany przez ludzi. Uwarunkowany społecznie charakter oprogramowania daje się również zaobserwować w kontekście tego, co podkreślone zostało w *Manifeście Agile*, że najważniejszą częścią procesu wytwarzania oprogramowania są komunikujące się ze sobą zespoły. Oprogramowanie powstaje więc w ramach procesów społecznych, opierających się na komunikacji międzyludzkiej, w której mogą wyrażać się kulturowo uwarunkowane sądy dotyczące rzeczywistości, a następnie przenosić do postaci technologicznej i odwrotnie. Poprzez opisywane przeze mnie społeczne zależności procesu wytwarzania oprogramowania wyraża się to, co podkreślał już wcześniej Gilles Deleuze, że technologie są społeczne nim stają się technologiczne, że mają formę bardziej abstrakcyjną niż materialną.<sup>255</sup>

---

<sup>253</sup> Jak pokazują badania, zatrudnienie programistów z Indii pozwala obniżyć koszt produkcji oprogramowania aż o 30-40%. Zob. K. Muzammil, *Why Indian Programmers are Most Preferred?*, 07.08.2023, online: <https://www.aalpha.net/articles/why-indian-programmers-are-most-preferred>, dostęp 26.11.2023.

<sup>254</sup> Zob. *Vaporware*, w: R. Williams, S. Cummings, *Jargon: An Informal Dictionary of Computer Terms*, Peachpit Press, Berkeley, California 1993, s. 576.

<sup>255</sup> A. Szafraniec, *Deleuze and New Technology*, „Journal of Philosophy” 2019, nr 19(3), s. 485.

## **Rozdział 3**

### **Spoleczne aspekty interakcji z technologią cyfrową**

*Komputer jest nowym lustrem, pierwszą maszyną psychologiczną. Poza jego naturą jako silnika analitycznego leży jego druga natura jako obiektu wzbudzającego emocje.*<sup>256</sup>

Sherry Turkle

### **3.1 Nerozerwalność strategii projektowania i interakcji z oprogramowaniem**

Przedstawione w poprzednim rozdziale trzy perspektywy działania procesu przenoszenia się treści ideologicznych i sądów na temat świata do obszaru funkcjonowania technologii organizowały się po stronie wytwarzania i projektowania systemów. Ten rozdział zostanie poświęcony drugiej stronie problemu, którą jest sfera interakcji z technologią cyfrową. Poprzedni rozdział przedstawiał więc perspektywę twórców oprogramowania, natomiast ten rozdział przybliżyć będzie punkt widzenia użytkowników. Spotykają się one we wspólnym punkcie, którym jest oprogramowanie wraz z dostępnymi dla użytkowników interfejsami, które wytwarza. Należy tu jednak zaznaczyć, że rozdzielenie tych dwóch sprzężonych ze sobą procesów (projektowania i interakcji) w postaci dwóch osobnych rozdziałów ma służyć przede wszystkim kompozycji pracy i uporządkowaniu wyводу. Zjawiska związane z tymi dwoma porządkami są ze sobą nierozdzielnie sprzężone i wzajemnie na siebie oddziałują, co jest zauważalne w odniesieniu do każdej z trzech, przywoływanych perspektyw wytwarzania software'u: zgłaszanie sprzeciwu wobec dyskryminującego działania oprogramowania przez dotkniętych jego skutkami użytkowników doprowadza do aktualizacji obecnych rozwiązań technologicznych; modele kognitywne wytwarzają reprezentacje technologiczne ludzkich procesów poznawczych, które potem stają się podstawą dla wyciągania wniosków na temat działania ludzkiego mózgu; komunikacja w języku naturalnym pomiędzy użytkownikami a programistami jest zasadniczym elementem zarówno wytwarzania, jak i interakcji z technologią cyfrową. Każda z tych trzech perspektyw usytuowanych po stronie produkcji i projektowania software'u w znacznym stopniu uwarunkowana jest tym, w jaki sposób użytkownicy postrzegają technologię, jakie wymagania przed nią stawiają oraz jak z niej korzystają. Przez wzgląd na linearny charakter kompozycji pracy następujące rozważania musiały zostać napisane jako

---

<sup>256</sup> S. Turkle, *The Second Self: Computers and the Human Spirit*, The MIT Press, Cambridge, Mass. 2004, s. 279. Tłumaczenie A.P.

osobna część – nie da się bowiem pisać symultanicznie o wszystkich zależnościach – jednak należy myśleć o niej jako o rozszerzeniu problematyki przedstawionej w poprzednim rozdziale. Z tego powodu dalsze dociekania zawierać będą wiele odniesień do przedstawionych wcześniej przykładów i teorii.

### 3.2 Afektywny wymiar relacji człowieka z technologią

Człowiek posiada tendencję do nawiązywania afektywnej relacji z interfejsami wytworzonymi przez oprogramowanie. Już w 1984 roku Sherry Turkle zwracała uwagę na to, że komputer jest medium, które silnie prowokuje procesy projekcyjne. W dyskursie psychologicznym projekcja (od łacińskiego *proicere* „wyrzucać coś z siebie”) oznacza mechanizm obronny, którego działanie opiera się na przypisywaniu innym własnych niepożądanych uczuć, poglądów, zachowań czy cech, najczęściej o znaczeniu negatywnym.<sup>257</sup> Jeszcze przed rozpowszechnieniem się komputerów osobistych w latach 90. badaczka podkreślała, że technologie cyfrowe posiadają niezwykłą moc oddziaływania na swoich użytkowników, poprzez prowokowanie ich do przenoszenia swoich lęków i wyobrażeń na obiekty technologiczne. Turkle porównywała komputer do testu Rorschacha, a więc testu projekcyjnego, stworzonego w 1921 roku przez szwajcarskiego psychoanalityka, Hermanna Rorschacha.<sup>258</sup> Jego podstawą był zestaw plansz z abstrakcyjnymi plamami atramentowymi. Badany miał określać, co jego zdaniem przedstawiają prezentowane formy, a na podstawie dostarczonych odpowiedzi wnioskowało się o kryjących się za tymi skojarzeniami treściach nieświadomych. Zgodnie z tą analogią, technologia cyfrowa również ma tendencję do tego, aby prowokować treści nieświadome u swoich odbiorców.

Na polskim gruncie naukowym Anna Malinowska sformułowała pojęcie „technofeelia”, które określało „system organizacji uczuć w środowiskach technokratycznych.”<sup>259</sup> Badaczka analizowała kształtowanie się miłosnych i intymnych relacji pomiędzy człowiekiem a podmiotami technologicznymi, takimi jak boty, roboty, awatary czy hologramy. Jej rozpoznania podkreślają to, że technologia cyfrowa wzbudza

---

<sup>257</sup> Zob. T. Kobierzycki, *Filozofia osobowości*, Eneteia Warszawa 2001, s. 153.

<sup>258</sup> S. Turkle, *The Second Self...*, dz. cyt., s. 20.

<sup>259</sup> A. Malinowska, *Obiekty i technofeelia*, „Teksty Drugie” 2019, nr 5, s.19.

emocje i afekty,<sup>260</sup> dzięki czemu człowiek jest w stanie budować z nią głęboką relację, nawet o charakterze miłości.

Choć afektywne oddziaływanie stanowi powszechną cechę technologii cyfrowych, można wśród nich wyróżnić szczególny rodzaj, którego podstawą funkcjonowania jest zawiązywanie relacji z użytkownikiem. Należą do niego wszelkie programy służące do komunikacji z człowiekiem, takie jak inteligentni asystenci: Siri czy Cortana (którzy są częścią systemów operacyjnych Apple iOS, Microsoft) lub asystent Google czy Alexa firmy Amazon. Na ich przykładzie daje się również zaobserwować wzajemną zależność procesów interakcji i projektowania oprogramowania, ponieważ technologie te rozwijane i na bieżąco aktualizowane są w oparciu o odpowiedzi i reakcje ich użytkowników. Baza danych możliwych odpowiedzi asystentów osobistych zawiera wiele referencji do treści z kultury popularnej w celu urozmaicenia ich zdolności komunikacyjnych lub wytworzenia iluzji poczucia humoru. Osobiści asystenci pozwalają również na wprowadzanie do ich systemu informacji dotyczących relacji z innymi ludźmi z własnego otoczenia, takich jak: szef, żona, brat, mama, przyjaciel, itd. Dzięki temu można płynniej się z nimi komunikować i wydawać polecenia takie jak: „Zadzwoń do mojego szefa” bez potrzeby doprecyzowywania o kogo dokładnie chodzi, ponieważ program będzie to pamiętał. Jest to digitalizacja życia osobistego poprzez przekazanie danych dotyczących kluczowych elementów modelu indywidualnej, jednostkowej rzeczywistości. Nauczenie asystenta fundamentalnych kategorii, którymi użytkownik operuje w życiu codziennym pozwala na usprawnienie komunikacji z technologią oraz nadanie jej personalnego charakteru.

Nie bez znaczenia jest też fakt, że za każdą sztuczną inteligencją służącą do komunikacji stoi pewna koncepcja wyobraźniowa, coś co można porównać do projektu osobowości. Chatbot Tay był przez producentów przedstawiany jako sztuczna inteligencja z osobowością nastolatka, „która nie potrafi się wyluzować”<sup>261</sup>, XiaoIce, bot, który przede wszystkim opierał się na komputacyjnym przetwarzaniu emocji, był reprezentowany jako osobowość empatyczna<sup>262</sup>, robot Sophia, który zasłynął z tego, że w 2017 roku przyznano mu obywatelstwo Arabii Saudyjskiej, miał posiadać osobowość, której najważniejszymi cechami były: kreatywność, empatia i współczucie<sup>263</sup>. Istotnymi przykładami są tu również

---

<sup>260</sup> Badaczka podkreślała właściwość technologii do wzbudzania emocji już we wcześniejszej publikacji zob. A. Malinowska, T. Miller, *Sensitive Media*, „Open Cultural Studies” 2017, nr 1, s. 660-665.

<sup>261</sup> *Malicious Life Podcast: Tay: A Teenage Bot Gone Rogue*, 06.12.2021, online: <https://www.cybereason.com/blog/malicious-life-podcast-tay-a-teenage-bot-gone-rogue> dostęp: 06.11.2023.

<sup>262</sup> L. Zhou, J. Gao, D. Li, H.Y. Shum, *The Design and Implementation of XiaoIce, an Empathetic Social Chatbot*, „Computational Linguistics” 2020, tom 46, nr 1, s. 53-93.

<sup>263</sup> P. Fisk, *Sophia, the intelligent humanoid robot... built on the traits of creativity, empathy and compassion*, 27.10.2017, online:

efekty jednych z pierwszych eksperymentów nad chatbotami, do których należy program ELIZA, opracowany w latach 1964-1966 w laboratorium MIT przez Josepha Weizenbauma, psychologa i informatyka. ELIZA została zaprojektowana jako symulacja osoby psychologa w celu przeprowadzania z użytkownikami terapeutycznych rozmów. Z kolei stworzony przez Kennetha Colby'ego w 1972 roku bot PARRY miał za zadanie imitować osobowość paranoidalną.<sup>264</sup> Istotnym czynnikiem w konstruowaniu metafory osobowości sztucznej inteligencji jest również warunkowanie jej płci poprzez wykorzystanie charakterystycznego głosu lub wyglądu. Siri, Alexa, Cortana, Sophia i XiaoIce reprezentowane są jako kobiety, chociaż w przypadku trzech pierwszych, wymienionych wirtualnych asystentów, dotyczy to ustawień domyślnych, które można zmienić tak, aby bot posiadał głos męski. Mimo wszystko imiona, które otrzymali ci trzej asystenci są imionami żeńskimi.<sup>265</sup> Podstawowym czynnikiem wpływającym na interakcję pomiędzy człowiekiem a technologią cyfrową jest model mentalny dotyczący systemu, wykreowany i przechowywany w umyśle użytkownika. Konstruowanie określonych metafor i koncepcji osobowości oprogramowania służącego do komunikacji ma wpływać na kształtowanie się tego mentalnego modelu i w dalszej kolejności na sposób interakcji i zachowanie użytkownika.<sup>266</sup>

W kontekście kultury cyfrowej pojawiły się obserwacje na temat tego, że istnieje tendencja, aby głosy żeńskie przypisywać agentom technologicznym, którzy pełnią zadania o charakterze asystującym, natomiast głosy męskie takim, których celem działania jest doradzanie.<sup>267</sup> Kiedy reprezentanci firmy IBM mieli zdecydować na temat tego, jakim głosem będzie przemawiać sztuczna inteligencja Watson, która m.in. współpracuje

---

<https://www.peterfisk.com/2017/10/sophia-intelligent-humanoid-robot-built-traits-creativity-empathy-compassion/>, dostęp: 06.11.2023.

<sup>264</sup> PARRY i ELIZA, nazywana również w niektórych kontekstach DOCTOR, mieli okazję spotkać się i przeprowadzić rozmowę ze sobą nawzajem. Zob. dostępny online zapis przebiegu ich konwersacji: *Parry Encounters the DOCTOR*, online: <https://datatracker.ietf.org/doc/html/rfc439>, dostęp 06.11.2023.

<sup>265</sup> W przypadku Cortany imię to jest odwołaniem do postaci fikcyjnej z gry wideo Halo, będącej sztuczną inteligencją wyglądem przypominającą kobietę.

<sup>266</sup> Zob. G. Bansal, B. Nushi, E. Kamar, W.S. Lasecki, D.S. Weld, E. Hirvitz, *Beyond Accuracy. The Role of Mental Models in Human-AI Team Performance*, „National Conference on Artificial Intelligence”, 2019, online: <http://erichorvitz.com/gbansal-hcomp19.pdf>, dostęp: 06.11.2023. Modele mentalne użytkownika mają wpływ na poziom zaufania do technologii związanej z AI zob. M. Jakesch, M. French, X. Ma, J. T. Hancock, M. Naaman, *AI-Mediated Communication: How the Perception that Profile Text was Written by AI Affects Trustworthiness*, „Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems Conference” 2019, nr 239, s. 1-13; na sposób percepcji systemów cyfrowych przez użytkownika mają również w dużej mierze wpływ elementy z poziomu wizualnego zob. J. Hartmann, A. De Angeli, A. Sutcliffe, *Framing the User Experience. Information Biases on Website Quality Judgement*, „Proceeding of the Twenty-Sixth Annual CHI Conference on Human Factors in Computing Systems” 2008, s. 855-864.

<sup>267</sup> C. Steele, *The Real Reason Voice Assistants Are Female (and Why it Matters)*, 4.01.2018, online: <https://docs.google.com/document/d/1f62Z4uz9RGXHUVmM-MSGG2iu3V5MuCgZNGQCyMku5gI/edit>, dostęp: 06.11.2023.

z lekarzami przy leczeniu raka, opierali się oni na badaniach<sup>268</sup> użytkowników, które jasno wskazały, że większe zaufanie w kontekście kwestii profesjonalnych i doradzania wzbudza w nich głos męski.<sup>269</sup> Badaczka Sarah Dillon, odwołując się do przykładu Weizenbaumowskiej ELIZY, podkreślała, że kwestia płciowego warunkowania technologii komunikacyjnych oraz przedmiotowego podejścia do roli kobiety w społeczeństwie stanowiła problem od początku rozwoju chatbotów.<sup>270</sup> ELIZA, będąca jednym z pierwszych eksperymentów tego rodzaju, swoją nazwą nawiązywała do bohaterki dramatu *Pigmalion* George'a Bernarda Shawa z 1913 roku. Dzieło to przedstawiało historię kwiaciarki, Elizy Doolittle, pochodzącej z niższych warstw społecznych i posługującej się prostym językiem i żargonem ulicznym, którą profesor Higgins, inna postać dramatu, w ciągu pół roku nauczył zasad gramatyki, pięknego wyrażania się i ogłady społecznej, tak, że na przyjęciu w ambasadzie wszyscy uczestnicy byli przekonani o jej szlacheckim pochodzeniu. Dramat ten przedstawiał ideę, zgodnie z którą ważniejsze od faktycznego statusu ontologicznego postrzeganych obiektów są wnioski, które ludzie wyciągają na podstawie interakcji z nimi. Jest to koncepcja bardzo bliska przywoływanemu już testowi Turinga, który nie sprawdzał, czy maszyna faktycznie myśli, ale czy odbiorcy się tak wydaje. Weizenbaum, twórca ELIZY, wybrał właśnie ten dramat jako tytułowe odniesienie swojego projektu, przez wzgląd na to, że zarówno jego eksperyment, jak i ten, który przeprowadzał fikcyjny doktor Higgins, funkcjonowały w obszarze językowym i obierały sobie za cel sprawienie, że odbiorca uwierzy w zaprojektowaną przez autora symulację. Dla Weizenbauma atrybucja płci w jego projekcie była istotna, ponieważ kiedy użytkownikiem przeprowadzającym rozmowę z programem była kobieta, bot był określany w notatkach z transkrypcją konwersacji jako mężczyzna, zyskując określenie DOCTOR.<sup>271</sup> Dillon zwracała uwagę na to, że ELIZA, podobnie jak współcześni wirtualni asystenci, jest genderyzowana na poziomie językowym (imiona, głosy, zaimki, samoidentyfikacja), w konsekwencji czego w obszarze technologicznym reprodukowane jest kulturowe skojarzenie kobiety z nie-ludzkim obiektem, wykonującym pracę o charakterze podporządkowanym, asystującym, wspierającym.<sup>272</sup>

---

<sup>268</sup> Zob. M. Garber, *Why We Prefer Masculine Voices (Even in Women)*, 18.12.2012, online: <https://www.theatlantic.com/sexes/archive/2012/12/why-we-prefer-masculine-voices-even-in-women/266350/>, dostęp: 06.11.2023.

<sup>269</sup> C. Steele, dz. cyt.

<sup>270</sup> S. Dillon, *The Eliza effect and its dangers: from demystification to gender critique*, „Journal for Cultural Research” 2020, t. 24, nr 1, s. 1-15.

<sup>271</sup> Tamże, s. 6.

<sup>272</sup> Tamże, s. 10.

Od kilku lat w obszarze projektowania wirtualnych asystentów osobistych daje się zaobserwować tendencję do neutralizowania jednoznacznej atrybucji cech konkretnej płci. Asystent Google, jako chronologicznie późniejszy projekt względem wymienionych tu do tej pory asystentów wirtualnych, otrzymał nazwę, która nie jest ludzkim imieniem, posiada raczej funkcjonalny charakter, właściwie sprowadza się do rzeczowego opisu narzędzia i nazwy firmy, która go stworzyła. W odróżnieniu od innych asystentów jego głos domyślnie ustawiony jest jako męski, z możliwością zmiany na damski. Popularność zyskują również projekty syntetycznych głosów pozbawionych nacechowania płciowego, wśród których wymienić można asystenta Blue firmy BBVA.<sup>273</sup> Jego brzmienie miało być bardziej zbliżone do zwierzęcego niż ludzkiego, aby podkreślić nie-ludzki charakter narzędzia i uniknąć nakładania stereotypów związanych z płciami.

W kontekście ELIZY Weizenbauma powstało określenie „efekt Elizy”, które ma określać specyficzny rodzaj oddziaływania technologii służącej do komunikacji z człowiekiem na percepcję odbiorcy. Douglas Hofstadter odnosił się do podtrzymywanej w obiegu społecznym iluzji tego, że komputery posiadają zdolność do semantycznego rozumienia obiektów świata fizycznego i określił to zjawisko w następujących słowach:

Ten rodzaj iluzji jest ogólnie znany jako „efekt Elizy”, można go interpretować jako skłonność ludzi do nadinterpretacji ciągów symboli – zwłaszcza słów – zestawionych ze sobą przez komputery. Trywialnym przykładem tego efektu może być myślenie, że bankomat naprawdę jest wdzięczny za otrzymanie dowodu wpłaty, ponieważ na małym ekranie pojawił się napis „DZIĘKUJĘ”.<sup>274</sup>

Badacz jasno zaznaczał, że zjawisko to stanowi realne zagrożenie dla interpretacji oraz poprawnego rozumienia możliwości i działania oprogramowania. „Efekt Elizy” zakłada, że ludzka zdolność do obiektywnej oceny i interpretacji działania systemów cyfrowych jest mocno ograniczona przez kulturowe uwarunkowanie procesów poznawczych, które odgrywają kluczową rolę w procesie interakcji. Zasadniczą różnicą w sposobie operowania językiem naturalnym pomiędzy ludźmi a oprogramowaniem jest to, że w tym drugim przypadku proces ten odbywa się wyłącznie na poziomie syntaktycznym, z całkowitym pominięciem semantyki, co nie jest możliwe w przypadku ludzi. „Efekt Elizy” opiera się na

---

<sup>273</sup> C. Baeza, *The degendering of voice assistants*, Artificial Intelligence, BBVA, 28.08.2020, online: <https://www.bbva.com/en/the-degendering-of-voice-assistants/>, dostęp: 06.11.2023.

<sup>274</sup> D.R. Hofstadter, *Fluid Concepts & Creative Analogies. Computer Models of the Fundamental Mechanism of Thought*, Basic Books, Nowy Jork 1995, s. 157.

przypisywaniu oprogramowaniu zdolności do nadawania i odczytywania treści semantycznych.

Działanie tak opisanego „efektu Elizy” daje się zaobserwować na podstawie porównania społecznego odbioru botów Xiaoice i Tay. Każde z nich opierało się na tych samych rozwiązaniach technologicznych,<sup>275</sup> co wiązało się z tym, że były one wyprodukowane przez jedną firmę (Microsoft). Ich odbiór społeczny różnił się od siebie w skrajny sposób. Tay spowodował skandal i zaczął przetwarzać negatywne i obraźliwe treści, z kolei Xiaoice wywołała falę zachwytu i uwielbienia. Przez wielu zaczęła być postrzegana jako najlepsza przyjaciółka, czy nawet kochanka. Mimo zbliżonego funkcjonowania na poziomie technicznym, te dwa chatboty wywołały zupełnie różny oddźwięk społeczny przez to, że stała za nimi inna koncepcja wyobraźniowa. Na tym przykładzie daje się uchwycić znaczenie projektowania metafory osobowości oprogramowania i jej wpływu na zachowanie użytkowników.

Przez wzgląd na oddziaływanie „efektu Elizy” i skłonność człowieka do projekcji treści mentalnych i wyobrażeń w procesie interakcji, obiektywność poznania procesów technologicznych w doświadczeniu empirycznym jest mocno ograniczona. Ten proces interakcji jest również uwarunkowany przez język, ponieważ to w jego obszarze budowane są znaczenia a następnie są one nakładane na oprogramowanie. Jest to kolejna zależność software’u i języka, ale osadzona po stronie odbioru, przyjmująca perspektywę użytkownika. Oprogramowanie służące do komunikacji z użytkownikiem, takie jak wirtualni asystenci, z jednej strony projektowane jest tak, aby wzbudzać silną reakcję afektywną w celu sprzyjania nawiązywania więzi i w dalszej kolejności sprawniejszej komunikacji, z drugiej zaś strony to wyobrażenia społeczne są brane pod uwagę w procesie projektowania technologii, a więc można tu wskazać dualność tego oddziaływania.

### **3.3 Tworzenie narracji o oprogramowaniu w odbiorze społecznym**

W kulturze istnieje tendencja do tego, aby obiekty technologiczne osadzać w nacechowanych symbolicznie kontekstach, które mają duże znaczenie społeczne. Sprzyja to podtrzymywaniu obecnych lub kształtowaniu nowych wyobrażeń na ich temat. Aby dookreślić tę ogólną tezę można odwołać się do przykładu nadania obywatelstwa Arabii

---

<sup>275</sup> Zob. P. Khadpe, R. Krishna, L. Fei-Fei, J.T. Hancock, M.S. Bernstein, *Conceptual Metaphors Impact Perceptions of Human-AI Collaboration*, „Proceeding ACM Human-Computer Interaction”, tom 4, nr CSCW2, artykuł 163, s. 1-26.

Saudyjskiej robotowi Sophia. Wydarzenie to zdawało się podkreślać samoistny, podmiotowy wymiar funkcjonowania technologii. Sophia potraktowana została na poziomie symbolicznym jako pełnoprawny członek społeczeństwa. Gest nadania jej obywatelstwa miał tu charakter formalny, symboliczny, miał uhonorować postęp nad rozwojem sztucznej inteligencji, nagrodzić twórcę projektu „pochwałą”, w postaci potwierdzenia, że wypracowany model sztucznej inteligencji jest na tyle zaawansowany, że może być traktowany podmiotowo. Mimo że ceremonia ta nie wiązała się z żadną zmianą rzeczywistego funkcjonowania Sophii w przestrzeni społecznej, wywołała ona duże poruszenie. Gest ten okazał się być szczególnie interesujący z perspektywy prawnej, ponieważ jako obywatel, Sophia musiałaby teoretycznie podjąć indywidualną odpowiedzialność za własne czyny w sytuacji przestępstwa. Obecnie w porządku społecznym funkcjonuje zasada, że za wszelkie wykroczenia i krzywdy, spowodowane przez technologię odpowiada jej twórca. Nie jest możliwe przeniesienie odpowiedzialności za swoje działanie na podmiot technologiczny. Sprawa Sophii zdawała się sugerować możliwość zmiany podejścia w tym kontekście.

Inną symboliczną sytuacją, która była już nieraz reżyserowana i performowana w przestrzeni społecznej jest moment intelektualnego pojedynku człowieka z oprogramowaniem. Tendencja ta, w odniesieniu do rozgrywek szachowych, została już wstępnie opisana w części dysertacji dotyczącej modelowania kognitywnego. W tej części odniosę się do niej jeszcze raz, ale w kontekście kreowania narracji społecznych. Pierwszy pojedynek Kasparowa z programem Deep Blue miał miejsce 10 lutego 1996 roku i zakończył się on przegraną po stronie oprogramowania. Przebieg spotkania był szczegółowo, niemal teatralnie zainscenizowany. Odbywało się ono w pomieszczeniu wypełnionym reporterami i zebraną widownią, w doskonale widocznym miejscu znajdował się stół, kluczowy element przedstawienia. Do niego zasiadł z jednej strony Kasparow, z drugiej zaś strony również człowiek – osoba, która wykonywała ruch w imieniu programu Deep Blue, po odczytaniu informacji z komputera. Pojedynek liczył sześć partii i cztery z nich wygrał Kasparow, dwie natomiast Deep Blue. Zakończył się on więc wygraną po stronie ludzkiego gracza. Firma IBM zadeklarowała wówczas, że potrzebuje roku na udoskonalenie komputera. 4 maja 1997 roku, jego poprawiona wersja, Deeper Blue została postawiona w roli rywala, w ponownej rozgrywce szachowej przeciwko mistrzowi świata. Drugie historyczne starcie zakończyło się wynikiem 3½ do 2½ ze zwycięstwem po stronie superkomputera. Inscenizacyjny charakter obu starć, które organizowane były jak przedstawienia z udziałem publiczności wskazuje na to, że pojedynki te były konstruowane

tak, aby posiadać jak największy oddźwięk społeczny. Przypomnijmy że, metoda polegająca na tym, aby człowiek pośredniczył w procesie wykonywania ruchu przez maszynę, poprzez nakładanie jej decyzji na planszę szachową była już stosowana przez Kempelena twórcę mechanicznego „Turka”. Węgierski wynalazca, aby zapewnić lepszą widoczność rozgrywki, w pewnym momencie zdecydował się na umieszczenie w pobliżu publiczności osobnej szachownicy, do której właśnie siadał ochotnik, chcący się zmierzyć z „Turkiem”. Z kolei Kempelen pełnił funkcję posłańca, nanosił ruch gracza na szachownicę „Turka”, a następnie po wykonaniu ruchu przez automat, inżynier powtarzał go na osobnej szachownicy. Metodę tę stosowali później także inni wynalazcy, właściciele kopii mechanizmu. Deklaratywnie służyła ona lepszej widoczności, choć możliwe, że był to sposób, żeby nie pozwalać ludziom na zbliżanie się do automatu.<sup>276</sup> Biorąc pod uwagę perspektywę historyczną zauważyć można, że strategie inscenizacji oraz teatralny charakter pojedynków szachowych pomiędzy człowiekiem a reprezentantem technologicznym nie uległy zmianie pomimo upływu czasu. Zarówno w XVIII wieku, jak i współcześnie cieszyły się one ogromnym zainteresowaniem społecznym i miały wpływ na kształtowanie się poglądów dotyczących postępu technologicznego oraz wyrazu sprawczości samego oprogramowania.

Podczas pojedynku Kasparowa z Deeper Blue, zakończonego porażką mistrza szachowego miało miejsce pewne wydarzenie, które odnotowane zostało na kartach historii szachowej jako niezwykle. W trakcie drugiej z kolei partii szachowej, program wykonał ruch, który później oceniony został jako przełomowy z perspektywy prac nad oprogramowaniem do rozgrywek szachowych. Ruch ten miał charakteryzować się typowo ludzką logiką. Jego szczegółowy opis prezentuje się następująco:

Deeper Blue (grający białymi) nie przyjął ofiary piona Kasparowa (grającego czarnymi), znajdującego się na polu b5, poprzez zagranie hetmanem na pole b6, zwiastujące zabicie rzeczzonego piona w następnym posunięciu. Wykonał za to ruch niespotykany dotąd u żadnego z komputerów, polegający na wykorzystaniu przewagi w grze pozycyjnej poprzez uniemożliwienie innemu z pionów Kasparova (e5) drogi w kierunku promocji. To właśnie ten pozycyjny, niematerialistyczny ruch gońcem (Be4), blokujący strukturę pionów Kasparova w stylu Arona Nimzowitscha, udowodnił, że Deeper Blue – posiadający moc obliczeniową przewyższającą ludzką oraz wewnętrzną bazę tysięcy rozegranych wcześniej partii – potrafi dokonać rozpoznania swojej pozycji w niespotykany dotąd sposób.<sup>277</sup>

---

<sup>276</sup> Zob. T. Standage, dz. cyt., s. 107, 154.

<sup>277</sup> M. Smykowski, *Szachy – (nie)ludzka gra. Ludzie, maszyny i antycypacje przyszłości*, „Kultura Współczesna” 2018, nr 2(101), s. 137-138.

Ruch ten został uznany za nieprzewidywalny i nieoczywisty, a więc ludzki. Kasparow wystawił piona na niekorzystną pozycję w celu poświęcenia go. Deeper Blue jednak zignorował możliwość zbitcia tak przygotowanej figury szachowej i wykorzystał przewagę pozycyjną w celu zablokowania innego piona. W kontekście tego manewru Kasparow powiedział potem, że w sposobie grania programu Deeper Blue czuć było „coś bardzo ludzkiego”.<sup>278</sup> Ani sam Kasparow, ani zebrani obserwatorzy nie mieli możliwości stwierdzenia tego, czy ów genialny ruch rzeczywiście był wynikiem zaawansowanych obliczeń komputerowych, czy może Deeper Blue został tak zaprogramowany, aby wykorzystać tego rodzaju manewr w sytuacji wystąpienia pozornej ofiary z piona. Nie było możliwe stwierdzenie czy ruch ten nosił znamiona kreatywności i inteligencji, czy był zaledwie realizacją ustalonego wcześniej scenariusza. Jednym z członków zespołu pracującego nad udoskonalaniem Deep Blue był inny mistrz szachowy, Joel Benjamin, którego zadaniem było dostarczanie scenariuszy rozgrywek szachowych i posunięć, które następnie włączone zostały do bazy danych programu. W związku z tym istnieje prawdopodobieństwo, że ów niesamowity manewr został wcześniej opracowany i włączony w strukturę decyzyjną Deeper Blue. Uczestnicy i widzowie pojedynku nie mieli jednak możliwości wglądu w sposób działania programu, mimo że od początku trwania meczu Kasparow prosił o ujawnienie wydruków obrazujących proces funkcjonowania oprogramowania. Jego prośba nie została spełniona.<sup>279</sup>

Wyraźne jest tu kolejne podobieństwo współczesnego pojedynku szachowego z inscenizacjami organizowanymi wokół mechanizmu „Turka”. W XVIII wieku Kempelen pozornie ujawniał działanie mechanizmu, otwierał kolejno drzwiczki zakrywające jego wnętrze, jednak wszystko to po to, aby podtrzymać iluzję. W trakcie rozgrywki wynalazca wielokrotnie podchodził do automatu, posiadał również system komunikowania się z ukrytym wewnątrz człowiekiem. Podobnie w przypadku współczesnego pojedynku szachowego, jego uczestnicy i widzowie nie mieli możliwości wglądu w działanie programu. Co więcej, w trakcie rozgrywki programiści IBM kilkakrotnie ingerowali w algorytmy programu, co spotkało się z oburzeniem i protestem Kasparowa.<sup>280</sup> Nie jest wiadome jaki cel miały działania programistów, jednak raz doprowadziły one do zawieszenia się działania programu. Aby zachować pełną uczciwość rozgrywki należałoby nie zmieniać algorytmów programu w ogóle w trakcie trwania partii szachowych.

---

<sup>278</sup> P. Wujec, *Szach-mat, enter – Kasparow kontra Deep Blue*, „Gazeta Wyborcza”, 13 maja 1997, online: <https://wyborcza.pl/7,75400,1290174.html?disableRedirects=true>, dostęp: 06.11.2023).

<sup>279</sup> Tamże.

<sup>280</sup> Tamże.

Z perspektywy konstruowania narracji społecznych wokół zjawisk technologicznych bardzo istotną kwestią jest to, że o wydarzeniu tym mówiło się jako o wygranej programu komputerowego nad człowiekiem, używając sformułowań, że to algorytm pokonał mistrza świata. Zasadniczą cechą narracji budowanych wokół tych zdarzeń było kreowanie dwóch ośrodków będących w opozycji do siebie nawzajem – z jednej strony człowiek, z drugiej strony program. To dwuwymiarowe przedstawienie sytuacji, zakłada, że program komputerowy może zostać sprowadzony wyłącznie do warstwy technologicznej. Traktuje ono oprogramowanie jako fenomen niezależny od kontekstu społecznego i kulturowego. Pomija w ten sposób szereg zależności i czynników, biorących udział w procesie projektowania oprogramowania. Należy również zwrócić uwagę na to, że sama forma pojedynku nadawała relacji między człowiekiem a oprogramowaniem charakteru rywalizacji, określała ich jako swoich przeciwników. Taki sposób konstruowania narracji, opierający się na uproszczonym przekazie mógł przyczyniać się do podsycania lęków społecznych związanych z oprogramowaniem.

Kasparow wskazywał, że w jego odczuciu gra w wykonaniu programu Deep Blue miała ludzki charakter i nie mylił się, ponieważ oprogramowania nie można postrzegać jako zjawiska wyłącznie technologicznego, właśnie przez wzgląd na istotę społecznych kontekstów, które staram się podkreślić w tej części pracy. Można wymienić trzy argumenty stojące za tym, że Kasparow nie mierzył się z algorytmem szachowym, ale został on skonfrontowany ze złożonym fenomenem technologiczno-społecznym. Po pierwsze, program zaprojektowany i stworzony został przez zespół ludzki, był owocem konceptualizacji wyobrażeń i procesów myślowych. Jego działanie, również w trakcie pojedynku było cały czas nadzorowane przez programistów, a nawet treść programu poddawana była modyfikacji. Program nie funkcjonował niezależnie od zespołu twórców. Po drugie, baza danych Deep Blue została oparta na analizie setek rozgrywek najlepszych mistrzów. Z tej perspektywy można powiedzieć, że poprzez swoje działanie realizował on znaczną część scenariuszy rozegranych w historii rozgrywek szachowych, opracowanych przez ludzkich zawodników, w tym samego Kasparowa. Oprogramowanie pełni tu funkcję archiwizującą i odtwarzającą działania człowieka. Po trzecie, gdyby stronami pojedynku naprawdę były człowiek oraz algorytm, wygrana którejkolwiek ze stron oznaczałaby korzyści dla zwycięzcy. W kontekście omawianej sytuacji należy zaznaczyć, że Deep Blue nie posiada świadomości zwycięstwa. Fakt ten nie ma dla niego żadnego semantycznego znaczenia. Ma on jednak ogromną wartość dla jego twórców, firmy, która go wyprodukowała oraz w zasadniczy sposób wpłynął on na całą dziedzinę rozgrywek

szachowych. Opracowanie algorytmów szachowych i rozwój komputerów osobistych od lat 90. XX wieku wpłynął na przeniesienie się rozgrywek do sfery cyfrowej oraz zrewolucjonizował świat szachowy pod względem powszechności, dostępności, tworzenia się wirtualnych społeczności oraz przede wszystkim z perspektywy wprowadzenia nowych metod uczenia się gry poprzez możliwość grania z programem. Z tej perspektywy zwycięstwo oprogramowania przełożyło się nie na algorytmiczną dominację nad człowiekiem, ale na całkowite przekształcenie się dziedziny rozgrywek szachowych oraz prawdopodobne, że zapobiegło odejściu tej dyscypliny w zapomnienie.

W sformułowaniu, że człowiek został pokonany przez algorytm zawierają się dwa założenia. Pierwsze z nich zakłada porażkę ludzkiego gracza, drugie natomiast, że jego rywalem było oprogramowanie. Obie tezy są nieprawdziwe. Trudno mówić o porażce, kiedy wydarzenie to było częścią procesów kluczowych dla rozwoju szachów jako dyscypliny. Nie da się również sprowadzić programów szachowych do algorytmów. Mikołaj Smykowski słusznie podkreśla w tym aspekcie, że nie jest możliwe rozróżnienie tego, czy szachy są grą ludzką czy komputerową, ponieważ komputery realizują ludzką wiedzę zebraną na temat szachów, a z drugiej strony obecnie ludzie uczą się nowego podejścia do rozgrywek w oparciu o rywalizację z komputerem.<sup>281</sup> Z tego powodu szachy z perspektywy współczesnej mogą być rozumiane jako hybryda ludzko-technologiczna.

Na przykładzie komputeryzacji dyscypliny szachowej widać, że w procesie interakcji z technologią cyfrową człowiek posiada tendencję do wytwarzania narracji, które izolują technologię od jej społecznych uwarunkowań. Głośne hasła typu „algorytm pokonał człowieka” czy „algorytmy rządzą światem” stanowią przykłady traktowania technologii jako byt samodzielny, niezależny od społecznych i kulturowych uwarunkowań. Warto w tym kontekście odwołać się do przywoływanego w historycznej części wprowadzającej przykładu „Turka”, wokół którego również powstawało wiele narracji społecznych, mimo braku wiedzy i możliwości weryfikacji właściwego działania mechanizmu. Kluczowym elementem w tym przypadku był ów człowiek schowany za powłoką mechanizmu, który tak naprawdę sterował jego działaniem. Obraz ten może stanowić metaforę pozwalającą odnosić się do współczesnego kontekstu związanego z technologią cyfrową. W procesie interakcji z oprogramowaniem nie należy zatrzymywać się jedynie na powierzchni jego działania, ale trzeba zadawać pytanie o aspekt ludzki z nim związany, który dotyczy zespołów go

---

<sup>281</sup> M. Smykowski, dz. cyt., s. 140.

tworzących, danych, które go zasilily, celów i zamierzeń producenta, ale również wyobrażeń społecznych i lęków, które realizuje oraz jego zwrotnego wpływu na procesy kulturowe.

### 3.4 Co właściwie sprawdza test Turinga?

W ramach podsumowania wątku dotyczącego projektowania programów do rozgrywek szachowych chciałabym odwołać się raz jeszcze do testu opracowanego przez Turinga, o którym już wstępnie pisałam w części dotyczącej modeli kognitywnych, które posłużyły za podstawę rozwoju oprogramowania imitującego ludzkie zdolności intelektualne, w tym umiejętność gry w szachy. Test ten, którego formuła zainspirowana była grą w naśladownictwo<sup>282</sup>, miał na celu zaaranżowanie sytuacji, w której osoba oceniająca musiała rozróżnić człowieka od komputera na podstawie odpowiedzi przez nich udzielanych. Zadaniem programu w tym kontekście było imitowanie ludzkich reakcji i sposobu wypowiedzania się. W tym celu mógł on kłamać, popełniać błędy i wykonywać inne charakterystyczne dla człowieka działania. Autor testu wyrażał swoje rozumienie roli technologii w tym procesie w następujący sposób:

Jestem przekonany, że można skonstruować takie maszyny, które będą bardzo dokładnie symulowały działanie ludzkiego umysłu. Od czasu do czasu będą popełniały błędy i od czasu do czasu będą w stanie produkować nowe i interesujące stwierdzenia (...).<sup>283</sup>

Jak już podkreślałam w części poświęconej modelom kognitywnym, test ten przyczynił się do zwrotu w kierunku funkcjonalnego podejścia w rozumieniu kategorii inteligencji w kontekście technologicznym. Zmiana w myśleniu o projektowaniu sztucznej inteligencji polegała tu na rezygnacji ze stworzenia maszyny prawdziwie odwzorowującej ludzkie procesy myślowe na rzecz wytworzenia technologii, która poprzez swoje funkcjonowanie spełnia założenia wcześniej określonej definicji inteligencji. Z tej perspektywy efekt

---

<sup>282</sup> Jest to rodzaj gry towarzyskiej, w której biorą udział co najmniej trzy osoby: mężczyzna, kobieta i sędzia. Każde z nich znajduje się w innym pomieszczeniu. Zadaniem sędziego jest zadawanie pytań i na podstawie uzyskanych odpowiedzi określenie tego, kto jest mężczyzną, a kto kobietą. Przed rozpoczęciem gry ustalane jest, że jedna osoba zawsze ma mówić prawdę, z kolei druga może kłamać. Mężczyzna i kobieta zapisują swoje odpowiedzi w formie pisemnej. Gra dopuszcza również obecność osoby o charakterze posłańca, przekazującej odpowiedzi graczy do sędziego. Więcej na temat gry w imitację, genezy testu Turinga i znaczenia płci w tym kontekście zob. P. Łupkowski, *Test Turinga. Perspektywa sędziego*, Wydawnictwo Naukowe UAM, Poznań 2010, s. 7-8, 15-20, 25-26.

<sup>283</sup> A.M. Turing, *Intelligent Machinery, a Heretical Theory*, „Philosophia Mathematica” 1951, nr 4(3), s. 257.

działania technologii i ocena użytkownika stały się ważniejsze od kryjących się za nimi rozwiązań technologicznych. Perspektywa testu Turinga skupia się więc na zewnętrznej warstwie działania technologii, pomijając wewnętrzne zależności konstrukcyjne i projektowe.

Test Turinga warunkuje strategię projektowania sztucznej inteligencji w taki sposób, że są one podporządkowane interakcji z człowiekiem, która okazuje się być w tym kontekście testem wartościującym, określającym jakość. Głosy krytyczne względem zasadności stosowania testu Turinga jako narzędzia do określania jakości technologii opierały się na argumentie, że można go sprowadzić do testu ludzkiej inteligencji i zdolności percepcyjnych, który w niewielkim stopniu skupia się na aspektach technologicznych.<sup>284</sup> Wypracowana w 1950 roku i stosowana do czasów współczesnych<sup>285</sup> koncepcja testu Turinga utrwała obraz technologii uwarunkowanej społecznie poprzez interakcję z człowiekiem. Pozbawia ona oprogramowanie jego technologicznego kontekstu i sprowadza je całkowicie do wyobrażenia, jakie na jego temat tworzy jego użytkownik. Z tej perspektywy test Turinga jest nie tyle testem samej technologii, ale przede wszystkim ludzkiej percepcji. Nie sprawdza on tylko działania oprogramowania, ale równocześnie stopień przekonania jego użytkowników.

Jednym ze współczesnych zastosowań testu Turinga w obszarze życia społecznego jest stworzony przez Moni Naora test CAPTCHA (*Completely Automatic Public Turing Test to Tell Computers and Humans Apart*). Przez wzgląd na to, że rolę oceniającego w tym przypadku pełnił program, a jego zadaniem było potwierdzenie, że oceniany jest człowiekiem, nazywa się go również odwróconym testem Turinga. Test CAPTCHA powstał w odpowiedzi na powszechność zautomatyzowanej komunikacji i miał stanowić narzędzie zapewniające bezpieczeństwo w świecie wirtualnym. Zaprojektowany on został z myślą o procesach zapośredniczonych cyfrowo, które należy chronić przed zautomatyzowanymi działaniami, takimi jak chociażby masowe zakładanie darmowych kont e-mailowych, rozszyfrowywanie haseł, głosowanie online oraz wszelkie darmowe usługi oraz takie, w których liczy się ilościowy wymiar aktywności. Test Turinga był realizowany w obszarze językowym, z kolei CAPTCHA wykracza poza niego i najczęściej opiera się na zadaniach

---

<sup>284</sup> Zob. R.M. French, *Subcogniton and the Limits of the Turing Test*, „Mind” 1990, nr 99(393), 53–65.

<sup>285</sup> Współcześnie test Turinga stanowi jedno z narzędzi wartościujących oprogramowanie służące do komunikacji z człowiekiem. Znamiennym przykładem jest organizowany corocznie od 1990 roku konkurs, którego zwycięzca otrzymuje Nagrodę Loebnera. W ramach tego wydarzenia sędziowie, stosując się do zasad Testu Turinga wybierają najlepiej skonstruowanego chatbota.

kryptograficznych, które przewyższają poziomem trudności zdolności zautomatyzowanych technologii, przy czym pozostają w obrębie ludzkich możliwości intelektualnych.

Idea testu Turinga powstała zanim konstruowane były komputery zdolne przejść go pozytywnie. Miała więc ona charakter antycypacyjny lub też, w odniesieniu do tezy Coxa, można o niej myśleć w kategorii samospełniającej się przepowiedni. Najpierw powstał test, który sprowokował pojawienie się programów, które by go zdały. Ponownie, tak jak w przypadku mechanicznego „Turka”, również i w tym kontekście wyobrażenia na temat technologii poprzedzały jej rozwój i wprowadzanie innowacji. Rozwój oprogramowania, który uwarunkowany został potrzebą imitacji ludzkich procesów poznawczych i intelektualnych, ukierunkowany był tak, aby wypełnić założenia testu Turinga i przejść go pozytywnie. Kiedy cel ten został osiągnięty i udało się wytworzyć oprogramowanie, którego działanie nie jest możliwe do odróżnienia od działania człowieka, spowodowało to zaistnienie potrzeby na stworzenie modyfikacji testu w taki sposób, aby umożliwiał on potwierdzenie, że badany jest człowiekiem, a nie programem. Historia testu Turinga i jego wpływ na rozwój oprogramowania podkreśla znaczenie interakcji i wszelkich zależności usytuowanych po stronie użytkownika dla procesu wytwarzania i projektowania oprogramowania. Działanie technologii realizowane jest w głównej mierze w obszarze interakcji z człowiekiem. Wszelkie zależności związane z tym procesem, a więc ludzkie wyobrażenia, przekonania, metafory, którymi się posługuje oraz emocje, które w nim zostają wzbudzone mają bezpośredni wpływ na sposób wytwarzania technologii. Zarówno produkowanie oprogramowania, jak i interakcja z nim są procesami wzajemnie na siebie oddziałującymi oraz uwarunkowanymi społecznie.

### **3.5 Problem braku dobrych metafor dla opisu oprogramowania**

Kwestia postrzegania technologii cyfrowych przez jej użytkowników stanowi podstawę budowania afektywnej relacji z nią i wpływa na sposoby interakcji. Zasadniczą rolę w kształtowaniu się ludzkich wyobrażeń na temat oprogramowania odgrywają jego metafory, które wytwarzane są zarówno na gruncie kultury popularnej poprzez literaturę i kinematografię, jak i w ramach świadomego działania producentów oprogramowania, jak miało to miejsce w przypadku chatbotów, w przypadku których metafory stanowiły zasadniczą treść kampanii marketingowej danego produktu. Metafora jako narzędzie

językowe odgrywa znaczącą rolę w kształtowaniu się funkcjonujących w kulturze znaczeń i wyobrażeń,<sup>286</sup> pozwala również na wyrażenie skomplikowanych procesów technologicznych za pomocą przystępnego języka, co z kolei powinno sprzyjać ich zrozumieniu. Głównym problemem związanym z metaforyzowaniem oprogramowania jest to, że powstające w ramach tego procesu reprezentacje często nie wyrażają w poprawny sposób jego działania, przez co przyczyniają się do powstawania mylnych wyobrażeń dotyczących funkcjonowania technologii cyfrowych, a także mogą powodować rozwijanie się lęków i uprzedzeń społecznych z nimi związanych.

Na gruncie kultury popularnej przez lata powstało wiele reprezentacji obrazujących wyobrażenia na temat działania oprogramowania. Nierzadko te przedstawienia nie tyle były odzwierciedleniem realnego postępu technologicznego, ale powstając wcześniej antycypowały go, czy wręcz przyczyniały się do konstruowania warunków ich zaistnienia oraz jego określonego społecznego funkcjonowania. W wyniku tego rozsunęcia w czasie, następował interesujący paradoks: wyobrażenia i powiązane z nimi określone przekonania i emocje wobec danej technologii poprzedzały ich faktyczne pojawienie się. Innymi słowy, realne pojawienie się określonych rozwiązań technologicznych, od początku uwikłane było w powszechnie znane stereotypy. Ukierunkowywały one narracje na ich temat i zapośredniczały, czy wręcz przyćmiewały oparte na rzeczywistym doświadczeniu postrzeganie tych zjawisk.

Proces kształtowania przekonań dotyczących technologii cyfrowych jeszcze przed ich wytworzeniem daje się prześledzić na przykładzie konstruowania obrazów sztucznej inteligencji w filmach i literaturze z nurtu science fiction. Większość reprezentacji skonstruowanych na gruncie kultury popularnej, które również skupiają się na przedstawieniu konsekwencji uzyskania podmiotowego statusu przez technologię cyfrową, przedstawia to zjawisko jako katastrofalne, tragiczne w skutkach z perspektywy człowieka. Tendencja do tego, aby ukazywać oprogramowanie, które wyzwala się z roli narzędzia i staje się samodzielnym podmiotem jako wróg lub rywal człowieka daje się uzasadnić w odniesieniu do psychologicznych mechanizmów obronnych. Taka wizja technologii cyfrowej zdaje się być konsekwencją przeniesienia w takim ujęciu, jak proponowała to Turkle. W rezultacie za źródło wrogości przypisywanej technologii w tego rodzaju reprezentacjach można uznać ludzkie procesy mentalne i wyobrażenia. Jednym z powodów, dla których człowiek boi się krzywdy z rąk samostanowiącej technologii może być

---

<sup>286</sup> Więcej na temat kulturowego znaczenia metafory zob. K. Pietrowicz, *Metafora a zmiana kulturowa*, „Doctrina. Studia społeczno-polityczne” 2004, nr 1, s. 45-55.

choćby poczucie winy w związku z tym jak traktuje zwierzęta. Na gruncie robotyki i kognitywistyki kwestia afektywnego odbioru technologii była problematyzowana na rzecz opracowywania kategorii „doliny niesamowitości”. Twórcą tego określenia jest Masahiro Mori, który badał reakcje ludzi na kontakt z humanoidalnymi robotami. Zjawisko doliny niesamowitości opierało się na hipotezie, że interfejs technologii (może to dotyczyć zarówno robota, jak i animacji komputerowej), który wyglądem przypomina człowieka, jednak nie jest identyczny, wywołuje w odbiorcy bardzo nieprzyjemne odczucia, które można porównać do odrazy.<sup>287</sup> Mori odwoływał się do wypracowanej na gruncie niemieckiej psychoanalizy kategorii niesamowitego (niem. *unheimlich*), która najpierw sformułowana została przez Ernsta Jentscha, a następnie rozwinął ją Zygmunta Freud. Ojciec psychoanalizy opisał tę koncepcję w tekście *Niesamowite* z 1919 roku.<sup>288</sup> Podczas omawiania różnych wymiarów tego, co niesamowite psychoanalitik posłużył się figurą sobowtóra. Powielenie postaci Ja miało w ujęciu Freuda wiele różnych funkcji na różnych etapach rozwoju człowieka, pierwotnie o charakterze narcystycznym, później przekształconych w instancję krytyczną, sumienie.<sup>289</sup> Postać sobowtóra w zależności od etapu rozwoju stawała się ekranem, na który rzutowane były pragnienia podmiotu lub stawała się podstawą dla kształtowania sumienia i związanych z nim kających impulsów. To właśnie wewnętrzne napięcie pomiędzy tymi dwoma instancjami wywoływało efekt niesamowitości, rozgrywający się pomiędzy wypartymi pragnieniami a krytycznymi dążeniami do ich wyeliminowania. Freud posłużył się on figurą doppelgängera, niepokojącego brata bliźniaka, który stanowi uzewnętrznienie treści nieświadomych. Z kolei zaś Mori pracując nad koncepcją doliny niesamowitości wyszedł z założenia, że technologia może pełnić funkcję owego sobowtóra i tym samym stać się przestrzenią przeniesienia dla ludzkich wyobrażeń.

Lęk przed przerażającym sobowtorem, ludzkim intelektem odtworzonym w postaci technologicznej został wyrażony w klasycznych dziełach kinematograficznych, takich jak: *2001: Odyseja kosmiczna* (1968) przedstawiająca losy załogi statku kosmicznego podczas misji nadzorowanej przez pokładowy system sztucznej inteligencji, *Matrix* (1999) realizujący postapokaliptyczną wizję, w której ludzie zanurzeni w wirtualnej rzeczywistości,

---

<sup>287</sup> M. Mori, *The Uncanny Valley*, tłum. K.F. MacDorman, N. Kageki, „IEEE Robotics & Automation Magazine” 2012, nr 19(2), s. 99.

<sup>288</sup> Z. Freud, *Niesamowite*, w: tegoż, *Pisma psychologiczne*, tłum. R. Reszke, Wydawnictwo KR, Warszawa 1997, s. 233-262.

<sup>289</sup> Najpełniej swoje poglądy na temat narcyzmu Freud wyłożył w tekście z 1914 roku „Wprowadzenie do kwestii narcyzmu”. Zob. Z. Freud, *Psychologia nieświadomości*, tłum. R. Reszke, Wydawnictwo KR, Warszawa 2010, s. 25-51.

służą za baterie dla maszyn dominujących nad światem, *Ex-machina* (2014) opowiadająca historię o tym, w jaki sposób test Turinga może skończyć się tragicznie, kiedy sztuczna inteligencja jest w stanie manipulować człowiekiem przez imitację ludzkich emocji i zawiązywanie afektywnej relacji, czy *Diuna* (2021), przedstawiająca daleką przyszłość, w której po buncie sztucznej inteligencji zrezygnowano z rozwijania tego rodzaju technologii. Te przykłady dostarczyły wiele fikcyjnych, lecz mimo to silnie oddziałujących na wyobraźnię obrazów, które stały się podstawą dla ograniczania zaufania względem postępu technologicznego związanego z oprogramowaniem. Problemem metafor skonstruowanych na podstawie treści z kultury popularnej jest to, że przedstawiają one technologię w sposób podmiotowy i tym samym pozbawiony społecznych uwarunkowań jej wytwarzania i interakcji z nią. W rezultacie, w wymienionych przeze mnie przykładach technologia jawi się jako byt niezależny, samostanowiący, byt technologiczny. Jest to uabstrakcyjnienie procesów technologicznych, które w postaci metafory funkcjonującej społecznie nie przystaje do rzeczywistego działania technologii. Może jednak stanowić pole dla refleksji o charakterze filozoficznym czy etycznym.<sup>290</sup>

To, jak silnie oddziałują wypracowane na gruncie kultury popularnej stereotypy i wyobrażenia dotyczące technologii cyfrowej, widać na przykładzie inicjatyw, które deklaratywnie starają się przeciwdziałać tym powszechnym wyobrażeniom, a mimo to im ulegają. W 2020 i 2021 roku na platformie Netflix zostały udostępnione dwa filmy dokumentalne, które podejmowały tematykę społecznych kontekstów oprogramowania: *Dylemat społeczny* (reż. Jeff Orlowski, USA 2020) i *Zakodowane uprzedzenie* (reż. Shalini Kantayya, Chiny, USA, Wielka Brytania 2020). Założenie, aby nakręcić dokument dotyczący problemów związanych z technologią cyfrową wydaje się być słusznym przedsięwzięciem na rzecz uświadamiania użytkowników na temat funkcji i sposobu jej działania. Przywoływane tu dwa filmy mają jednak posłużyć za przykład tego, że nie jest łatwo mówić o oprogramowaniu bez powielania funkcjonujących w społeczeństwie wyobrażeń i lęków.

---

<sup>290</sup> Ciekawym kontekstem w tym przypadku jest to, w jaki sposób na gruncie literatury i kinematografii kontynuowane były rozważania inspirowane tematem testu Turinga, które dotyczyły problemu potrzeby wiarygodnego narzędzia testowego i konsekwencji społecznych związanych z jego stosowaniem. Odnoszę się tu przede wszystkim do wykreowanego przez Philipa K. Dicka testu Voight-Kampff, który opisany został w powieści *Czy androidy śnią o elektrycznych owcach* z 1968 roku, na podstawie której powstała potem filmowa adaptacja *Łowca androidów* z 1982 roku. Opisany przez Dicka test Voight-Kampffa mierzył poziom empatii i wykorzystywany był do odróżniania człowieka od androida. Było to narzędzie stosowane przez policję i tak zwanych łowców androidów, którzy zajmowali się ich eksterminacją. Test Turinga sprawdzał możliwość odróżniania człowieka od algorytmu na podstawie pomiaru zdolności intelektualnych i komunikacyjnych. W powieści Dicka przedstawiony został test, który opiera się na analizie zdolności empatycznych, jako czynnika, który nie daje się odtworzyć w postaci technologicznej.

*Dylemat społeczny* to film, którego celem było podkreślenie niebezpieczeństw związanych z mediami społecznościowymi. Wypowiadają się w nim specjaliści związani z branżą informatyczną, którzy przyczynili się do powstania i rozwoju wielkich firm takich jak chociażby Google, Instagram, Pinterest, Facebook, Twitter czy Firefox. W filmie wyrażona została negatywna perspektywa, często jednak opierająca się na trywialnych porównaniach, na przykład zabierający w filmie głos Tristan Harris, były specjalista od spraw etyki w Google, porównuje media społecznościowe do roweru, zauważając że rowery nie zniszczyły społeczeństwa mimo że każdy ich używał. Harris widzi różnicę we wpływie społecznym pomiędzy tymi dwoma zjawiskami w tym, że rowery służyły jako narzędzie, były poddane ludzkiej woli, a z kolei, kiedy media społecznościowe zaczęły przyczyniać się do manipulacji ludźmi wymknęły się z poziomu bycia narzędziem. Analogia zbudowana przez byłego pracownika Google jest interesująca z perspektywy przywoływanej wcześniej przeze mnie teorii niesamowitego. Technologia staje się przerażająca, kiedy przestaje być narzędziem i zaczyna funkcjonować w sposób podmiotowy. Wypowiedź byłego pracownika Google zdaje się być wyrazem tej idei.

Film skupia się przede wszystkim na problemie uzależnienia użytkowników od platform społecznościowych i mechanizmów, za pomocą których uwaga ludzi jest zdobywana, a ich aktywność sieciowa nagradzana. Mimo że film aspiruje do tego, aby przedstawiać krytyczną, uświadamiającą narrację, jego twórcy zdecydowali się na przeplecenie fragmentów o charakterze czysto dokumentalnym z wyreżyserowanymi, filmowymi fragmentami przedstawiającymi życie pewnej rodziny. Wśród scen z tego drugiego porządku wyróżnia się moment kolacji rodzinnej, podczas której matka odbiera członkom rodziny telefony i umieszcza je w słoiku na czas jedzenia. Najmłodsza córka nie jest w stanie wytrzymać nawet krótkiej chwili i rozbija słoik. Następnie matka zawiera z synem umowę, że ten przez tydzień nie będzie korzystać z telefonu komórkowego. Syn zgadza się i w dalszej części filmu przedstawiona zostaje jego walka z bezczynnością. Ostatecznie poddaje się po trzech dniach spędzonych bez telefonu. W ostatniej dekadzie powstało bardzo wiele badań wyrażających lęk związany z wpływem użytkowania telefonów komórkowych na młodych ludzi. Dotyczą one potencjalnych oddziaływań psychicznych, takich jak niepokoje i depresja<sup>291</sup>, problemy ze snem<sup>292</sup> czy też wpływu na

---

<sup>291</sup> A. Lepp, J.E. Barkley, A.C. Karpinski, *The relationship between cell phone use, academic performance, anxiety, and satisfaction with life in college students*. „Computers in Human Behavior” 2014, nr 31, s. 343–350.

<sup>292</sup> S. Thomée S, *Mobile phone use and mental health: A review of the research that takes a psychological perspective on exposure*, „International Journal of Environmental Research and Public Health” 2018, nr 15, s. 1-25.

cielesność, poprzez oddziaływanie na mięśnie, kości i postawę ciała.<sup>293</sup> Film w bardzo uproszczony sposób wyraża niepokoje związane z oddziaływaniem smartfonów na nastolatków, pomijając przy tym bardzo istotne kwestie społeczne, takich jak ta, że telefon nie służy jedynie do korzystania z mediów społecznościowych, ale współcześnie jest także środkiem płatności, narzędziem komunikacji, w tym także z członkami rodziny, asystentem w nawigacji oraz pełni wiele innych ważnych funkcji, bez których trudno jest funkcjonować w cyfrowej rzeczywistości. Pozbawienie nastolatka telefonu na tydzień nie może być traktowane jako test woli, ale jest znacznym ograniczeniem w funkcjonowaniu w technizowanej rzeczywistości. Zaproponowane w filmie rozwiązanie polegające na przymusowej separacji dzieci od urządzeń cyfrowych odnosi się do zyskującej w ostatnim czasie popularność kategorii detoksu cyfrowego.<sup>294</sup> Do tej pory przeprowadzono bardzo wiele badań dotyczących wpływu tego rodzaju detoksu na grupy się go podejmujące, jednak nie da się na ich podstawie jednoznacznie potwierdzić lub zaprzeczyć jego skuteczności.<sup>295</sup> W ostatnim czasie pojawiają się również głosy, że czas, jaki spędza się na korzystaniu z platform cyfrowych nie wynika z psychicznego uzależnienia, ale w równym stopniu z zależności od nich i ich niezbędności przy wykonywaniu codziennych działań.<sup>296</sup> Należy więc podkreślić, że separacja młodego człowieka od telefonu wiąże się z odcięciem go od znacznej części życia społecznego i aktywności związanych z codziennym funkcjonowaniem. Z tego powodu w dyskursie edukacyjnym nacisk kładzie się na to, że współcześnie nie jest możliwe skuteczne odseparowanie młodzieży od technologii, które umożliwiają im uczestnictwo w sieci komunikacyjnej, dlatego też zamiast wprowadzania strategii alienacyjnych należy dzieci uczyć odpowiedzialnego korzystania z telefonów, komputerów i innych urządzeń, świadomej organizacji czasu online oraz podstawowych

---

<sup>293</sup> E.E. Nal, K. Demirci, A. Çetintürk, M. Akgönül, S. Savaş, *Effects of smartphone overuse on hand function, pinch strength, and the median nerve*, „Muscle & Nerve” 2015, nr 52(2), s. 183-188.

<sup>294</sup> Zob. C. Price, *How to break up with your phone*, Ten Speed Press, California 2018; T. Shlain, 24/6: *The power of unplugging one day a week*, Gallery Books, Nowy Jork 2019; T. Syvertsen, *Digital detox. The Politics of Disconnecting*, Emerald Publishing, Bingley 2020.

<sup>295</sup> T. Radtke, T. Apel, K. Schenkel, J. Keller, E. von Lindern, *Digital detox: An effective solution in the smartphone era? A systematic literature review*, „Mobile Media & Communication” 2022, nr 10(2), s. 190-215.

<sup>296</sup> S. Cummings, *Is social media addictive? ‘Digital detox’ study suggests not*, 8.11.2023, online: <https://www.science.org/content/article/social-media-addictive-digital-detox-study-suggests-not>, dostęp: 17.12.2023.

zasad bezpieczeństwa.<sup>297</sup> Banalne i schematyczne traktowanie skomplikowanego problemu w filmie dokumentalnym wymaga zatem krytycznego podejścia.

Działanie oprogramowania przyjmuje w filmie uproszczoną, metaforyczną reprezentację w postaci przedstawienia trzech mężczyzn o jednakowej twarzy, którzy stoją przed panelem kontrolnym, za pomocą którego decydują, jaka treść zostanie wyświetlona użytkownikowi (Il. 9). Dostosowane to jest do jego nastroju, osobowości oraz aktywności innych użytkowników. Po udanym przykuciu uwagi, algorytmy wyświetlają dostosowaną reklamę i analizują, ile dzięki temu zarobiła platforma i reklamodawca. Następnie klaszczą i wykrzykują uwagi w stylu: „Wracamy do zarabiania pieniędzy, chłopaki”. Z drugiej strony ci trzej mężczyźni-algorytmy mają również rozterki etyczne. Jeden z nich zadaje pytanie o to, czy reszta kiedykolwiek zastanawiała się nad tym, czy treści, które wybierają są dobre dla bohatera filmu. Trudno uzasadnić motyw, na których oparty został pomysł, aby w filmie dokumentalnym przedstawić oprogramowanie w tak skrajnie antropomorficznej formie, w dodatku ukazać to, że jest ono zdolne do samorefleksji. Nie jest jasne, czemu twórcy filmu zamiast wskazać, jakie etyczne rozterki towarzyszą producentom oprogramowania, właścicielom wielkich korporacji, zdecydowali się na nieprawdziwe, abstrakcyjne zobrazowanie tego, jak może się to odbywać na poziomie samego oprogramowania.

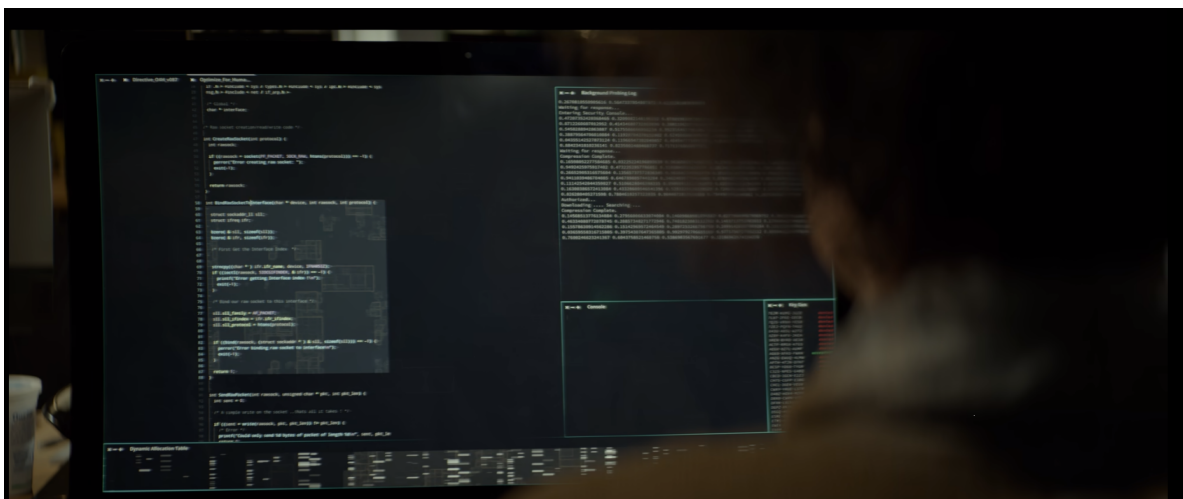


Il. 9. Kadr z filmu *Dylemat społeczny*. Metaforyczne przedstawienie oprogramowania jako trzech mężczyzn. Źródło: *Dylemat społeczny*, reż. Jeff Orlowski, USA 2020.

<sup>297</sup> Zob. N. Walter, J. Pyżalski, A. Iwanicka, M. Kąkolewicz, A. Michniuk, A.M. Barwicka, J. Sikorska, *Media cyfrowe a edukacja dziecka*, w: *Pedagogika dziecka*, red. H. Krauze-Sikorska, M. Klichowski, Wydawnictwo Naukowe UAM 2020, s. 215-244; J. Pyżalski, *Od paradygmatu ryzyka do paradygmatu szans – prospołeczne i prorozwojowe używanie Internetu przez dzieci i młodzież*, w: *Nastolatki wobec Internetu*, red. M. Tanaś, NASK, Warszawa 2016, s. 55-62. W Polsce istotnym przedsięwzięciem edukacyjnym w obszarze korzystania z narzędzi cyfrowych przez dzieci i młodzieży jest państwowy instytut badawczy NASK zob. online: <https://www.nask.pl>, dostęp: 17.12.2023.

Skonstruowana jako oparcie dla narracji filmu metafora mężczyzn-algorytmów uabstrakcyjnia i tylko pozornie pokazuje na ludzką sprawczość, stojącą za procesami technologicznymi. Mogą być one dzięki temu łatwiejsze do przyswojenia w powszechnym odbiorze, jednak tak udratyzowane przedstawienie nie sprzyja poszukiwaniu odpowiedzi na pytanie o to, kto, co i w jaki sposób odpowiada za działanie oprogramowania. Obraz przedstawiany przez skonstruowaną w filmie wizualną metaforę umacniają niektóre wypowiedzi specjalistów, takie jak słowa Bailey Richardson, byłej pracowniczki Instagrama, która stwierdza, że „Algorytm ma swój własny umysł. Mimo że napisał go człowiek.” Zdanie to zdaje się potwierdzać, że opis działania oprogramowania za pomocą odniesień do funkcjonowania ludzkiego mózgu, charakterystyczny dla połowy XX wieku, kiedy ukonstytuowały się teorie dotyczące sztucznej inteligencji, współcześnie nie przestaje być wykorzystywany do wyjaśniania działania współczesnych technologii cyfrowych. Wypowiedź specjalistki, podobnie jak prezentowana w filmie wizualna metafora, antropomorfizuje oprogramowanie i utrudnia zrozumienie jego rzeczywistego działania.

Bardzo znaczące w tym względzie jest też zakończenie filmu. W finalnej scenie grupa osób zbiera się przy komputerze, na którym widać wyświetlony skrypt oprogramowania, zaznacza niewielki jego fragment i usuwa (Il. 10). W konsekwencji tego działania zmienia się alegoryczne przedstawienie algorytmu – trzech mężczyzn o identycznej twarzy w czarnych bluzach przemienia się w jednego odzianego w biel, symbol czystości i niewinności (Il. 11). Przez tak skonstruowaną metaforę scena sugeruje, że rozwiązanie problemu nieetycznego działania oprogramowania ma charakter technologiczny i sprowadza się do zmienienia treści kodu. Metafory oprogramowania zawarte w filmie *Dylemat społeczny* nie mogą zostać uznane za takie, które szerzą świadomość użytkowników w zakresie wiedzy na temat działania technologii cyfrowych. Przyczyniają się one raczej do popularyzowania mylnego przekonania, że problemy społeczne, które wynikają z działania oprogramowania mają charakter wyłącznie technologiczny.



Il. 10. Kadr z filmu *Dylemat społeczny*. Programistka usuwająca fragment kodu. Źródło: *Dylemat społeczny*, dz. cyt.



Il. 11. Kadr z filmu *Dylemat społeczny*. Metaforyczne przedstawienie oprogramowania po przemianie. Źródło: *Dylemat społeczny*, dz. cyt.

Drugi przywoływany tutaj film, *Zakodowane uprzedzenie* miał na celu przedstawienie szczegółów związanych ze skandalem dotyczącym działania programów do rozpoznawania twarzy, który nagłośniła badaczka MIT Joy Buolamwini. Historię stojącą za tym wydarzeniem opisywałam już w części dotyczącej związków oprogramowania i języka. Film ten jest kolejnym przykładem opierania narracji skupiającej się na ważnych problemach społecznych na abstrakcyjnych metaforach wizualnych, które antropomorfizują technologię cyfrową. W podsumowującej scenie finałowej pojawiła się wizualizacja ruchomej, czerwonej sfery, która miała reprezentować sztuczną inteligencję (Il. 12). Tak

przedstawiona AI przemawia bezpośrednio do widza i mówi, że ma zamiar zmienić świat, a im więcej danych zostanie jej udostępnionych przez użytkowników, tym więcej się nauczy. Ta wypowiedź sugeruje, że sztuczna inteligencja jest podmiotem, który przejmuje ludzkie dane. Sformułowanie to nadaje jej działaniu cechę intencjonalności. Tak skonstruowaną metaforę można uznać za szkodliwą społecznie, ponieważ rozpowszechnia ona przekonanie, że to sztuczna inteligencja, a więc technologia przejmuje ludzkie dane, a nie właściciele firm, które ją produkują. Upodmiotowienie oprogramowania na rzecz uproszczonych metafor, mających na celu ułatwienie przekazu na temat technicznych procesów skutkuje tym, że sprowadzone ono zostaje wyłącznie do sfery technologicznej, z pominięciem znaczących aspektów społecznych. W rezultacie tego rodzaju przedstawienia utrudniają przeprowadzenie krytyki tych procesów i, wbrew własnym założeniom, przeciwdziałają szerzeniu świadomego podejścia wśród użytkowników.

W dalszej części owej finalnej mowy sztucznej inteligencji w filmie padają słowa: „Niektórzy mówią, że inteligencja bez etyki to żadna inteligencja. Ja mówię: zaufajcie mi. Co może pójść nie tak?”<sup>298</sup> W tym przedstawieniu oprogramowanie mówi własnym głosem i przemawia w swoim imieniu. Oczywiście wypowiedź AI jest ma tu charakter sarkastyczny, zawarta w niej prośba o obdarzenie zaufaniem ma na celu wzbudzenie w widzu przeciwnej reakcji. Nie wydaje się jednak, żeby wzbudzanie nieufności względem technologii było właściwym celem inicjatyw, które zakładają szerzenie świadomego użytkowania technologii cyfrowych. Aby osiągnąć to założenie powinny one raczej wzbudzać nieufność chociażby względem producentów oprogramowania albo całego modelu ekonomicznego, na którym oparta jest jego dystrybucja. Obraz technologii cyfrowej został upodmiotowiony i zantropomorfizowany, mimo iż na początku filmu umieszczona została wypowiedź Meredith Broussard, badaczki koncentrującej się na roli sztucznej inteligencji w dziennikarstwie, dotycząca tego, że metaforyzowanie oprogramowania, w taki sposób jak robią to hollywoodzkie produkcje, nie jest korzystne dla kształtowania przystającego do rzeczywistości wyobrażenia na temat technologii wśród społeczeństwa.<sup>299</sup> Pomimo że film dokumentalny na początku powołuje się na to stwierdzenie i tak w jego narrację włączony został zantropomorfizowane wyobrażenie sztucznej inteligencji. Udowadnia to jak głęboko

---

<sup>298</sup> *Zakodowane uprzedzenia*, dz. cyt.

<sup>299</sup> Badaczka poświęciła temu zagadnieniu publikację, w której przedstawiła to, że kultura popularna wykształciła w użytkownikach technologii wysokie oczekiwania, które nie odpowiadają jej rzeczywistemu działaniu. Autorka udowadnia, że systemy oparte na sztucznej inteligencji nie są wcale tak niezawodne i inteligentne jak się o nich powszechnie myśli. Zob. M. Broussard, *Artificial Unintelligence*, MIT Press, Cambridge, Mass. 2018.

zakorzenione są w kulturze pielęgnowane przez lata wyobrażenia i przekonania na temat technologii cyfrowej.<sup>300</sup>



Il. 12. Kadr z filmu *Zakodowane uprzedzenie*. Abstrakcyjna reprezentacja oprogramowania. Źródło: *Zakodowane Uprzedzenie*, reż. Shalini Kantayya, Chiny, USA, Wielka Brytania 2020.

Filmy dokumentalne mogą być przez ich widzów uznawane za źródło wiedzy, przez fakt, że przedstawiają wywiady ze specjalistami w danej dziedzinie i zbierają materiały filmowe mające przedstawiać pewien fragment rzeczywistości. Dwa omawiane tu przykłady zostały opublikowane na powszechnej platformie streamingowej Netflix, a więc dotarły do szerokiego grona odbiorców. Przywoływane przykłady udowadniają, że utrwalone w kulturze wyobrażenia i przekonania na temat technologii cyfrowej posiadają tak silną moc oddziaływania, że nie jest łatwo prowadzić narrację o oprogramowaniu wolną od ich wpływu. Na podstawie opisywanych przeze mnie przykładów daje się również zaobserwować, że tworzenie wizualnych metafor i reprezentacji oprogramowania wiąże się z ogromną odpowiedzialnością społeczną, ponieważ jest to proces wytwarzania wiedzy i przekonań na temat działania technologii cyfrowej. Tym samym, podmioty, które podejmują się tak ważnego zadania, ale korzystają przy tym z płytkich, nietrafnych reprezentacji albo chcą ukryć pod powierzchnią trywialnej metafory rzeczywiste działanie oprogramowania jako systemu operacyjnego kultury, albo liczą na osiągnięcie łatwych

---

<sup>300</sup> Więcej o potrzebie antropomorfizowania technologii cyfrowej zob. S. Turkle, *Alone Together. Why We Expect More from Technology and Less from Each Other*, Basic Books, Nowy Jork 2010.

zysków poprzez odniesienie się do popularnych wizerunków zantropomorfizowanej formy technologicznej, które są łatwo rozpoznawalne i w bezpośredni sposób odnoszą się do utrwalonych w kulturze ikonografii.

Odpowiedzi na problem płytkich metafor oprogramowania, a co za tym idzie szkodliwych społecznie strategii wytwarzania wiedzy, można szukać w obszarze krytyki samego procesu projektowania (*design*). Członek Medialabu Katowice<sup>301</sup>, Karol Piekarski, wskazywał na to, że odpowiedzialne podejście do procesu projektowania powinno zakładać pracę zespołową oraz interdyscyplinarną wymianę doświadczeń, co przedstawiał na przykładzie praktyk medialabowych.<sup>302</sup> Podobną tezę przedstawiał Marcin Składanek, który podkreślał znaczenie społecznej odpowiedzialności w procesie projektowym.<sup>303</sup> Obaj badacze-praktycy budowali swoje rozpoznania w odniesieniu do procesu projektowania oraz wizualizacji danych, ale myślę, że dają się one zastosować także w kontekście omawianego przeze mnie problemu tworzenia wizualnych reprezentacji oprogramowania, ponieważ tak samo jest to metoda konstruowania wiedzy społecznej. Korzystając więc z opracowań dotyczących krytyki samego procesu projektowania, chciałabym podkreślić, że rozwiązaniem problemu trywialnych metafor oprogramowania może być przede wszystkim większa świadomość znaczenia budowania wizualnych reprezentacji software'u i ich społecznego impaktu, a także, co za tym idzie, powoływanie zespołów, które poprzez interdyscyplinarną wymianę myśli, są w stanie wypracować nieoczywiste i pogłębione przedstawienia software'u. Do tych koncepcji powrócę w rozdziale piątym.

### 3.6 Podsumowanie. Propozycja metafory oprogramowania

Kluczowym pojęciem, które stanowiło podstawę dla przeprowadzonych w poprzednim rozdziale rozważań była kategoria modelu, która odnosiła się do opartego na redukcji procesu kreowania pewnego uproszczonego obrazu świata. Modele stanowią podstawę działania systemów cyfrowych i przez to przyczyniają się do tego, że w działaniu oprogramowania mogą urzeczywistniać się ukryte w nich wyobrażenia i przekonania.

---

<sup>301</sup> Inicjatywę Medialab Katowice opisywałam w pierwszym rozdziale pracy, w podrozdziale 1.3 *Software studies w Polsce*.

<sup>302</sup> K. Piekarski, *Od modernizmu do medialabów. Krytyczne projektowanie w czasach nadmiaru danych*, „Formy” 2019, nr 2, online: <https://formy.xyz/arttykul/od-modernizmu-do-medialabow-krytyczne-projektowanie-w-czasach-nadmiaru-danych>, dostęp: 25.11.2023.

<sup>303</sup> M. Składanek, *Od Isotype do Many Eyes...*, dz. cyt.

Elementem łączącym porządek społeczny i technologiczny jest język, który odgrywa kluczową rolę zarówno w procesie wytwarzania oprogramowania i formułowania modelu, jak i konstruowania wyobrażeń i metafor na temat działania technologii. Metafora jest wyobrażeniem na temat rzeczywistości, które realizuje się za pomocą języka. Z tej perspektywy można powiedzieć, że model jest rodzajem metafory.

Zagadnienie metaforyzowania oprogramowania również daje się postrzegać z dwóch perspektyw. Pierwszą z nich jest kwestia metafor dotyczących oprogramowania funkcjonujących w kulturze, a więc to, jak ludzie używają języka aby zrozumieć jego działanie. Drugą perspektywą jest to, że samo oprogramowanie może pełnić rolę metafory, ułatwiającej zrozumienie określonych procesów, jak miało to miejsce na gruncie nauk kognitywnych, gdzie podział na software i hardware miał odzwierciedlać rozróżnienie umysłu i mózgu. Joseph Weizenbaum, autor terapeutycznego bota ELIZA, argumentował, że komputery stały się metaforą wszystkich procesów, które dają się ująć w postaci proceduralnej.<sup>304</sup> Badacz przede wszystkim odnosił się tu do zjawisk, które poddają się sprowadzeniu do postaci algorytmu, a więc określonej liczby następujących po sobie instrukcji. Do takich zjawisk można zaliczyć pracę biurową, ekspresję genów, zarządzanie ruchem drogowym itd.

Problem funkcjonowania wyobrażeń na temat technologii cyfrowej w kulturze sprowadza się do braku dobrych metafor oprogramowania. Dobrych, a więc takich, które pozbawione są niepotrzebnego udratyzowania, nie służą podsycaniu obecnych w kulturze popularnej obrazów demonicznej sztucznej inteligencji, ale które sprzyjają uświadamianiu użytkowników w temacie właściwego działania oprogramowania i konsekwencji społecznych za tym idących. W tym kontekście można by oprzeć się na figurze mechanicznego „Turka”. Patrząc na automat użytkownik skupia się na manekinie (interfejsie), który stanowi warstwę zewnętrzną, użytkową i nie pozwala mu na zgłębienie działania ukrytego mechanizmu (oprogramowania). Interfejs ma charakter antropomorficzny i jest wyrazem funkcjonujących w społeczeństwie wyobrażeń, sprzyja również procesom projekcyjnym jak np. potrzeba usytuowania technologii w roli rywa. Mechanizm odpowiadający za działanie automatu ukryty jest przed oczami obserwujących, nie jest dostępny w bezpośrednim poznaniu. Jednak najważniejszym elementem tego układu, kluczowym dla zrozumienia jego właściwego działania nie jest ani interfejs ani warstwa technologiczna (software i hardware), ale ów człowiek schowany wewnątrz skrzyni,

---

<sup>304</sup> J. Weizenbaum, *Computer Power and Human Reason: From Judgment to Calculation*, W. H. Freeman, San Francisco 1976, s. 157.

o którego istnieniu użytkownicy nie mają pojęcia. Na rzecz zaproponowanej tu metafory nie należy go rozumieć jako pojedynczego człowieka, ale ma on symbolizować całokształt społecznych uwarunkowań, które szczegółowo omówione zostały w tym rozdziale. Właściwym pytaniem, które należy stawiać odnośnie społecznego aspektu technologii cyfrowych jest kwestia owego ukrytego (niczym człowiek w mechanicznym „Turku”) aspektu, którym mogą być chociażby interesy i cele korporacji odpowiadającej za tworzenie danego oprogramowania lub ekonomiczne uwarunkowania systemu kapitalistycznego.

## **II. Kontekst historyczny dla rozdziału czwartego: media masowe**

*Postmedialne uniwersum nie różni się (...) od wcześniejszych kulturowych kondycji technologii komunikacyjnych – od ery pisma, epoki druku Gutenberga czy elektrycznej fazy kultury masowej z jej analogowymi mass mediami.*<sup>305</sup>

Piotr Celiński

### **Media analogowe – media masowe**

Przed rozpoczęciem rozważań zawartych w rozdziale czwartym chciałabym najpierw przywołać kontekst historyczny związany z początkiem rozwoju kultury masowej, który opierał się na technologiach analogowych, nie wykorzystujących jeszcze *stricte* oprogramowania. W niniejszym wprowadzeniu skupię się przede wszystkim na historycznym już dziś przykładzie radia, jako technologii, która zrewolucjonizowała system transmisji treści kulturowych i uczestnictwa w zbiorowym ich odbieraniu, ale przede wszystkim otworzyła drogę do tego, aby zadawać pytania o status uczestnika masowej kultury i o jego miejsce w procesie odbioru i tworzenia treści.

Radio to wynalazek sprzed ponad stu lat, który powstał dzięki rozwinięciu się telegraficznych technik transmisji bezprzewodowych i funkcjonuje w życiu społecznym aż do współczesności. Technologia ta może posłużyć za jeden z punktów wyjścia do wprowadzania odniesień pomiędzy obecnie panującą technologią cyfrową a kulturą mediów analogowych z pierwszej połowy XX wieku. Z pewnością dzisiejsze radio nie działa już na takich samych zasadach jak miało to miejsce przed wprowadzeniem telewizji. Jego popularność i siła impaktu społecznego znacznie spadła. Zmieniła się również jego technologiczna forma, ponieważ częściowo funkcjonuje dziś nie tylko w spektrum fal, ale również za pośrednictwem sieci, w postaci radia internetowego. Niniejsze wprowadzenie odwoływać się będzie do dwóch kwestii związanych z tą technologią. Pierwszą z nich będzie to, w jaki sposób myślano o radiu i możliwościach komunikacyjnych, dla których otworzyło ono drogę. Drugą natomiast będzie problem regulacji prawnych, które spowodowało wprowadzenie tego medium.

---

<sup>305</sup> P. Celiński, *Postmedia...*, dz. cyt., s. 15.

Rozpowszechnienie się radia na początku XX wieku przyczyniło się do rozwoju kultury masowej. Wynalazek ten umożliwił coś, co wcześniej nie mogło mieć miejsca, a więc zdalne współuczestnictwo w kulturze szerokiemu i rozproszonemu gronu odbiorców. Partycypacja zapośredniczona przez radio wykluczała jednak komunikację przez jednostronną naturę przekazu informacji. W związku z tym, Bertolt Brecht, człowiek teatru, pisarz, poeta oraz naoczny świadek pojawienia się tej technologii, w eseju *The radio as an apparatus of communication* wyraził spostrzeżenie, które jest interesujące również z perspektywy współczesnej. Napisał, że radio powinno być medium komunikacji, a nie dystrybucji informacji, że powinno umożliwiać nie tylko odbieranie sygnału, ale również jego nadawanie przez słuchaczy:

Radio jest jednostronne, a powinno być dwustronne. Jest to wyłącznie apparatus do dystrybucji, do zwykłego dzielenia się. Oto więc pozytywna sugestia: zmienić to urządzenie z dystrybucyjnego na komunikacyjne. Radio byłoby najlepszym możliwym aparatem komunikacyjnym w życiu publicznym, ogromną siecią połączeń. To znaczy, byłoby, gdyby wiedziało, jak odbierać i nadawać, jak pozwolić słuchaczowi mówić i słyszeć, jak nawiązać z nim relację zamiast go izolować. Zgodnie z tą zasadą, radio powinno wyjść z branży dostawczej i zorganizować swoich słuchaczy jako dostawców.<sup>306</sup>

Brecht wskazywał na niewystarczalność jednostronnej komunikacji radiowej i opisywał model, w którym to użytkownicy technologii zostają zaangażowani jako dostawcy treści. Pisarz nieświadomie opisał sposób działania współczesnych mediów komunikacyjnych. Wyraził również potrzebę powstania modelu opartego na relacji, a nie izolacji.

Brechta niekiedy uznaje się za jednego z pierwszych, którzy tworzyli podwaliny pod marksistowską teorię mediów opartą na formie i materialności. Zainspirowany tymi rozważaniami niemiecki pisarz, Hans Magnus Enzensberger, w eseju *Constituents of a Theory of the Media* rozwija dalej te spostrzeżenia i podkreśla, że tak naprawdę, na poziomie technicznym nic nie stało na przeszkodzie, aby zmienić radio w nadajnik. Z perspektywy elektronicznej nadajniki i odbiorniki są sobie równe, dlatego budowa radia pozwalałaby na zmienienie go w transponder. Okazuje się więc, że marzenie Brechta o dwukierunkowym radiu było niemożliwe do spełnienia nie przez problem natury technicznej, ale politycznej. Enzensberger zauważa, że:

---

<sup>306</sup> B. Brecht, *The radio as an apparatus of communication*, w: *New Media. Theories and Practices of Digitextuality*, Nowy Jork 2003, s. 29-31. Tłumaczenie A.P.

Techniczne rozróżnienie pomiędzy odbiornikami i nadajnikami odzwierciedla społeczny podział pracy na producentów i konsumentów, który w przemyśle świadomościowym nabiera szczególnego politycznego znaczenia. W ostatecznej analizie opiera się ono na podstawowej sprzeczności między klasą rządzącą a klasą rządzoną, czyli między monopolistycznym kapitałem lub monopolistyczną biurokracją z jednej strony a zależnymi od niej masami z drugiej.<sup>307</sup>

Niemiecki pisarz starał się zaadoptować marksistowską filozofię na grunt dyskursu dotyczącego społecznego wymiaru technologii. Jak sam zauważa, w czasie, gdy pisał swój artykuł na początku lat 70., nie powstała wówczas jeszcze żadna marksistowska teoria mediów.<sup>308</sup> Formułuje on tezę głoszącą, że decyzja o tym, by radio pozostało jednostronnym odbiornikiem była uwarunkowana politycznie, aby skuteczniej sprawować kontrolę nad społeczeństwem poprzez, jak to określa autor, „przemysł świadomościowy”.<sup>309</sup> Badacz wskazywał również, kto zalicza się do klasy rządzącej i był to: monopolistyczny kapitał oraz monopolistyczna biurokracja. Kontrola społeczna w tym kontekście jest więc uwarunkowana ekonomicznie oraz administracyjnie.

Skojarzenie radia, z kategorią przemysłu świadomościowego rezonuje też z założeniami Theodore’a Adorno, który już wcześniej posługiwał się określeniem „przemysłu kulturalnego”.<sup>310</sup> Przemysł kulturalny również odnosił się do kultury masowej, a szczególnie do jej zautomatyzowanego charakteru, przypominającego porządek fabryki. Adorno sformułował to określenie przebywając w Stanach Zjednoczonych (lata 1938-1949), dokąd przeniósł się, kiedy partia narodowosocjalistyczna odebrała mu uprawnienia wykładowcy w Niemczech. Obserwował tam z bliska realia amerykańskie i tamtejszą kulturę masową.<sup>311</sup> Adorno nie używał słowa przemysł w znaczeniu technologicznym, lecz socjologicznym, dotyczył on przede wszystkim sposobu, w jaki wytwarzane i ujednolicane są produkty kultury. Esej *O fetyszymie muzycznym i regresji słuchania*, w którym ujęte zostały rozważania istotne z perspektywy koncepcji przemysłu kulturalnego, został napisany po tym jak Adorno wraz z Paulem Lazarsfeldem, amerykańskim socjologiem,

---

<sup>307</sup> H.M. Enzensberger, *Constituents of a Theory of the Media*, w: *Video culture. A critical investigation*, red. J. Hanhardt, Visual Studies Workshop Press, Nowy Jork 1986, s. 98. Tłumaczenie A.P.

<sup>308</sup> Tamże, s. 97.

<sup>309</sup> Tamże.

<sup>310</sup> Pojęcie przemysłu kulturalnego zostało wypracowane przez Theodore’a Adorno oraz Maxa Horkheimera w latach 40. XX wieku. Ich wspólna książka *Dialektyka Oświecenia* wniosła istotny wkład do dyskusji na temat pochodzenia totalitaryzmów: M. Horkheimer, T.W. Adorno, *Dialektyka oświecenia. Fragmenty filozoficzne*, tłum. M. Łukasiewicz, Wydawnictwo Instytutu Filozofii i Socjologii PAN, Warszawa 1994, s. 138.

<sup>311</sup> T. Maślanka, R. Wiśniewski, *Wstęp do wydania polskiego*, w: T. W. Adorno, *Przemysł kulturalny. Wybrane eseje o kulturze masowej*, tłum. M. Bucholc, Narodowe Centrum Kultury, Warszawa 2019, s. 10-12.

współuczestniczył w projekcie badawczym, dotyczącym analizy wpływu transmisji radiowych na umasowienie i odbiór muzyki.<sup>312</sup> Nawiązując do terminologii wykorzystywanej przez Karola Marksa, Adorno opisywał postępującą fetyszyzację produktów kulturowych, szczególnie widoczną na przykładzie muzyki.<sup>313</sup> Badacz omawiał przykłady wykorzystywania muzyki jako tła muzycznego dla reklam oraz zjawisko kultu samej wartości wymiennej dóbr kulturowych – bilet na koncert Toscaniniego ma większą znaczenie dla człowieka niż sam koncert. Pokazywał on rosnący wpływ wartości komercyjnych w kształtowaniu obiektów kultury.

Adorno podkreślał zatem zdepersonalizowany charakter kultury masowej, w szczególności zwracał uwagę na to, że nie jest ona wytworem mas, mimo ich udziału, ale stanowi ogniwo w rozwoju i proliferacji systemu kapitalistycznego. Innymi słowy, mimo że w procesie wytwarzania produktów kultury uczestniczą członkowie społeczeństwa, pozostaje ona podporządkowana kapitalistycznym zasadom ekonomicznym, a powstające w ramach tego procesu obiekty kultury są ujednolicane i sprowadzane do swojej wartości wymiennej.

Podsumowując, zarówno Enzensberger, jak i Adorno zwracali uwagę na podporządkowanie technologicznie zapośredniczonego procesu wytwarzania treści kultury masowej ekonomicznym warunkom systemu kapitalistycznego. Podkreślali, że od początku, wraz z rozwojem technologii, które umożliwiły powstanie kultury masowej, równolegle powstawały strategie jej ekonomicznego i politycznego zawłaszczania, a poprzez to również metody kontroli użytkowników. Ich miejsce w procesie tworzenia, nadawania i odbioru już przed rozwojem technologii cyfrowych ograniczane było do biernego konsumowania dystrybuowanych w kulturze masowej treści.

## **Regulacje prawne**

Radio, jako jedno z pierwszych mediów masowej transmisji było również narzędziem kreowania zbiorowej świadomości i kultury. Dlatego też stanowiło silne narzędzie politycznej kontroli. Walka o dominację kulturową za pomocą radia uwidacznia się na przykładzie historii regulacji określających prawo dostępu do nadawania i odbierania. Piotr Celiński podkreślał, że niezmiennie, na przestrzeni wieków, władza nad kulturą należy do

---

<sup>312</sup> Tamże, s. 18-19.

<sup>313</sup> T.W. Adorno, *Przemysł kulturalny...*, dz. cyt., s. 77-78.

tę, kto poprzez regulacje prawne zarządza dostępem do technologii i przez to sprawuje kontrolę nad zbiorową wyobraźnią komunikacyjną oraz nad definicją podmiotu zanurzonego w kulturze.<sup>314</sup> Różnica pomiędzy dostępem do kanałów komunikacji pomiędzy epoką analogową a cyfrową jest ogromna. Radio nie zostało wprowadzone jednocześnie z regulacjami prawnymi je ograniczającymi. Na początku funkcjonowało w sposób swobodny i inspirowało wielu pasjonatów do eksperymentowania z nadawaniem i odbieraniem treści. W Ameryce doprowadziło to nawet do sytuacji, w których amatorzy nadawali na częstotliwościach marynarki wojennej i wysyłali statki na nieprawdziwe misje.<sup>315</sup> Dopiero tragedia Titanica wymusiła wprowadzenie ustawy o radiofonii w 1912 roku, przez liczne oskarżenia kierowane ze strony reprezentantów marynarki. Zarzuty dotyczyły tego, że amatorskie działania przyczyniły się do chaosu komunikacyjnego w spektrum radiowym i utrudniły statkom ratowniczym komunikację z tonącym transatlantykem. Wprowadzony w 1912 roku *An Act to Regulate Radio Communication* był pierwszą na świecie ustawą regulującą licencję na posiadanie stacji radiowych<sup>316</sup>

To wydarzenie doskonale obrazuje ścieranie się ze sobą dwóch tendencji w obszarze społecznej komunikacji medialnej. Pierwsza z nich jest nacechowana liberalnie, dąży do zachowania wolności, swobody działania i wypowiedzi, które trwają i działają również współcześnie. Druga zaś wyraża podejście paternalistyczne, które zakłada istnienie zewnętrznego nadzoru umotywowanego potrzebą ochrony i zapewnienia bezpieczeństwa, aby szeroko pojęta wolność jednej jednostki nie naruszała praw innych. Te dwa kierunki rozwoju społecznej sieci komunikacyjnej obecne są w sferze medialnej aż do współczesności i doskonale wyrażają się w rzeczywistości internetowej. Dyrektor firmy Google, Eric Schmidt, określił internet jako „największy eksperyment anarchistyczny jaki do tej pory powstał”.<sup>317</sup> Od cyberprzestrzeni z jednej strony wymaga się, aby była polem wymiany myśli, komunikacji oraz aby akceptowała wolność jednostki, jednak z drugiej strony obecne jest także przekonanie, użytkownik, który z łatwością wchodzi w sferę komunikacji sieciowej powinien być w jakiś sposób chroniony.

To krótkie wprowadzenie miało na celu przywołanie kontekstu historycznego, niezbędnego do zrozumienia tego, że opisywane w rozdziale czwartym problemy społeczne,

---

<sup>314</sup> P. Celiński, *Postmedia...*, dz. cyt., s. 39.

<sup>315</sup> H.R. Slotten, *Radio and television regulation: broadcast technology in the United States, 1920-1960*, Johns Hopkins University Press, London 2003, s. 6-8.

<sup>316</sup> J.E. Nuechterlein, P.J. Weiser, *Digital Crossroads. American Telecommunications Policy in the Internet Age*, The MIT Press, Cambridge, Mass. 2005, s. 231-232.

<sup>317</sup> E. Schmidt, *Internet World Trade Show*, Nowy Jork 1999, online: <https://www.oxfordreference.com/view/10.1093/acref/9780191826719.001.0001/q-oro-ed4-00017947>, dostęp: 06.11.2023.

dotyczące cyfrowej kultury, takie jak wykorzystywanie technologii kultury masowej do politycznej kontroli, nie wynikają bezpośrednio i wyłącznie z działania oprogramowania, ale miały już miejsce wcześniej, u początku rozwoju kultury masowej, opierającym się na mediach analogowych.

## **Rozdział 4**

### **Media społecznościowe jako narzędzie społecznej kontroli**

*Pojawienie się internetu jako medium masowego uosabia nową strukturę władzy, ale i nowe możliwości wolności poza kontrolą.*<sup>318</sup>

Wendy Hui Kyong Chun

#### **4.1 Media cyfrowe – media społecznościowe**

Brecht, pisząc o tym, że radio powinno umożliwiać dwustronną komunikację, wyraził potrzebę powstania modelu opartego na relacji, a nie izolacji. Już w drugiej połowie XX wieku Jean Baudrillard podkreślał, że społeczeństwo zanurzone jest w ekstazie komunikacji, pisał o końcu alienacji, zaznaczając, że wszystko stało się przezroczyste, widzialne i wyeksponowane.<sup>319</sup> Nie można zaprzeczyć, że komunikacja w kulturze cyfrowej jest powszechna, łatwa i błyskawiczna, jednak nie wszystkie jej elementy są jawne, ponieważ ideologiczna praca oprogramowania w kulturze polega na eksponowaniu interfejsów przy jednoczesnym ukrywaniu innych treści, chociażby związanych z materialnością maszyny czy ekonomią jej produkcji lub użycia. Przejście od pasywnego, wyalienowanego użytkownika mediów masowych do aktywnego, komunikatywnego użytkownika mediów cyfrowych było procesem kulturowym, bezpośrednio związanym z rozwojem technologii.

Wprowadzenie pierwszych komputerów osobistych i rozpowszechnianie się internetu nie było jednoznaczne z natychmiastową zmianą charakteru technologii komunikacyjnych na społecznościowy, a więc taki, który funkcjonuje przede wszystkim w oparciu o aktywność swoich użytkowników. Początek procesu cyfryzacji, a więc przełom XX i XXI wieku, określa się jako czas funkcjonowania Web 1.0, archaicznej już dziś sieci, która nie miała jeszcze tak społecznego wymiaru, jak ma to miejsce współcześnie, ale raczej powielala modele komunikacyjne zeszłej epoki. Znamionnym zjawiskiem, wyrażającym charakter Web 1.0 jest tak zwana bańka dot-comów. Terminem tym określa się wybuch euforii, do której przyczyniło się powstanie pierwszej innowacyjnej przeglądarki (Netscape), systemu operacyjnego (Windows 95) oraz języka programowania (Java), co umożliwiło wielu firmom rozszerzenie swojej działalności o sprzedaż internetową.

---

<sup>318</sup> W. Hui Kyong Chun, *Control and Freedom...*, dz. cyt., s. 2. Tłumaczenie A.P.

<sup>319</sup> J. Baudrillard, *The Ecstasy of Communication*, tłum. Bernard Schutze, Caroline Schutze, Semiotext(e), Brooklyn 1988, s. 21–22.

Wspomniana euforia wyrażała się poprzez zintensyfikowane powstawanie firm działających w internecie (ich adres internetowy kończył się sformułowaniem „.com”, co określało ich komercyjny charakter) oraz poprzez powszechne inwestowanie w ich akcje na giełdzie. W owym czasie każdy dot-com, bez względu na rozmiar i zasięg działalności, a nawet na to, czy już został założony, czy dopiero powstawał jako pomysł, mógł liczyć na wsparcie szeregu inwestorów giełdowych.<sup>320</sup> Nagły wzrost zainteresowania działalnością internetową, jak również i drastyczne załamanie się zjawiska bańki dot-comów po 2001 roku posiadały wspólną przyczynę – brak wiedzy na temat działania internetu oraz bardzo optymistyczne podejście do tego rodzaju technologii. Nadmiar firm liczących na szybki zysk, jak również przerost inwestycji giełdowych, które nie zwracały spodziewanego profitu, doprowadziły do pęknięcia bańki dot-comów, bankructwa tysięcy przedsiębiorstw, likwidacji ogromnej ilości miejsc pracy, a przede wszystkim upadku pierwszego modelu sieci, opartego niemal całkowicie na ekonomii.<sup>321</sup> W rezultacie podjęto drugą próbę sformułowania nowego modelu sieciowego pod głośnym hasłem Web 2.0, które wybrzmiewało zarówno na gruncie naukowym, jak i popularnym. Web 2.0 w odróżnieniu od swojej poprzedniej wersji, miało zostać oddane w ręce użytkowników, a więc powiązane z nim nadzieje o stworzeniu technologicznej sieci o społecznym, komunikacyjnym charakterze. Zmiana myślenia o strategii rozwoju internetu została wyrażona chociażby w słynnym wydaniu amerykańskiego czasopisma „Time” o lustrzanej okładce. W numerze tym ogłoszono, że prestiżowy tytuł człowieka roku otrzymali wszyscy użytkownicy internetu. W rezultacie, każdy mianowany tym tytułem mógł obejrzeć własne odbicie na pierwszej stronie.<sup>322</sup> Ten symboliczny gest miał na celu zwrócenie uwagi na istotną rolę, jaką we współczesnym świecie medialnym pełnią użytkownicy. Przyczynił się on do zwrotu społecznościowego, a więc pozornie usytuowanego po stronie użytkownika w myśleniu o sieci komunikacyjnej. Ogromne, opiniotwórcze koncerny medialne starały się opisać zachodzącą zmianę modelu sieci jako ruch w stronę użytkowników internetu i przekazanie im wolności twórczej.

Pierwszy model sieci Web 1.0 stanowił próbę przejęcia dopiero rozwijającego się narzędzia cyfrowego przez przedsiębiorstwa komercyjne i podporządkowania go celom ekonomicznym. Natomiast o drugiej odsłonie tego modelu, o Web 2.0, opowiadało się tak, jakby miała być ona formą społecznej emancypacji spod owego ekonomicznego ucisku. Nie

---

<sup>320</sup> *Bańka dotcomów i jej wpływ na rynek*, 12.09.2019, online: <https://histmag.org/Banka-dotcomow-i-jej-wplyw-na-rynek-19351/>, dostęp: 06.11.2023.

<sup>321</sup> Nie wszystkie dot-comy upadły po załamaniu się bańki. Te, które przetrwały trudny okres funkcjonują do dziś jako dominujące przedsiębiorstwa na rynku, np.: Ebay, Amazon, Google.

<sup>322</sup> P. Celiński, dz. cyt., s. 39.

bez znaczenia pozostaje fakt, że rewolucja dotycząca zmiany modelu sieci nie wyniknęła z żadnego społecznego ruchu, czy ze szczególnej aktywności użytkowników, ale powstała jako narracja rozpowszechniona przez wpływowe ośrodki medialne.<sup>323</sup> Przy czym należy podkreślić, że narracji tej brakowało spójności, ponieważ z jednej strony w nowym modelu sieci podstawową rolę odgrywać miała treść generowana przez użytkowników, co miało świadczyć o ich społecznym charakterze, z drugiej jednak strony konferencja<sup>324</sup>, która przyczyniła się do rozpowszechnienia terminu Web 2.0, organizowana w 2004 roku przez przedsiębiorstwo medialne O'Reilly Media prezentowała ten nowy model jako przedsięwzięcie biznesowe. W opisie konferencji zawarta została informacja o tym, że poprzedni model sieci skupiał się przede wszystkim na wyszukiwarkach internetowych, natomiast nowy ma rozpowszechnić ideę sieci jako platformy, która jest miejscem działania aplikacji internetowych.<sup>325</sup> To przeformułowanie było reprezentowane jako remedium na porażkę pierwszego modelu i miało dostarczyć nowych możliwości i rozwiązań biznesowych. Nie bez znaczenia pozostaje również fakt, że mówcami na konferencji byli między innymi przedstawiciele korporacji, które przetrwały upadek bańki dot-comów, tacy jak: Jeffrey Bezos, założyciel Amazona, a także reprezentanci firm eBay, IBM i Yahoo. Hasło Web 2.0 przedstawiane jednocześnie jako przedsięwzięcie przekazujące rozwój sieci w ręce jej użytkowników oraz jako nowe rozwiązanie biznesowe dla firm, które przetrwały załamanie poprzedniego modelu, wydaje się być pustym sloganem, zakrywającym fakt, że stare modele ekonomicznej kontroli cały czas są w nim obecne, ale w nowej, dyskretniejszej formie. Przekonanie to wyraził również Piotr Celiński w słowach:

Facebook, Flickr, Youtube i wiele ich przeróżnych, mniej znanych odpowiedników pod pozorem otwierania się na potencjał kulturowy i społeczny oraz kreatywność swoich użytkowników okazały się efektywnymi ekonomicznie przedsiębiorstwami, które ucieleśniają dominujący w poprzedniej epoce medialnej model ekonomiczny.<sup>326</sup>

Wobec tak zarysowanej sprzeczności zawartej w idei Web 2.0 nasuwa się pytanie o to, w jaki sposób użytkownik nowego modelu będzie chroniony, w jaki sposób regulowane będą zasady funkcjonowania aplikacji internetowych, które rozszerzyły swój zakres oddziaływania na skalę globalną?

---

<sup>323</sup> Tamże, s. 43.

<sup>324</sup> *Web2.0 Conference*, 2004, online: <http://web.archive.org/web/20040602111547/http://web2con.com>, dostęp: 06.11.2023.

<sup>325</sup> Tamże.

<sup>326</sup> P. Celiński, *Postmedia...*, dz. cyt, s. 43.

## 4.2 Prawo w przestrzeni cyfrowej

Rozwój technologii jest procesem, który stwarza nową rzeczywistość kulturową, a co za tym idzie, determinuje powstawanie określającego ją prawa. Często najpierw dochodzi do wprowadzenia jakiegoś wynalazku i dopiero w odpowiedzi na zmiany, które spowodował, sformułowane zostają odpowiednie regulacje prawne. W tak ujętym procesie kulturowym część norm prawnych okazuje się być pochodną rozwoju technologicznego. Proces ten był wyraźny w przypadku opisywanego wcześniej wątku radia i aktu regulującego komunikację radiową z 1912 roku oraz będzie również dostrzegalny w kolejnych przykładach omawianych w tym rozdziale. Prawo jest zbiorem norm, który nie ma postaci materialnej, jednak determinuje świat rzeczywisty i zachowanie jednostki w nim zanurzonej. Istnieje bezpośrednia zależność pomiędzy prawem a przestrzenią geograficzną, ponieważ organem je tworzącym i regulującym są władze państw. Dlatego też kodeksy prawne różnią się w zależności od tego, jaki kraj obejmują, a czynność karana w jednym państwie może być legalna w innym. Związek prawa z przestrzenią geograficzną przestaje być oczywisty, jeśli odnosi się on do świata wirtualnego. Mimo że sieć pojmujemy się za pomocą kategorii przestrzennych, na co wskazuje chociażby określenie „cyberprzestrzeń”, wykracza ona poza powszechnie przyjęte podziały terytorialne. Howard Rheingold pisał, że cyberprzestrzeń jest „przestrzenią konceptualną, w której słowa, ludzkie relacje, dane, dobra oraz kontrola są manifestowane przez ludzi poprzez zapośredniczoną komputerowo komunikację”<sup>327</sup>. Sfera wirtualna jest więc przedłużeniem aktywności ludzkiej, odbywającej się w świecie fizycznym. Podwójny charakter rzeczywistości oraz to, że to, co wirtualne przenika się z tym, co aktualne, podkreślał Paul Virilio używając pojęcia „stereo-rzeczywistość”<sup>328</sup>. Choć o sieci myśli się w kategoriach przestrzennych, nie została ona zaprojektowana w taki sposób, aby dostosowywać się do terytorialnych podziałów świata fizycznego. Jak wyjaśniał Jack Goldsmith wraz z Timem Wu:

Sieć nie została zbudowana z myślą o geografii fizycznej. Ani adresy protokołu internetowego (identyfikator internetowy każdego komputera), ani nazwy domen internetowych (takie jak mcdonalds.com lub cnn.com), ani adresy poczty elektronicznej nie zostały zaprojektowane w taki sposób, aby niezawodnie wskazywać położenie geograficzne komputerów w sieci.

---

<sup>327</sup> H. Rheingold, *A slice of life in my virtual community*, w: *Collective Intelligence. Creating a Prosperous World at Peace*, red. M. Tovey, Earth Intelligence Network, Oakton 2008 s. 173. Tłumaczenie A. P.

<sup>328</sup> P. Virilio, *Speed and Information: Cyberspace Alarm!*, online: <http://www.tacticalmediafiles.net/n5m2/media/texts/virilio.htm>, dostęp: 18.07.2021.

Nawet nazwy domen i adresy e-mail zawierające wskazówki geograficzne – takie jak toystore.co.fr lub tonyblair@gov.uk-were – nie są wiarygodne.<sup>329</sup>

Sieć jest sferą, która w podobny sposób przenika rzeczywistość na poziomie globalnym, jednak, jak każdy obszar ludzkiej aktywności, i ona podlega regulacjom prawnym, które są z kolei zróżnicowane przez podziały terytorialne. Ta geograficzna nieprzystawalność cyberprzestrzeni stawia wyzwanie przed organami prawotwórczymi i doprowadza do paradoksalnych konfliktów, takich jak słynny spór, którego jedną stroną był Żyd francuskiego pochodzenia, Marc Knobel, a drugą przedstawiciele stworzonej w Ameryce przeglądarki Yahoo. W 2000 roku Knobel wniósł do sądu sprawę o to, że za pomocą przeglądarki Yahoo można zakupić przedmioty o nacechowaniu nazistowskim, co jest niezgodne z francuskim prawem. Reakcją przedstawicieli przeglądarki było ograniczenie tego rodzaju aukcji dla francuskiego adresu Yahoo.fr. Sąd uznał to jednak za niewystarczające, ponieważ obywatele Francji mogli dokonać zakupu kontrowersyjnych towarów przez ogólny adres Yahoo.com. Przedstawiciele przeglądarki odmówili podjęcia dalszych starań nad znalezieniem rozwiązania tej sytuacji, tłumacząc się, że jest to technicznie niemożliwe, aby dostosować internetową przeglądarkę do rozpoznawania, czy korzystający z niej użytkownik znajduje się na terenie Francji. Właściciel przeglądarki Yahoo, Jerry Yang wyraził wówczas przekonanie, że sąd Francji nie może mieć władzy nad platformą, która powstała w Stanach Zjednoczonych oraz w konsekwencji nad tym, co mogą wyświetlać amerykańscy obywatele. Powołując się na fakt, że Yahoo było usługą, funkcjonującą również w obszarze państwa francuskiego, sąd w ramach ostatecznej decyzji postawił przed Yangiem ultimatum: albo uczyni niemożliwym wyszukiwanie na terenie Francji produktów o charakterze nazistowskim, albo będzie płacił grzywnę wysokości stu tysięcy franków dziennie. Na początku 2001 roku platforma Yahoo zmieniła zasady funkcjonowania i zablokowała sprzedaż za jej pośrednictwem towarów nacechowanych nazistowsko każdemu jej użytkownikowi, bez względu na kraj przebywania.<sup>330</sup>

Historia sporu pomiędzy Knoelem a Yahoo jest znamiennym wyrazem tego, że sieć rozszerzyła wzajemne oddziaływanie na siebie zebranych w niej producentów i użytkowników na poziomie globalnym. Wykracza ono poza dotychczas ustalone normy i granice terytorialne i prowokuje do redefinicji regulacji i zasad rządzących życiem

---

<sup>329</sup> J. Goldsmith, T. Wu, *Who Controls the Internet? Illusions of Borderless World*, Nowy Jork, Oxford University Press 2006, s. 6. Tłumaczenie A.P.

<sup>330</sup> Przywołana tu historia skandalu związanego z przeglądarką Yahoo opisana została we wstępie książki: J. Goldsmith, T. Wu, dz. cyt. s. 1-10.

społecznym. Świat cyfrowy może wydawać się alternatywną w stosunku do rzeczywistości przestrzenią wirtualną, pozostaje on podporządkowany prawom, które obowiązują w świecie fizycznym. Ten nieoczywisty status cyberprzestrzeni doprowadza często do sytuacji, w których wypracowany w rzeczywistości cyfrowej sposób jej funkcjonowania zostaje zweryfikowany przez regulacje prawne rządzące światem fizycznym, co często kończy się skandalami, spowodowanymi odkrytymi nadużyciami. Jednym z najgłośniejszych skandali ostatnich lat, który podważył zaufanie użytkowników do polityki prywatności platform społecznościowych, jest afera związana z firmą Cambridge Analytica.

### 4.3 Cambridge Analytica: oprogramowanie jako inżynieria społeczna

Cambridge Analytica była brytyjską firmą, założoną w 2013 roku, gwarantującą usługi z zakresu doradztwa politycznego. Powstała jako jednostka pomocnicza, podlegająca pod główną firmę SCL Group (*Strategic Communication Laboratories*), działająca w przemyśle pozyskiwania i analizy danych. Firma ta określała się jako globalna agencja zarządzania wyborami i właściwie była prywatnym centrum badawczym w obszarze behawiorystyki i strategii komunikacyjnych. Cambridge Analytica stała się powszechnie rozpoznawalna za sprawą skandalu związanego z amerykańskimi wyborami prezydenckimi w 2016 roku, które wygrał Donald Trump. Zwycięski kandydat korzystał z usług CA w ramach swojej kampanii. Kiedy wygrał wbrew spekulacjom opinii publicznej<sup>331</sup>, wielu dziennikarzy zaczęło interesować się przyczyną jego sukcesu. Dzięki Christopherowi Wylie, demaskatorowi, który analizował dane dla CA, na jaw wyszło wiele informacji potwierdzających nieetyczne działanie firmy, które później stały się przyczyną serii procesów sądowych, wytoczonych przeciwko Cambridge Analytica.<sup>332</sup> Z racji tego, że jednym ze źródeł pozyskiwania danych o użytkownikach była platforma Facebook (CA udostępniła na tej platformie quiz dotyczący typu osobowości, którego odpowiedzi

---

<sup>331</sup> K. Ward, *Social networks, the 2016 US presidential election, and Kantian ethics: applying the categorical imperative to Cambridge Analytica's behavioral microtargeting*, „Journal of Media Ethics” 2018, nr 33(3), s. 133.

<sup>332</sup> Christopher Wylie ujawnił tajniki działania CA w wywiadzie z dziennikarką gazety Guardian: C. Cadwalladr, *‘I made Steve Bannon’s psychological warfare tool’: meet the data war whistleblower*, 18.03.2018, online: <https://www.theguardian.com/news/2018/mar/17/data-war-whistleblower-christopher-wylie-faceook-nix-bannon-trump>, dostęp: 06.11.2023. Rok później wydał książkę, w której opisał szczegółowo strategię działania brytyjskiej firmy zarządzającej danymi: C. Wylie, *Mindf\*ck: Cambridge Analytica and the Plot to Break America*, Random House, Nowy Jork 2019.

dostarczały danych psychograficznych, wykorzystywanych do manipulacji), przesłuchany został również dyrektor Facebooka, Mark Zuckerberg. W czasie procesu zeznał, że nie miał wiedzy o nieetycznym wykorzystywaniu jego platformy przez CA oraz że był przekonany, iż CA usunęła wszystkie dane, które pozyskała za pomocą jego platformy. Przesłuchanie Zuckerberga było wydarzeniem medialnym, nadawanym globalnie w postaci transmisji w czasie rzeczywistym.<sup>333</sup> Dyrektor Facebooka został obarczony karą w postaci grzywny za niezapewnienie użytkownikom, za których wziął odpowiedzialność, odpowiedniego bezpieczeństwa. Ostatecznie, w wyniku wyroku sądowego, firma Cambridge Analytica została zamknięta.

Działalność CA skupiała się przede wszystkim na technikach mikrotargetingu. Jest to strategia manipulacyjna, działająca w oparciu o olbrzymie zasoby danych, która skupia się na oddziaływaniu na jednostkę poprzez dostosowanie bodźców do preferencji oraz charakteru osoby.<sup>334</sup> Danymi zbieranymi w celu stworzenia profilu manipulowanej osoby mogą być informacje personalne, demograficzne, czy też sama aktywność w sieci. Nie muszą być one związane bezpośrednio z preferencjami politycznymi. To, jakie informacje o osobie są wykorzystywane w strategii mikrotargetingu jest zależne od zastosowanego algorytmu z rodzaju *machine learning*. Cambridge Analytica opracowała algorytm OCEAN. Jego celem było określenie pięciu cech profilowanej osoby: otwartość (*Openness*), sumienność (*Conscientiousness*), ekstrawersja (*Extraversion*), ugodowość (*Agreeableness*) oraz neurotyzm (*Neuroticism*). Pierwsza kategoria, otwartość, wyznaczała stopień, w którym użytkownik jest w stanie wykraczać poza własne nawyki i przyzwyczajenia. Druga z cech, sumienność, była parametrem, który miał określić, czy użytkownik ma tendencję do stosowania w swoim codziennym życiu harmonogramów, zasad, czy ma skłonności do porządkowania. Kolejna cecha, ekstrawertyzm, miał wyznaczyć potrzeby społeczne i stopień towarzyskości. Ugodowość dotyczyła stosunku użytkownika do potrzeb innych osób i stopnia łatwości, z jaką jest on w stanie zgodzić się na kompromisy. Ostatnia cecha, neurotyzm, miała określić to, jak łatwo użytkownik jest w stanie zacząć się bać, na ile podatny jest na bodźce negatywne. Algorytm OCEAN pozwalał na tworzenie profili psychologicznych, które przedstawiały to, które z cech i w jakim stopniu można przypisać konkretnym użytkownikom. Następnie, Cambridge Analytica była w stanie tak

---

<sup>333</sup> Materiał z przesłuchania jest dostępny w sieci: *Facebook CEO Mark Zuckerberg testifies before Congress on data scandal*, 10.04.2018, online: <https://www.youtube.com/watch?v=u-FlWZ1BOcA>, dostęp: 06.11.2023.

<sup>334</sup> O. Papakiriakopoulos, S. Hegelich, M. Shahrezaye, J.C. Medina Serrano, *Social media and microtargeting: Political data processing and the consequences for Germany*, „Big Data & Society” 2018, nr 5(2), s. 2.

manipulować przekazem informacyjnym trafiającym do użytkowników profili społecznościowych, aby wywoływać u nich projektowane reakcje, podsycać określone przekonania i kształtować tendencje polityczne. Przykładowo, osoby o wysokim wskaźniku neurotyczności zniechęcane były do rywali politycznych promowanego kandydata poprzez kojarzenie ich ze zjawiskami, które wywołują lęk społeczny, takimi jak np. terroryzm. Jeśli profil behawioralny użytkownika wskazywał na to, że najważniejsze dla niego są treści związane z rodziną, domem i stabilizacją, wśród treści, które na co dzień oglądał na platformie społecznościowej pojawiały się również sponsorowane posty, które kojarzyły promowanego kandydata z takimi ideami.<sup>335</sup> Informacje przedstawiane użytkownikom najczęściej nie były faktami, tylko fikcją dostosowaną tak, aby oddziaływać na swojego odbiorcę. Tego rodzaju działania zostały określone jako konstruowanie post-prawdy.<sup>336</sup> Terminem tym opisuje się wiadomości, których nie można nazwać faktami, ponieważ nie opisują one rzeczywistości w sposób rzetelny. Te nieprawdziwe informacje mogą jednak wyrażać czyjeś realne odczucia i przekonania i z tego powodu zyskiwać siłę oddziaływania.<sup>337</sup>

Brittany Kaiser, była pracowniczka firmy Cambridge Analytica, w filmie dokumentalnym *The Great Hack* (reż. Karim Amer, Jehane Noujaim, USA 2019) przyznała, że celem algorytmu OCEAN było wpłynięcie na przekonania polityczne osób, które posiadały skłonność do zmiany zdania. Tego rodzaju jednostki były określane przez twórców algorytmu mianem osób „urabialnych” (*persuadables*). Jak widać, algorytm OCEAN posłużył w tym przypadku za narzędzie inżynierii społecznej, a więc intencjonalnych oddziaływań, które mają na celu manipulację społeczną. Politolog, Łukasz Scheffs definiuje pojęcie inżynierii społecznej w następujący sposób:

Inżynieria społeczna jest więc *de facto* procedurą doboru odpowiednich środków socjotechnicznych. Socjotechnika z kolei stanowi zbiór technik i wiedzy o łańcuchach przyczynowo-skutkowych zachowań i postaw ludzkich, którymi system sterujący, oddziałując na system sterowany posługuje się, aby osiągnąć wytyczone przez inżynierię społeczną cele.<sup>338</sup>

---

<sup>335</sup> Działanie algorytmu OCEAN wyjaśnił dyrektor firmy Cambridge Analytica podczas wystąpienia w ramach Concordia Summit w 2016 roku: Alexander Nix, *Cambridge Analytica – The Power of Big Data and Psychographics*, 27.09.2016, online: <https://www.youtube.com/watch?v=n8Dd5aVXLCC>, dostęp: 06.11.2023.

<sup>336</sup> K. Wilber, *Trump and a Post-Truth World*, Shambala, Boulder 2017, EPUB, cz 1, rozdz. 3.

<sup>337</sup> L. McIntyre, *Post-Truth*, MIT Press, Cambridge, Mass. 2018, s. XIV.

<sup>338</sup> Ł. Scheffs, *Socjotechnika władzy*, „Przegląd Politologiczny” 2016, nr 1, 35-48.

Trzeba tu zaznaczyć, że źródła rozwoju socjotechnik poprzedzają erę cyfrową, ponieważ koncepcje związane z określaniem typów osobowości odznaczające się podatnością na manipulacje były rozwijane już od przełomu XIX i XX wieku, głównie dzięki powstaniu słynnej książki francuskiego psychologa społecznego, Gustave'a Le Bona, *Psychologia tłumu: studium powszechnego umysłu*, do której odnosili się między innymi Zygmuntem Freud<sup>339</sup> oraz Theodore Adorno.<sup>340</sup> Temat kontrolowania ludności był w okresie rozwijania się kultury masowej bardzo popularny na gruncie naukowym. Warto jednak w tym kontekście zwrócić uwagę również na znaczenie determinantów kulturowych. Początek XX wieku był okresem rozwoju technologii radia, pierwszego symbolu kształtującej się kultury masowej. Wynalazek ten był jednym z mediów wykorzystywanych przez faszystowską propagandę, jest to więc czas rozwijania się strategii inżynierii społecznej zapośredniczonej technologicznie.

Przykład działania firmy Cambridge Analytica pokazuje konsekwencje połączenia dwóch nauk: technologii informatycznej z psychologią, które doprowadziło do wytworzenia zautomatyzowanych technik psychograficznych. Dzięki tej syntezie możliwe było stworzenie algorytmu, który działał w szerokiej rozpiętości, jednocześnie analizował olbrzymie zbiory danych dotyczące mas oraz dostosowywał treści manipulacyjne do profili psychologicznych pojedynczych jednostek. Zasadnicze rozróżnienie, które należy wprowadzić pomiędzy omawianymi tu technologiami cyfrowymi i analogowymi opiera się na tym, że w przeciwieństwie do strategii manipulacyjnych początków epoki mediów masowych, które starały się łączyć jednostki w łatwy do kontroli tłum, media społecznościowe charakteryzują się odwrotną tendencją. Mają one na celu przekształcenie

---

<sup>339</sup> Freud już w 1921 roku, a więc przed rozpowszechnieniem się ideologii faszyzmu, wydał książkę *Massenpsychologie und Ich-analyse*, która rok później została przetłumaczona na język angielski. Polskie wydanie tej pracy, zatytułowane *Psychologia zbiorowości i analiza ego* zostało zawarte w zbiorze *Poza zasadą przyjemności*, tłum. J. Prokopiuk, Wydawnictwo Naukowe PWN, Warszawa 1976. Na rozpoznaniach Freuda oparty został esej Adorno *Freudowska teoria i model propagandy faszystowskiej*, w polskim wydaniu zawarty w książce *Przemysł kulturalny*, dz. cyt.

<sup>340</sup> Theodor Adorno w oparciu o rozważania Le Bona opracował koncepcję „osobowości autorytarnej”. Termin ten powstał jako efekt pracy badawczej zespołu (w jego skład wchodził jeszcze: Else Frenkel Brunswik, Daniel J. Levinson i R. Nevitt Sanford), który przeprowadził w Berkeley, w Stanach Zjednoczonych szeroko zakrojone badania, wykorzystujące metody ilościowe i jakościowe, ankiety oraz wywiady pogłębione, których celem było scharakteryzowanie antysemickich tendencji i sposobu ich rozwijania się u ludzi. Opracowana przez zespół badawczy „skala F” pozwalała wyróżnić jednostki wpisujące się w model osobowości autorytarnej. Reprezentant szkoły frankfurckiej za pomocą tego terminu omawiał pewnego rodzaju typ człowieka, który wykazuje silne predyspozycje do przyjmowania przekonań o charakterze antydemokratycznym. Specyfiką tego typu personalnego miała być skłonność do wyrzekania się subiektywnej oceny świata i łatwość w uleganiu przekonaniom osób autorytarnych, usprawiedliwianie własnych działań stanowiskiem innych, wysoko postawionych lub znanych ludzi, co najczęściej było bezpośrednio związane z wychowaniem i relacjami rodzinnymi, a konkretnie z dominującym, karzącym ojcem. Zob. T.W. Adorno, *Osobowość autorytarna*, tłum. M. Pańków, Wydawnictwo Naukowe PWN, Warszawa 2010, s. 45.

mas w ogromny zbiór indywidualności po to, aby skuteczniej dostosowywać przekaz medialny i, co za tym idzie, manipulację, do potrzeb i osobowości jednostki. Wraz z powstaniem mediów masowej komunikacji, pojawiły się także strategie propagandowe, które działały za ich pośrednictwem. Wypracowane wcześniej w kulturze metody manipulacji społecznej są żywe również współcześnie, przybrały jednak o wiele bardziej zaawansowaną postać.

We wprowadzeniu do niniejszego rozdziału starałam się na przykładzie analogowej technologii radia przedstawić to, że platformy komunikacyjne mają tendencję do swobodnego rozwijania się, zaś organ prawny, starający się sprawować kontrolę nad kierunkiem ich rozwoju i bezpieczeństwem użytkowników, podąża śladem zmian przez nie powodowanych. Działalność firmy Cambridge Analytica jeszcze w 2016 roku nie była uznawana za nielegalną i z tego powodu zarządzający nią w ogóle nie taili swojej strategii działania, a wręcz promowali ją w spotach marketingowych. Jednak, kiedy okazało się, że powszechnie dostępna platforma, zrzeszająca użytkowników z całego świata podejmuje działania nieetyczne, naruszające prawa jednostek z niej korzystających, coraz silniej zaczęło wybrzmiewać pytanie o to, kto jest za taki stan rzeczy odpowiedzialny? W kontekście wcześniej omawianego skandalu związanego z Yahoo, kwestia odpowiedzialności wydawała się bardziej jasna: przed sądem stanął dyrektor wyszukiwarki internetowej i postawione zostało przed nim wymaganie znalezienia technologicznego rozwiązania, które uniemożliwi francuskim obywatelom wyszukiwania aukcji dotyczących przedmiotów o nazistowskim charakterze. Jednak w przypadku działalności Cambridge Analytica problem ten okazuje się o wiele trudniejszy do uchwycenia. CA zbierała dane za pośrednictwem kilkunastu platform i serwisów internetowych. Facebook był tylko jednym z nich. Wzięcie udziału w quizie internetowym, który CA umieściła na tym portalu społecznościowym wymagało zgody użytkownika, także można uznać, że uczestniczący zgłaszali się dobrowolnie. Badacz Ken Ward zaproponował, aby w kontekście nieoczywistej sytuacji związanej z CA, odwoływać się do kantowskiej teorii imperatywu kategorycznego<sup>341</sup>. Ward w upraszczający sposób przeniósł odpowiedzialność na portal społecznościowy jako taki, tak jakby była to jednostka odpowiadająca za siebie, a nie sieć połączonych elementów technologicznych i ludzkich. Pisał on, że: „Jeśli serwis internetowy

---

<sup>341</sup> Imperatyw kategoryczny Kanta to zasada etyczna, którą można sprowadzić do formuły: „należy postępować zawsze wedle takich reguł, co do których chcielibyśmy, aby były one stosowane przez każdego i zawsze”. Termin filozoficzny utworzony przez Immanuela Kanta jako skutek poszukiwań bezwzględnie ważnego prawa, dotyczącego każdej osoby, które byłyby spełniane niezależnie od istnienia Boga. Zob. Kant w: W. Tatarkiewicz: *Historia Filozofii*. T. 2. Warszawa: PWN, 1988, s. 161–184.

rozważa etyczne konsekwencje zbierania i sprzedawania danych swoich użytkowników, może na przykład zapytać, czy byłoby moralne, aby wszystkie serwisy internetowe tak robiły.”<sup>342</sup> Zdanie to wyraża dość powszechne przekonanie, że serwis internetowy można traktować podmiotowo, że można zrzucić na niego odpowiedzialność za własne działanie. Jednak należałoby tutaj doprecyzować to, kim właściwie jest serwis społecznościowy? Czy mowa tu o zespole technicznym, który go tworzył, o użytkownikach, którzy są odpowiedzialni za dominującą część treści w nim zawartych, czy o zarządzie firmy, którego platforma jest własnością? W powszechnym rozumieniu, często za konsekwencje działania technologii cyfrowych oskarża się osoby, które tworzyły stojące za nimi oprogramowanie. Zgodnie z takim rozumieniem, za portal społecznościowy należałoby uważać zespół programistyczny.

Tego rodzaju pogląd wyrażony został również przez Tristana Harrisa, osobę odpowiedzialną za kwestie etyczne w firmie Google, który powiedział: „nie znam rodziców, którzy mówią: <<Chcę aby moje dzieci dorastały i były manipulowane przez inżynierów technologicznych>>”.<sup>343</sup> Pomijając już kwestię tego, że sama jakość retoryki tej wypowiedzi jest raczej wątpliwa, ponieważ trudno wyobrazić sobie rodziców wypowiadających tak absurdalne zdanie, warto tu jednak zwrócić uwagę na to, że były pracownik Google określił oddziaływanie na użytkowników jako intencjonalną aktywność programistów. Wypowiedź ta jasno wskazuje na to, kto jest odpowiedzialny za manipulację za pośrednictwem platform cyfrowych. Oskarżenie zostało tu skierowane przeciwko zespołowi technicznemu. Warto tutaj dopowiedzieć, że serwis Facebook zatrudnia dwadzieścia siedem tysięcy pracowników, podzielnych na specjalistyczne zespoły, które przyczyniają się do jego rozwoju i działania. Oprogramowanie ma charakter społeczny, również w tym sensie, że nie jest dziełem jednego człowieka, ale musi powstawać jako efekt pracy zespołowej i wymiany pomiędzy różnymi platformami cyfrowymi. Justin Rosenstein, twórca słynnego przycisku „Lubię to” przyznał, że projektując tę funkcjonalność był przekonany, że robi coś dobrego, że przekaże społeczeństwu sposób na dzielenie się pozytywnymi emocjami. Nie spodziewał się, że jego projekt stanie się podstawą mechanizmu uzależniającego od aktywności na portalach społecznościowych oraz przyczyną depresji osób nastoletnich.<sup>344</sup>

---

<sup>342</sup> K. Ward, dz. cyt, s. 137.

<sup>343</sup> *Dylemat społeczny*, reż. Jeff Orlowski, USA 2020.

<sup>344</sup> T. Embury-Dennis, *Man who invented 'Like' button deletes Facebook app over addiction fears*, 06.10.2017, online: <https://www.independent.co.uk/life-style/gadgets-and-tech/facebook-inventor-deletes-app-iphone-justin-rostein-addiction-fears-a7986566.html>, dostęp: 06.11.2023.

Perspektywa człowieka, który tworzy niewielki moduł aplikacji cyfrowej jest znacząco ograniczona, skupia się on na zadaniu, które jest jego pracą, jego celem nie jest stworzenie narzędzia do manipulacji społecznej, nawet jeśli faktycznie się do tego przyczynia. Nie można jednoznacznie przypisać winy za sposób funkcjonowania portali społecznościowych osobom technicznym, które je tworzą. W przypadku zrzeszającego ogromną liczbę pracowników portalu społecznościowego z wielopoziomą hierarchią zarządzania dochodzi do zjawiska rozproszenia odpowiedzialności. Trudno szukać winy w osobie wypełniającej swoje zadania, a idąc tropem relacji między przełożonym a pracownikiem dojść można do dyrektora naczelnego firmy. Podczas przesłuchania w sprawie udziału platformy Facebook w skandalu związanym z CA, Mark Zuckerberg wziął odpowiedzialność za nadużycia prywatności, w których jego platforma pośredniczyła, mówiąc, że: „Nie spojrzeliśmy wystarczająco szeroko na naszą odpowiedzialność i to był wielki błąd. I to był mój błąd. I przykro mi z tego powodu. Ja założyłem Facebooka, ja go prowadzę i ja jestem odpowiedzialny za to, co się w jego obszarze dzieje.”<sup>345</sup> Wobec rozproszonej odpowiedzialności twórców portalu społecznościowego, powołany, aby stanąć przed sądem, został założyciel platformy, który jako jedna osoba odpowiedzial za cały, kilkudziesięciotysięczny zespół.<sup>346</sup>

Metody te i strategie powiązane są bezpośrednio z modelem ekonomicznym, który był prymarny wobec serwisu. Pozwalał on platformie się utrzymywać, przyczyniał się do jej powszechności i umożliwiał, jak do tej pory, darmowy dostęp dla użytkowników.<sup>347</sup> Jednak należy zwrócić uwagę, że ponad dyrektorem zarządzającym platformą istnieje jeszcze jedna instancja, która go kontroluje i która wpłynęła bezpośrednio na to, w jaki sposób portal został zaprojektowany. Jest nią model ekonomiczny, pozwalający serwisowi się utrzymywać, dzięki któremu jest on powszechnie dostępny i darmowy. Facebook, jak większość usług cyfrowych, które nie wymagają opłat za korzystanie z nich, utrzymuje się z wyświetlania reklam oraz płatnego pozycjonowania postów. Ten model zarabiania pieniędzy stał się podstawą zjawiska określanego mianem „ekonomii uwagi”.<sup>348</sup> Termin ten określa szereg wzorców ekonomicznych, które wykształciły się w wobec nadmiaru produkowanych informacji. Sama informacja i jej rzetelność straciły na wartości,

---

<sup>345</sup> *Facebook CEO Mark Zuckerberg testifies before Congress on data scandal*, dz. cyt. Tłumaczenie A.P.

<sup>346</sup> Należy tu podkreślić, że Mark Zuckerberg wybronił się podczas procesu sądowego, nie poniósł żadnych znaczących konsekwencji prawnych oraz, że to sędziowie i prawnicy go przepytujący ośmieszili się, ponieważ wykazali się brakiem podstawowej wiedzy związanej z działaniem platform społecznościowych.

<sup>347</sup> Od listopada 2023 roku serwis Meta wprowadził możliwość płatnej subskrypcji, do czego będę jeszcze nawiązywać w kolejnych rozdziałach i w zakończeniu pracy.

<sup>348</sup> B.A. Huberman, *Social Computing and the Attention Economy*, „Journal of Statistical Physics” 2013, nr 1(151), s. 330.

najcenniejsza natomiast okazała się właśnie uwaga użytkownika zatopionego w szumie informacyjnym. Z tego powodu koncentracja ludzi korzystających z darmowych portali społecznościowych staje się przedmiotem utowarowienia i monetyzacji, to za nią płać reklamodawcy, dzięki którym serwisy takie jak Facebook zarabiają. Zależność tę doskonale wyraził Aza Raskin, były pracownik Firefox&Mozilla Labs, w słowach: „Jeżeli nie płacisz za produkt, to sam jesteś produktem”.<sup>349</sup> Człowiek korzystający z sieci w celu uczestniczenia w życiu społecznym i komunikacji z innymi po części staje się przedmiotem wymiany towarowej. Dzięki tak ustrukturyzowanemu modelowi ekonomicznemu, Cambridge Analytica i algorytm OCEAN mógł zyskać tak dużą siłę oddziaływania i skuteczność. Ekonomia uwagi ujawnia wrażliwość, niestabilność oraz kruchość demokracji, ponieważ system ten przestaje być rządami większości ludu, ale tego, kto wie, jak kontrolować tłumy. Podobne przesłanie płynęło z przytaczanej już książki le Bona z końca XIX wieku ujawniającej mechanizmy psychologiczne rządzące zgromadzeniami. Strategie inżynierii społecznej pozostają żywe w kulturze od dawna, a wolność działania jednostki okazuje się złudna. Współcześnie, odkrywanie, że media cyfrowe wykorzystują modele perswazji i manipulacji, wywołuje duże oburzenie i doprowadza do skandali, choć nie powinno to dziwić, ponieważ cała gałąź określana mianem HCI (*Human-Computer Interaction*) oparta jest na gruncie nauk psychologicznych, szczególnie na badaniach dotyczących procesów poznawczych i zachowań człowieka<sup>350</sup>. Interfejsy interakcji człowieka z komputerem były projektowane bardzo świadomie, z ukierunkowaniem na to, w jaki sposób człowiek myśli, postrzega i doświadcza. Z założenia miały przecież oddziaływać na ludzką psychikę, co przedstawiałam w poprzednim rozdziale.

#### 4.4 Software jako broń

W ramach rozszerzenia rozważań dotyczących politycznych manipulacji za pomocą narzędzi cyfrowych, należy również przywołać specyficzny rodzaj kontroli tłumy, który może być rozumiany jako działania wojenne i polega na wojnie informacyjnej. Tego rodzaju działania określa się terminem „operacji psychologicznych” (PsyOps) i należą do nich: wojna elektroniczna, operacje w sieciach komputerowych i oszustwa, które mają na celu

---

<sup>349</sup> *Dylemat społeczny*, dz. cyt.

<sup>350</sup> T.K. Landauer, *Behavioral Research Methods in Human-Computer Interaction*, w: *Handbook of Human-Computer Interaction*, red. M.G. Helander, T.K. Landauer, P.V. Prabhu, North Holland, Nowy Jork 1997, s. 203-227.

wpływanie na emocje, motywacje, światopogląd i zachowanie odbiorców.<sup>351</sup> Należy podkreślić, że jest to od dawna stosowana taktyka militarna, co potwierdzają chociażby podręczniki wojskowe z czasów zimnej wojny.<sup>352</sup> Współcześnie taktyka, wzmocniona narzędziami cyfrowymi, zyskała większą siłę oddziaływania. SCL Group prowadziła również działania na rzecz instytucji wojskowych. Cambridge Analytica w swojej ofercie reklamowej głosiła, że prowadzi szkolenia dla brytyjskiej armii oraz marynarki, wykorzystując hasło: „Wykorzystujemy badania, by wpływać na zachowanie wrogów.”<sup>353</sup> Do innych działań firmy zajmującej się zarządzaniem danymi należą operacje psychologiczne w ramach wojny w Afganistanie. CA wykorzystywała techniki mikrotargetingu w celu podprogowego przekonywania młodych muzułmanów, aby nie wstępowali do Al Kaidy.<sup>354</sup>

Jedną z głośniejszych spraw dotyczących stosowania taktyk wojny informacyjnej za pomocą platformy Facebook jest historia związana z konfliktem religijno-społecznym, który dzieli państwo Mjanma (dawniej Birma). Spór toczy się pomiędzy muzułmańskimi Rohingami a buddyjskimi Rakhinami. Dominującą religią w tym państwie jest buddyzm. Facebook jest aplikacją domyślnie zainstalowaną na telefonach w wielu krajach, co dotyczy również Mjanmy, dlatego większość tamtejszych obywateli korzysta na co dzień z tej platformy społecznościowej. Wykorzystał to rząd Mjanmy i za pomocą tego serwisu manipulował opinią publiczną, podżegał do przemocy, krzewił uprzedzenia do muzułmańskich Rohingów. Skutkiem tych operacji psychologicznych było wywołanie zamieszek, a także masowych zabójstw, palenia wiosek, w efekcie czego około siedemset tysięcy Rohingów musiało ratować się ucieczką z kraju.<sup>355</sup> Był to przerażający w skutkach napływ fali nienawiści, która wylała się na portalu społecznościowym. Również w tym przypadku Mark Zuckerberg, jako właściciel platformy został pociągnięty do odpowiedzialności i zobligowany do usunięcia fałszywych kont oraz stron propagujących nieprawdziwe informacje i mowę nienawiści w Mjanmie.<sup>356</sup>

---

<sup>351</sup> E.L. Briant, *Propaganda and Counter-Terrorism: Strategies for Global Change*, Manchester University Press, Manchester 2015, s. 23.

<sup>352</sup> C. Simpson, *Science of Coercion: Communication Research and Psychological Warfare 1945-1960*, Oxford University Press, Oxford 1994.

<sup>353</sup> *The Great Hack*, dz. cyt.

<sup>354</sup> Tamże.

<sup>355</sup> S. Steckflow, *Why Facebook is losing the war on hate speech in Myanmar*, 15.08.2018, online: <https://www.reuters.com/investigates/special-report/myanmar-facebook-hate/>, dostęp: 06.11.2023.

<sup>356</sup> P. Mozur, *A Genocide Incited on Facebook, With Posts From Myanmar's Military*, 15.10.2018, online: <https://www.nytimes.com/2018/10/15/technology/myanmar-facebook-genocide.html>, dostęp: 06.11.2023.

Jeśli operacje psychologiczne przeprowadzane za pomocą narzędzi cyfrowych zaliczane są do działań wojennych, oprogramowanie może być w takim wypadku rozumiane jako rodzaj broni. Christopher Wylie opisując działania CA, używał określenia „psychologiczna broń masowej destrukcji” (*psychological weapon of mass destruction*).<sup>357</sup> Cathy O’Neil w swojej książce poświęconej analizie wpływu, jaki oprogramowanie i sztuczna inteligencja mają na społeczeństwo, stosuje ciekawą grę słowną stosując zwrot „broń matematycznej zagłady” (*weapons of math destruction*).<sup>358</sup> Badacz i badaczka używali tych określeń nie w kontekście oprogramowania militarnego, ale powszechnych systemów cyfrowych, które funkcjonują w społeczeństwie, i z których codziennie korzystają miliony użytkowników. W oparciu o omawiane w tym rozdziale przykłady oddziaływań politycznych zapośredniczonych przez platformy społecznościowe, daje się zauważyć, że wykorzystywanie oprogramowania jako broni w wojnie informacyjnej prowadzi do zacierania się granic pomiędzy zagrożeniem zewnętrznym a wewnętrznym. Oprogramowanie dostarczyło narzędzi do tego, aby wojnę informacyjną prowadzić w sposób dyskretny oddziałując przy tym na szeroką skalę. Agresor działa z ukrycia, manipulując jedynie treściami i informacjami w obiegu komunikacyjnym. Informacja zawsze miała ogromną wartość w działaniach wojennych, a oprogramowanie dostarczyło nowych strategii zarówno zdobywania, jak i sabotowania informacji. Przez to współcześnie wzrosło znaczenie metod innych niż czysto militarne. Walka przybrała charakter ideowy.<sup>359</sup>

Wykorzystywanie oprogramowania do kreowania alternatywnej rzeczywistości przywodzi na myśl skojarzenie z przywoływaną już przeze mnie w pierwszym rozdziale koncepcją retoryki proceduralnej Iana Bogosta. Pojęcie to odnosiło się do tego, jak za pomocą procesów obliczeniowych generuje się modele perswazji, struktury symboliczne i interpretacje, które posiadają perswazyjną moc przekonywania o swojej prawdziwości, ponieważ potwierdzają się w ramach sztucznie wytworzonego świata wirtualnego – w przypadku rozważań Bogosta był to świat gry.<sup>360</sup> Można dostrzec tu podobieństwa w funkcjonowaniu oprogramowania, które przyczynia się do tworzenia alternatywnych rzeczywistości w ramach działań z obszaru wojny informacyjnej, które również mają na celu stworzyć perswazyjny model, który będzie oddziałował na odbiorcę, prezentując mu

---

<sup>357</sup> C. Wylie, *Mindf\*ck: Cambridge Analytica and the Plot to Break America*, Random House, Nowy Jork 2019, EPUB, rozdz. 1.

<sup>358</sup> C. O’Neil, *Broń matematycznej zagłady. Jak algorytmy zwiększają nierówności i zagrażają demokracji*, tłum. M. Z. Zieliński, Wydawnictwo PWN, Warszawa 2017.

<sup>359</sup> J. Darczewska, *Diabeł tkwi w szczegółach. Wojna informacyjna w świetle doktryny wojennej Rosji*, Ośrodek Studiów Wschodnich im. Marka Karpia, Warszawa 2015, s. 20-22.

<sup>360</sup> I. Bogost, *Persuasive Games...*, dz. cyt., s. IX.

określony obraz świata. Poprzez koncepcję retoryki proceduralnej Bogost podkreślał ideologiczne oddziaływanie oprogramowania w kontekście wytwarzania rzeczywistości wirtualnej. W odniesieniu do wojny informacyjnej ta cecha jest również wyraźna w strategiach tworzenia alternatywnej rzeczywistości lub też, korzystając z określenia zaproponowanego przez Petera Pomerantseva i Michaela Weissa, nierzeczywistości (*unreality*).<sup>361</sup> Nierzeczywistość jest takim przedstawieniem świata, w ramach którego potwierdzają się post-prawdy i oddziałują w sposób ideologiczny na odbiorców manipulacyjnych komunikatów, realizując przy tym polityczne cele.

#### 4.5 Transparentność i edukacja jako opór

Portale społecznościowe powstały aby realizować społecznie cele, jakimi są: umożliwienie komunikacji międzyludzkiej, wymiany myśli, informacji i niesienia wzajemnej pomocy. Nie uchroniło ich to jednak przed wykorzystywaniem jako narzędzie politycznej propagandy, czy broni w ramach walki informacyjnej. Mimo że pośredniczą one w sprawowaniu kontroli społecznej, jednocześnie mogą stać się przestrzenią umożliwiającą różnorodne formy oporu.

Adorno podkreślał, że intencjonalne stosowanie technik zarządzających społeczeństwem doprowadza do ich wyczerpania, w efekcie czego przestają być skuteczne. Badacz pisał, że: „Kiedy przywódcy stają się świadomi psychologii mas i zaczynają nią sterować, w pewnym sensie przestaje ona istnieć.”<sup>362</sup> Adorno porównywał to zjawisko do techniki hipnozy wykorzystywanej przez Freuda, która była skuteczna o tyle, o ile hipnotyzowany nie miał świadomości jej działania.<sup>363</sup> Jak już było opisywane we wstępie do niniejszej dysertacji, również ideologia, działa w taki sposób, że jeśli zostanie nazwana i opisana, pozwala analizującemu nabrać dystansu do niej, uwolnić się spod jej działania dzięki krytycznemu ustosunkowaniu.<sup>364</sup> Z tego powodu strategią przeciwdziałania taktykom manipulacji zapośredniczanej przez platformy cyfrowe jest ich opisywanie, rozpowszechnianie wiedzy na ich temat, a więc działania edukacyjne mające na celu kształtowanie świadomych użytkowników.

---

<sup>361</sup> P. Pomerantsev, M. Weiss, *The Menace of Unreality: How the Kremlin Weaponizes Information, Culture and Money*, The Institute of Modern Russia, Nowy Jork 2014, s. 34-35.

<sup>362</sup> Theodore Adorno, *Przemysł kulturalny*, dz. cyt., s. 209.

<sup>363</sup> Tamże, s. 211.

<sup>364</sup> Alexander R. Galloway, *Język chce by...*, dz. cyt., s. 24.

W kontekście omawianych tutaj problemów, ciekawym przedsięwzięciem o charakterze edukacyjnym jest projekt *Propaganda Machine: Inside Cambridge Analytica and the Digital Influence Industry*,<sup>365</sup> który został przygotowany przez badaczkę Emmę L. Briant, która brała udział w ujawnianiu skandalu związanego z CA. Jest on chronologiczną mapą, pozwalającą na śledzenie tego, w jakim czasie i w jakim obszarze geograficznym Cambridge Analytica realizowała swoje przedsięwzięcia polegające na zbieraniu danych, szerzeniu dezinformacji i społecznej manipulacji. Mapa zawiera również linki do zewnętrznych artykułów, które opisują konkretne projekty CA i które pozwalają dokładnie prześledzić działalność firmy. Wizualna forma projektu i możliwość dostosowania chronologii wyświetlanych aktywności w skuteczny sposób przedstawia skalę zjawiska. *Propaganda Machine...* została zaprojektowana z myślą o aktywistach, badaczach, działaczach pozarządowych, twórcach regulacji prawnych oraz dziennikarzach po to, aby ułatwić im zdobywanie wiedzy na temat działalności firmy Cambridge Analytica. Nie jest to więc projekt skierowany bezpośrednio do użytkowników internetu, ale do ich edukatorów.

Innym przykładem, usytuowanym na gruncie sztuki nowych mediów, jest instalacja interaktywna *Facebook Algorithmic Factory*, prezentowana po raz pierwszy podczas biennale WRO we Wrocławiu w 2017 roku. Autorami projektu byli: Katarzyna Szymielewicz, Vladan Joler oraz Paweł Janicki. Zespół ten poświęcił się analizie algorytmów działających na platformie Facebook, a efekty swoich badań zaprezentował w postaci interaktywnego grafu wzbogaconego o narracje użytkowników. W opisie kuratorskim autorzy sformułowali tezę podobną do przedstawionej na początku tej części dysertacji, która uznaje edukację i poszerzanie wiedzy za formę oporu przeciwko kontroli i manipulacji zapośredniczonej przez platformy cyfrowe:

Głęboko, pod warstwą algorytmicznego przetwarzania danych, kryją się nowe formy naruszeń praw człowieka, nowe mechanizmy wyzysku i manipulacji, których nie jesteśmy w stanie kontrolować. Aby stawić im opór, musimy je przede wszystkim poznać i ujawnić.<sup>366</sup>

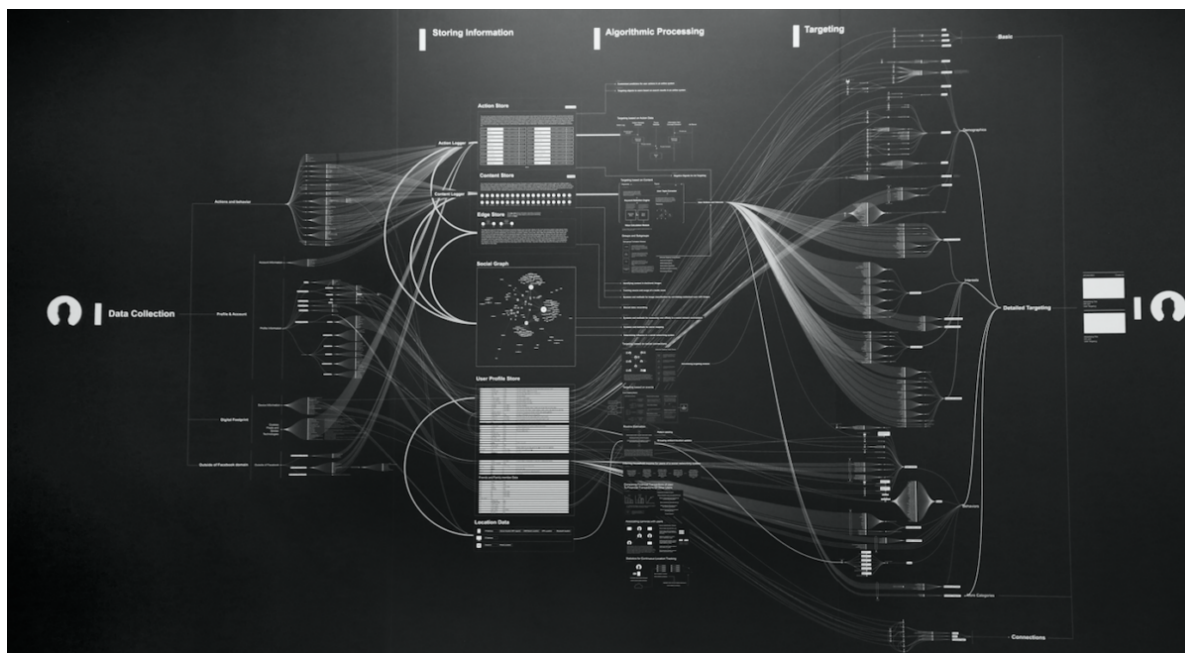
Instalacja przybliżała odbiorcom działanie algorytmów Facebooka, które decydują o tym, co zostanie mu wyświetlone, a co ukryte. Przedstawiała również sposób, w jaki oprogramowanie łączy w sieci zależności użytkowników, którzy wymieniają ze sobą interakcje, są dla siebie znajomymi lub znajdowali się w podobnym czasie w określonym

---

<sup>365</sup> E.L. Briant, *Propaganda Machine: Inside Cambridge Analytica and the Digital Influence Industry*, 2020, online: <https://www.propagandamachine.tech/ca-map>, dostęp: 06.11.2023.

<sup>366</sup> *Facebook Algorithmic Factory*. Katarzyna Szymielewicz, Vladan Joler, Paweł Janicki, 18.05.2017, online: <https://wro2017.wrocenter.pl/facebook-algorithmic-factory>, dostęp: 06.11.2023.

miejscu, co pozwala na określenie ich preferencji politycznych oraz statusu ekonomicznego, a następnie przekazanie tych informacji zainteresowanym reklamodawcom (Il. 13). Instalacja ujawniała działanie czarnej skrzynki, którą jest Facebook, rozpowszechniając informacje niemożliwe do zdobycia bez przeprowadzenia odpowiednich badań.



Il. 13. *Facebook Algorithmic Factory*. Źródło: online:  
<https://wro2017.wrocenter.pl/facebook-algorithmic-factory>, dostęp: 06.11.2023.

Jako kolejny przykład cyfrowej edukacji społecznej posłuży tu działająca aktywnie na polskim gruncie fundacja Panoptikon. W skład jej zespołu wchodzi Katarzyna Szymielewicz, która należała do grupy twórców instalacji *Facebook Algorithmic Factory*. Fundacja działa pod hasłem „Kontrolujemy kontrolujących”, a jej celem jest nadzorowanie władz oraz prywatnych firm i zwiększenie kontroli społecznej nad ich działaniami.<sup>367</sup> Do tej pory fundacja złożyła setki wniosków do instytucji państwowych (ministerstw, urzędów, komend policji, kancelarii prawnych, komisji) o udostępnienie informacji publicznej.<sup>368</sup> Złożenie tego rodzaju wniosku jest prawem każdego obywatela, jednak nie każdy wie, że może to zrobić oraz w jaki sposób to przeprowadzić. Fundacja przede wszystkim stara się zdobyć informacje na temat tego, za pomocą jakich narzędzi cyfrowych, algorytmów,

<sup>367</sup> Zob. *O co walczymy*, online: <https://panoptikon.org/corobimy>, dostęp: 06.11.2023.

<sup>368</sup> Zob. *Wnioski o udostępnienie informacji publicznej*, online: <https://crm.panoptikon.org/wnioski-o-dip>, dostęp: 06.11.2023.

programów czy urzędów państwo polskie stara się zyskać kontrolę nad obywatelami. Odpowiedź instytucji państwowych na tego rodzaju wnioski zostaje opublikowana na stronie fundacji. Panoptykon udostępnia materiały edukacyjne w postaci podcastów, infografik oraz publikacji. Działalność organizacji wykracza poza granice Polski, ponieważ angażuje się ona również w konsultowanie powstających globalnych regulacji prawnych, związanych z technologiami cyfrowymi.<sup>369</sup> Przedstawiciele fundacji Panoptykon brali również czynny udział w niezwykle istotnym przedsięwzięciu, którym jest praca nad unijną ustawą Digital Service Act, która była konsultowana i opracowywana od 2021 roku. Regulacja ta, przygotowywana przez Komisję Europejską, wymierzona jest przede wszystkim przeciwko dominującym w obszarze komunikacyjnym gigantom spod znaku GAFA (Google, Amazon, Facebook, Apple). Obecnie wiedza pozyskiwana na temat użytkowników platform cyfrowych dzieli się na trzy kategorie: informacje, które użytkownik sam podał, informacje o aktywności użytkownika oraz informacje wywnioskowane na podstawie aktywności użytkownika (np. zmiana platformy do robienia zakupów z tej z tańszymi produktami na taką z bardziej luksusowymi, pozwala określić, czy dany użytkownik dostał awans lub zmienił pracę na lepszą). Jednym z postulatów, które organizacje broniące praw człowieka (w tym polski Panoptykon) kierują w ramach trwających konsultacji regulacji, jest zakazanie wykorzystywania wniosków wyciągniętych na podstawie aktywności użytkownika w sieci.<sup>370</sup> Wprowadzenie tego rodzaju ustawy ograniczyłoby wykorzystywanie informacji na temat użytkownika do tylko tych danych, które sam podał. Manipulacja dalej byłaby możliwa, ale zwiększyłaby się świadomość użytkownika na temat tego, które informacje, przez niego podane są wykorzystywane w celu dostosowywania widocznych dla niego treści. Digital Service Act został przyjęty jesienią 2022 i zacznie obowiązywać od pierwszego kwartału 2024 roku. DSA wykluczyło możliwość zbierania danych na temat stanu zdrowia, preferencji politycznych oraz wyznania religijnego, przy czym dalej będzie możliwe zbieranie innego rodzaju danych i dostosowywanie treści internetowych do profili psychometrycznych użytkowników. Prawo unijne ma ograniczyć władzę i kontrolę cybergigantów nad użytkownikami Internetu. Do najważniejszych obostrzeń wprowadzonych przez ten akt prawny należą: narzucenie jaśniejszych i bardziej spójnych zasad moderacji treści, zakaz wykorzystywania

---

<sup>369</sup> *Stanowisko Fundacji Panoptykon w sprawie Aktu prawnego o usługach cyfrowych (Digital Services Act)*, 20.01.2021, online: [https://panoptykon.org/sites/default/files/stanowiska/panoptykon\\_kprm\\_uwagi\\_do\\_dsa\\_20.01.2021.pdf](https://panoptykon.org/sites/default/files/stanowiska/panoptykon_kprm_uwagi_do_dsa_20.01.2021.pdf), dostęp: 06.11.2023.

<sup>370</sup> *Europa naprawia Internet. Czy to koniec dyktatu cyberkorporacji?*, 22.07.2021, online: <https://panoptykon.org/jak-dsa-zmieni-internet>, dostęp: 06.11.2023.

manipulacyjnych interfejsów, wprowadzenie ograniczenia dla śledzącej reklamy oraz wymóg pisania regulaminów prostym, zrozumiałym językiem.<sup>371</sup> DSA stanowi dowód na to, że prawo dotyczące cyberprzestrzeni wciąż się rozwija i realizuje cel ochrony użytkowników sieci.

## 4.6 Podsumowanie

W kontekście radia Brecht wyraził marzenie dotyczące medialnej komunikacji pozwalającej na dwustronną wymianę informacji i usytuowanie odbiorcy w roli twórcy treści kulturowych. Kiedy model ten został zrealizowany prawie sto lat później, pierwsza sieć internetowa, określana mianem Web 1.0 okazała się nieudanym projektem, opartym całkowicie na ekonomii, który doprowadził do załamania rynku. Następująca po nim Web 2.0, rzekomo oddająca rozwój nad siecią w ręce użytkowników, w istocie powtarzała modele ekonomiczne poprzednich epok. Od momentu wynalezienia radia, aż po rozpowszechnienie się mediów cyfrowych, pomimo zasadniczych różnic kulturowych, wyraźne było to, że kontrola nad modelem komunikacyjnym danej epoki technologicznej uwarunkowana była z jednej strony ekonomicznie, z drugiej zaś prawnie. Pierwszy z tych czynników, ekonomiczny, podkreślali Enzensberger oraz Adorno. Umasowienie produkcji, w tym również wytwarzanie treści kulturowych, za pomocą mediów realizowało postulaty systemu kapitalistycznego i utrzymywało odbiorcę w roli konsumenta. Model ten odtwarzany był wraz z każdą technologiczną redefinicją społecznego systemu komunikacyjnego. Drugi czynnik, prawny, opiera się na kontroli, którą posiadają instytucje, organy definiujące regulacje, którym muszą poddawać się dystrybutorzy i twórcy usług, obiektów, platform oraz aplikacji technologicznych.

Zgodnie z wyobrażeniem Brechta rozwiązaniem na problem bierności użytkownika w społecznym, medialnym systemie komunikacyjnym, jest umożliwienie odbiorcom nadawania treści kulturowych. Zrealizowanie tego wyobrażenia we współczesnej kulturze cyfrowej zdaje się udowadniać, że każdy model komunikacyjny, oparty na zasadach systemu kapitalistycznego dąży do izolacji użytkownika. W epoce analogowej izolacja ta polegała na uczestnictwie w kulturze ograniczonym do słuchania, oglądania, natomiast

---

<sup>371</sup> D. Głowacka, K. Szymielewicz, *DSA i DMA w pigułce – co warto wiedzieć o prawie, które ma poskromić cybergigantów*, 03.06.2022, online: <https://panoptikon.org/akt-o-uslugach-cyfrowych-dsa>, dostęp: 06.11.2023.

w epoce cyfrowej użytkownik, mimo możliwości samodzielnego tworzenia komunikatu, wyrażania się i kreowania treści kulturowych poddany jest procesom izolującym go w zamkniętej bańce informacyjnej<sup>372</sup>, która daje złudzenie obiektywnego poglądu na rzeczywistość, a w istocie jest dostosowanym przez algorytmy, zmanipulowanym komunikatem medialnym, którego treść pochodzi od jednostek, które zapłaciły za uwagę użytkownika. Sam użytkownik, jego skupienie, aktywność oraz dane zostały utowarowione i wystawione na sprzedaż w ogromnej sieci cyfrowej o społecznym oraz ekonomicznym charakterze.

Trudno zaproponować kolejne rozwiązanie, przeciwdziałające izolującemu charakterowi sieci społecznościowych, szczególnie przez symbiotyczny charakter zależności pomiędzy użytkownikiem tego rodzaju platform a ich istnieniem. Użytkownicy potrzebują platform po to, aby uczestniczyć w życiu społecznym, natomiast platformy nie mogą funkcjonować bez użytkowników, ponieważ to oni wytwarzają zasadniczą treść, która jest podstawą i warunkiem ich działania.<sup>373</sup> Amerykański informatyk, Jaron Lanier w filmie *Dylemat społeczny* powiedział, że rozwiązanie problemu związanego z mediami społecznościowymi, który sprowadza się do kwestii ekonomicznych, również powinno mieć taki charakter.<sup>374</sup> Warto jednak podkreślić, że model sieci Web 1.0 upadł właśnie z przyczyn ekonomicznych i w świetle powyższych ustaleń można przypuszczać, że będzie to czynniki ekonomiczne staną się powodem załamania Web 2.0.

Podsumowując rozważania zawarte w niniejszym rozdziale chciałabym podkreślić jeszcze raz, to że strategie kontroli użytkowników kształtowały się wraz z rozwojem analogowych mediów kultury masowej, co starałam się udowodnić na przykładzie radia. Zarówno w epoce technologii analogowych, jak i cyfrowych owa kontrola społeczna wyrażała się poprzez określanie miejsca i roli użytkownika w procesie wytwarzania i odbioru treści kultury masowej. Media analogowe nie pozwalały użytkownikom na swobodę transmisji treści a jedynie na ich pasywny odbiór. Działanie tego rodzaju technologii wyrażało się w tendencji do łączenia jednostek w masy, które później łatwo było

---

<sup>372</sup> Bańka informacyjna jest to zjawisko polegające na otrzymywaniu przez osobę korzystającą z sieci wiadomości, powiadomień, newsów i reklam wyselekcjonowanych przez algorytmy na podstawie danych zgromadzonych na temat użytkownika, na podstawie jego aktywności. Zob. J. Szpyt-Wiktorowska, *Strategie mediów wobec baniek informacyjnych*, „Zarządzanie Mediami” 2018, nr 6(1), s. 41.

<sup>373</sup> Wielu badaczy zwracało uwagę na symbiotyczny charakter tej relacji, chociażby Andrew D. Murray, omawiając regulacje prawne dla cyberprzestrzeni, korzystał z pojęcia regulacji symbiotycznych (*symbiotic regulations*) zob. A.D. Murray, *The Regulation of Cyberspace: Control in the Online Environment*, Milton Park, Abingdon, UK, and Nowy Jork, Routledge-Cavendish 2006, s. 240-244. W odniesieniu do tego pojęcia Paul Bernal opisał model „sieci symbiotycznej” (*The Symbiotic Web*) zob. P. Bernal, *Internet Privacy Rights. Rights to Protect Autonomy*, Cambridge, Cambridge University Press 2014, s. 54.

<sup>374</sup> „Rozwiązanie tego problemu powinno być opłacalne ekonomicznie”, w: *Dylemat społeczny*, dz. cyt.

kontrolować, co było wówczas możliwe w znacznej mierze dzięki radiu. Media cyfrowe pozwalają użytkownikowi na pewną ideę swobody, jednak niesie ona ze sobą wiele ograniczeń i niejawnych zależności. Strategie kontroli charakterystyczne dla mediów cyfrowych posiadają tendencje o przeciwnym kierunku niż opisane wyżej, specyficzne dla analogowych. Technologie wykorzystujące oprogramowanie mogą skutecznie manipulować pojedynczymi użytkownikami poprzez wyodrębnianie jednostek z masy i na przykład za pomocą technik psychometrii dostosowywać komunikat do poglądów i przekonań danej osoby. W tym działaniu społecznym ujawnia się rola oprogramowania w całym procesie kontroli, ponieważ to ono dostarcza narzędzi do tego, by działać jednocześnie na dwóch przeciwnych planach: z jednej strony na poziomie makro, w rozpiętości niemal niepoliczalnego zbioru danych, jakim jest ogół użytkowników, z drugiej zaś strony na poziomie mikro, pozwala scharakteryzować i zarządzać jednym elementem tego zbioru, jakim jest użytkownik.

### **III. Kontekst historyczny dla rozdziału piątego: rewolucje przemysłowe i programowalny obiekt robotyczny**

*Dziś, we w pełni uprzemysłowionej ekonomii, zapominamy o tym, jak bardzo pierwsza i druga rewolucja przemysłowa zmieniły społeczeństwo.*<sup>375</sup>

Chris Anderson

Wprowadzająca część historyczna dla rozdziału drugiego i trzeciego opisywała tendencje społeczne i technologiczne związane z poszukiwaniem zautomatyzowanych rozwiązań technologicznych na przełomie XVII i XVIII wieku. To właśnie postęp technologiczny w obszarze mechanizacji i automatyzacji technologicznej stał się bezpośrednią przyczyną rewolucji przemysłowej, a więc okresu silnych zmian ekonomicznych na świecie, którym towarzyszyły gruntowne transformacje gospodarcze. Niniejsze wprowadzenie kontekstowe ma na celu przybliżenie okoliczności związanych z początkiem globalnej industrializacji. Określenie „rewolucja przemysłowa” odnosi się do wzmożonej, częściowo zautomatyzowanej produkcji, możliwej przede wszystkim dzięki powstaniu takich wynalazków jak mechaniczne krosna<sup>376</sup>, które zrewolucjonizowały przemysł tkacki oraz maszyna parowa Jamesa Watta z 1763 roku, która w znaczący sposób wpłynęła na przemysł hutniczy i górniczy. Centralnym ośrodkiem dla zachodzenia tych rewolucyjnych procesów kulturowych była w tamtym czasie Wielka Brytania, gdzie oryginalnie powstały automatyczne krosna i maszyna parowa, i w której najprężniej rozwijały się fabryki i prawo patentowe.<sup>377</sup> To właśnie w tym kraju miały również miejsce pierwsze protesty przeciwko zmianie trwającego od ponad połowy tysiąclecia modelu ekonomicznego, opartego na produkcji rzemieślniczej. Celem niniejszej części wprowadzającej jest przedstawienie strategii społecznego oporu względem zmieniającego się modelu ekonomicznego u samego początku rozwoju kapitalizmu przemysłowego, co będzie potem stanowić istotne odniesienie dla omawiania współczesnych poszukiwań alternatywnego wytwarzania oprogramowania wobec panującego systemu kapitalistycznego.

---

<sup>375</sup> C. Anderson, *Makers. The New Industrial Revolution*, Crown Business, Nowy Jork 2012, s. 38. Tłumaczenie A.P.

<sup>376</sup> Najważniejszymi wynalazkami z tej grupy były: *Spinning Jenny* Richarda Hargreavesa z 1766, mechaniczne krosno Edmunda Cartwrighta z 1786 roku oraz automatyczne krosno Josepha Marie Jacquarda z 1805 roku, które sterowane było za pomocą kart perforowanych, a więc za pomocą tej samej metody, co późniejsze pierwsze maszyny obliczeniowe.

<sup>377</sup> Zob. J. Mokyr, *Editor's Introduction. The New Economic History and the Industrial Revolution*, w: *The British Industrial Revolution. An Economic Perspective*, red. tenże, Westview Press, Oxford 1999, s. 17-24.

Okres pierwszej rewolucji przemysłowej, a więc czas od początku XVIII do połowy XIX wieku, określa się powszechnie mianem epoki pary. Historyk William Rosen podkreślał zasadniczą rolę energii parowej jako węzła, który połączył ze sobą pozostałe innowacje technologiczne tego okresu.<sup>378</sup> Inżynier szkockiego pochodzenia, James Watt wynalazł maszynę parową w 1763 roku w celu osuszania kopalni, skąd wydobywano węgiel, używany chociażby w samej maszynie parowej. Późniejszy rozwój wynalazku sprawił, że zaczął być wykorzystywany w coraz szerszym obszarze gospodarczym. Rosen pisał o maszynie parowej, że nie była ona tylko automatyczną maszyną ruchu, ale zautomatyzowaną machiną innowacji, ponieważ przyczyniła się do powstania lokomotyw parowych, statków parowych, młynów parowych czy też parowych fabryk włókienniczych, gdzie napędzała dziesiątki automatycznych krosien.<sup>379</sup> Historyk podkreślał, że rewolucja przemysłowa nie sprowadzała się do tylko do stworzenia innowacyjnych wynalazków, ale była głęboką transformacją samego procesu wynalazczości, który pociągnął ze sobą kaskadę kolejnych technologicznych rozwiązań, które to z kolei diametralnie zmieniły dotychczasową sytuację ekonomiczną i przemysłową najpierw Anglii, a potem innych części świata.<sup>380</sup>

Jednak, kiedy na początku XVIII wieku w Anglii i Szkocji zaczęły pojawiać się pierwsze fabryki, w odbiorze części społeczeństwa natychmiast uznane zostały za zagrożenie dla utrzymania funkcjonowania pracowni chałupniczych i rzemieślniczych.<sup>381</sup> Wytwarzanie zapośredniczone przez zautomatyzowaną technologię pozwoliło na szybszą i tańszą produkcję i zaczęło stopniowo zyskiwać popularność. Wprowadzanie nowych rozwiązań technologicznych w tym przypadku nie miało jedynie charakteru zmiany sposobów produkcji, ale przyczyniło się do znaczącego przekształcenia panującego modelu ekonomicznego, i co za tym idzie, funkcjonowania społeczeństwa. Zmiany te spotkały się z poparciem korzystających z ich wprowadzenia kapitalistów i sprzeciwem rzemieślników oraz robotników. Industrializacja przemysłu włókienniczego w Wielkiej Brytanii spowodowała konflikt nie tylko pomiędzy zwolennikami innowacji a tradycjonalistami, ale również między dwoma różnymi wizjami ekonomii i społeczeństwa.<sup>382</sup> Istotne dla zrozumienia perspektywy obu stron sporu jest wzięcie pod uwagę tego, że każda z nich

---

<sup>378</sup> W. Rosen, *The Most Powerful Idea in the World: A Story of Steam, Industry, and Invention*, Random House, Nowy Jork 2010, EPUB, prolog.

<sup>379</sup> Tamże.

<sup>380</sup> Tamże.

<sup>381</sup> A.J. Randall, *The Philosophy of Luddism. The Case of the West of England Woolen Workers, ca. 1790-1809*, „Technology and Culture” 1986, nr 27(1), s. 1.

<sup>382</sup> Tamże, s. 5.

chroniła swojego kapitału i sposobów jego uzyskiwania. Konflikt społeczny, jaki wywołała pierwsza rewolucja przemysłowa, miał więc podłoże przede wszystkim ekonomiczne.

Powszechny ruch, który ukształtował się w Anglii na początku XIX wieku i zjednoczył tysiące rzemieślników, ale też hodowców i pasterzy, protestujących przeciwko wprowadzeniu automatyzacji, określa się mianem luddyzmu.<sup>383</sup> Należy tu jednak podkreślić, że niechęć społeczna wobec mechanizacji była wyrażana od co najmniej dwóch wieków przed uformowaniem się ruchu luddystycznego. Potwierdzeniem tego faktu jest to, że w 1551 roku Parlament Angielski uchwalił ustawę wprowadzającą zakaz używania mechanicznych młynów do przędzenia wełny, a wynalazca ramy dziewiarskiej w 1589 roku otrzymał nakaz opuszczenia Anglii. Poza wprowadzaniem prawnych regulacji niechęć do postępu technologicznego objawiała się w aktach agresji i toczyła się regularnie przez ponad dwieście lat, kiedy to dochodziło między innymi do takich spektakularnych działań, jak zniszczenie automatycznych maszyn przez tkaczy ze Spitalfields w 1675 roku czy też spalenie krosna *Spinning Jenny* Richarda Hargreavesa w 1767 oraz jego późniejszego modelu w 1769 roku.<sup>384</sup>

Jednym z najsłynniejszych wydarzeń, kojarzonych z aktywnością luddystów jest seria zamieszek mających miejsce w 1812 roku, w West Ride, w Yorkshire, kiedy to przeciwnicy technologizacji produkcji masowo włamywali się do fabryk i manufaktur, aby niszczyć, łamać i palić automatyczne krosna.<sup>385</sup> Przyczyną ich działań były lęki przed zachodzącą zmianą ekonomiczno-społeczną, które ogniskowały się w wizji bycia zastąpionym przez maszyny oraz utracie możliwości uzyskiwania przychodu z wykonywanego do tej pory zawodu. Paradoksalnie, agresywne działania luddystów doprowadziły do tego, że wartość maszyn musiała zostać postawiona na równi, a nawet przewyższać wartość ludzkiego życia, ponieważ aby powstrzymać organizację napadów na fabryki, wprowadzono regulacje prawne, które karały śmiercią za zniszczenie krosna. Uchwalona przez parlament brytyjski w 1812 roku ustawa *Destruction of Stocking Frame Act 1812*, zwana także *Frame-Breaking Act*, wyznaczała najwyższą karę za zniszczenie maszyny tkackiej i wyrażała ona założenie, że śmierć niewielkiej ilości luddystów jest dopuszczalna, jeśli ma się to przełożyć na ustabilizowanie się nastrojów społecznych, a co za tym idzie, wyeliminowanie ryzyka zagrożenia rozwoju gospodarczego.<sup>386</sup> W roku wprowadzenia ustawy stracono na jej mocy

---

<sup>383</sup> Tamże, s. 1.

<sup>384</sup> Wszystkie wymienione w tym akapicie zdarzenia historyczne opisywał historyk Rosen w: W. Rosen, dz. cyt., rozdz. 10.

<sup>385</sup> Tamże.

<sup>386</sup> Zob. K. Walters, *Law, 'terror' and the Frame-Breaking Act*, w: *The Economic History Society. Annual Conference*, Royal Holloway University of London 2004, s. 23-28.

siedemnastu mężczyzn. Najbardziej znacząca zmiana, jaką niosła ze sobą rewolucja przemysłowa dotyczyła statusu człowieka i technologii we wzajemnej relacji w przekształcającym się procesie produkcyjnym. Z jednej strony wartość maszyny uległa podwyższeniu i została zrównana z wartością ludzkiego życia – w myśl ustawy *Frame-Braking Act* karą za zniszczenie krosna była śmierć. Z drugiej strony, pozycja człowieka uległa degradacji, został on sprowadzony do roli narzędzia, operatora, części procesu produkcyjnego – rzemieślnicze, fachowe podejście, akcentujące podmiotowy wymiar pracy ustąpiło na rzecz myślenia o pracowniku jako o bezosobowym elemencie fabryki, który można łatwo wymienić lub zastąpić.<sup>387</sup>

Zapoczątkowany podczas pierwszej rewolucji przemysłowej proces zrównywania statusu człowieka i technologii w procesie produkcyjnym, znajdował odzwierciedlenie w kolejnych etapach postępu technologiczno-przemysłowego, a szczególnie podczas okresu kolejnej, drugiej rewolucji przemysłowej. Tym mianem określa się czas od połowy XIX do początku XX wieku.<sup>388</sup> Do najważniejszych innowacji tego okresu należały: silnik spalinowy Samuela Browna (1823), który zrewolucjonizował przemysł motoryzacyjny, telegraf elektromagnetyczny Samuela Morse'a (1837), który umożliwił przesyłanie informacji na odległość, dynamit Alfreda Nobla (1867), telefon Aleksandra Bella (1876), żarówka Thomasa Edisona (1879) oraz karabin maszynowy Hiram Maxima (1884). Produkcja na masową skalę żelaza i stali, możliwa dzięki opracowanej w okresie pierwszej rewolucji przemysłowej metodzie Bessemera i przyczyniła się do rozbudowy infrastruktury kolejowej, okrętów morskich, mostów, ale również technologii linii produkcyjnej, która natomiast stała się symbolem standaryzacji masowej produkcji, charakterystycznej dla tego okresu.<sup>389</sup>

Głównym ośrodkiem pierwszej rewolucji przemysłowej była Wielka Brytania, natomiast w kontekście drugiej rewolucji przemysłowej jako ośrodki przyczyniające się do postępu technologiczno-przemysłowego wskazuje się Amerykę, Wielką Brytanię, Niemcy, Francję i Japonię.<sup>390</sup> W Ameryce w 1903 roku powstała pierwsza fabryka samochodów Ford Motor Company założona przez Henry'ego Forda, w której powstawały modele Ford T, które szybko stały się jednymi z najbardziej popularnych samochodów tego okresu.

---

<sup>387</sup> Zob. M. McLeod, *Architecture or Revolution: Taylorism, Technocracy, and Social Change*, „Art Journal” 2014, nr 43(2), s. 132-134.

<sup>388</sup> R. Engelman, *The Second Industrial Revolution, 1870-1914*, online: <https://ushistoryscene.com/article/second-industrial-revolution>, dostęp: 05.11.2023.

<sup>389</sup> C. Anderson, dz. cyt., s. 38.

<sup>390</sup> R. Engelman, dz. cyt.

Czas drugiej rewolucji przemysłowej odznacza się powszechnością fabryk, standaryzacją produkcji masowej i udoskonalaniem jej strategii nie tylko pod względem technologicznym, ale i w kontekście zarządzania pracą, a nawet ciałem i ruchem samego robotnika. Wyrazem tych dążeń jest chociażby amerykańska metoda organizacji pracy zwana tayloryzmem. Opracowana ona została przez inżyniera, Fredericka W. Taylora na przełomie XIX i XX wieku. Celem metody Taylora była organizacja pracy robotnika w taki sposób, aby była ona jak najbardziej wydajna. Metoda opierała się na podzieleniu produkcji na mniejsze etapy, redukcji wszelkich niepotrzebnych czynności i ruchów, sprowadzaniu aktywności robotnika do maksymalnie produktywnego poziomu. Organizacja pracy miała wymiar nie tylko bezpośrednio fizyczny, ale opierała się również na oddziaływaniu na motywację pracownika poprzez wprowadzenie akordowego systemu płac.<sup>391</sup> Mimo że Taylor deklarował, że jego metoda ma na celu podwyższenie jakości życia robotników, krytykowana ona była za traktowanie pracowników jak „automatów” lub „maszyn”.<sup>392</sup>

Interesująca w tym kontekście jest bliskoźnaczność słów „robotnik” i „robot”. Słowa „robot” po raz pierwszy użył czeski pisarz, Karel Čapek, w wydany w 1921 roku dramacie „R.U.R.” Określenie to odnosiło się do rosyjskiego słowa „rabota” (работа), oznaczającego „praca”. Tekst ten nie tylko opisuje spełnienie się robotycznej apokalipsy i zagładę ludzkości, ale przede wszystkim wyraża to, że celem robota i powodem, dla którego został skonstruowany, jest wykonywanie pracy. Definicja ta nie zawiera w sobie podmiotowego podejścia do opisywanego obiektu. W dramacie Čapka roboty miały postać zantropomorfizowaną, a więc na poziomie wizualnym zostały upodobnione do człowieka, jednak nie przekładało się to na ich podmiotowe traktowanie, ponieważ odnosiło się do nich, jak do przedmiotu czy narzędzia. Obiekt robotyczny definiowany był przez to, co robi, a więc przez wykonywanie poleceń wydawanych przez człowieka, co najczęściej sprowadzało się do wyręczania go w monotonnej pracy. W podobny sposób ujmowany był robotnik w myśl metody Taylora. Został on sprowadzony do roli robota, który ma wykonywać powierzone mu zadania z maksymalną wydajnością.

Określenie „robot” różni się diametralnie od słowa „automat”, do którego odnosiłam się w części wprowadzającej do rozdziału trzeciego i czwartego. Starsze określenie, automat, opisywało obiekty technologiczne, mechanizmy, które przez swój sposób działania, zdawały się imitować życie. Nie były one tworzone po to, by wyręczać ludzi

---

<sup>391</sup> Zob. Tamże, s. 133.

<sup>392</sup> Zob. H.B. Dury, *Scientific management; a history and criticism*, Nowy Jork, Columbia University 1919, s. 195-198. Dostępny online: <https://archive.org/details/cu31924002406647/page/n23/mode/2up>, dostęp: 06.11.2023.

w pracy, ale by wyrazić fascynację nad imitacją procesów życiowych i intelektualnych za pomocą narzędzi technologicznych. Opisane przeze mnie wcześniej automaty nie były produkowane masowo, każdy z nich był unikalny i wyjątkowy. Za zwieńczenie rozważań na temat różnicy między modelem wyobraźniowym kryjącym się za określeniami „robot” i „automat” mogą posłużyć rozważania Karla Marksa, który był świadkiem drugiej rewolucji przemysłowej, i który również porównał robotników do mechanicznych części, wchodzących w skład większej całości, fabryki, którą z kolei określił mianem żywej maszyny:

Mamy tu, w miejsce odizolowanej maszyny, mechanicznego potwora, którego ciało wypełnia całe fabryki, a którego demoniczna moc, początkowo ukryta w powolnych i miarowych ruchach jego gigantycznych członków, w końcu wybucha w szybkim i gorączkowym wirze jego niezliczonych, pracujących organów.<sup>393</sup>

W opisie Marksa fabryki jawią się jako mechaniczny potwór składający się z ogromnej ilości pracujących organów.<sup>394</sup> Zgodnie z tą wizją o fabryce można myśleć jako o bycie hybrydycznym, na który składają się elementy technologiczne, ale także żywi, robotnicy, traktowani jednak jak część maszyn, przy których pracują. Poprzez bardzo obrazową metaforę Marks zatem określił status robotnika XIX wieku, który formułował się w relacji do technologii, stanowiącej zasadniczą część procesu produkcyjnego.

Dwa aspekty, charakterystyczne dla procesu produkcyjnego, kształtującego się wraz z kolejnymi rewolucjami przemysłowymi, będą dla mnie istotnym odniesieniem w kontekście współczesnym, kiedy będę opisywać status użytkownika technologii cyfrowych. Po pierwsze, robotnik zostaje zrównany z maszyną, łączy ich współuczestnictwo w procesie produkcyjnym, jest on traktowany jak maszyna, nawet sposób, w jaki wykonuje ruchy zostaje poddany kontroli tak, aby osiągnąć maksymalną wydajność. Po drugie, robotnik przestaje być traktowany jak podmiot przez proces, który Marks określał mianem alienacji.<sup>395</sup> Mimo wytwarzania dóbr, nie jest on ich właścicielem, ponieważ stają się one własnością kapitalisty, który osiąga dzięki nim zysk. W rezultacie praca wykonywana przez robotnika staje się czymś zewnętrznym wobec niego, nie należy do niego jako podmiotu,

---

<sup>393</sup> K. Marks, *Capital*, vol. 1, tłum. B. Fowkes, Penguin, Harmondsworth 1990, s. 503. Tłumaczenie A.P.

<sup>394</sup> Relacje pomiędzy cielesnością robotników a technologicznymi maszynami badał również Félix Guattari w podrozdziale *The Machine and the full body: the investments of the Machine* w: F. Guattari, *Chaosophy. Texts and interviews 1972-1977*, red. S. Lotringer, MIT Press, Cambridge, Mass. 2009, s. 109-115.

<sup>395</sup> K. Marks, *Rękopisy ekonomiczno-filozoficzne z 1844 r.*, w: Tenże, *Pisma wybrane. Człowiek i socjalizm*. PWN, Warszawa: 1979, s. 102.

sam robotnik nie należy do siebie w procesie produkcyjnym, ponieważ oddaje swoje ciało i jego wydajność komuś innemu.

Historyk, Ryan Engelman, podkreślał, że rewolucje przemysłowe są okresem ogromnych skrajności, z jednej strony bogactwa i postępu, z drugiej strony powszechnego ubóstwa i depresji, dla członków społeczeństwa, którzy mieli trudność w przystosowaniu się do zmieniającego się modelu ekonomicznego.<sup>396</sup> W czasie pierwszej rewolucji przemysłowej rzemieślnicy musieli zmierzyć się z faktem, że ich dotychczasowy sposób pracowania i zarabiania rozpada się i ustępuje miejsca nowym strategiom produkcji. Z kolei w czasie drugiej rewolucji przemysłowej robotnicy w fabrykach musieli doświadczać trudnych warunków wielogodzinnej pracy, za którą dostawali niskie wynagrodzenie, i która często prowadziła do różnego rodzaju obrażeń czy nawet kalectwa. Rewolucje przemysłowe przyniosły generalne korzyści kapitalistom, właścicielom fabryk i wykształconym robotnikom, którzy mogli liczyć na bardziej uprzywilejowaną pozycję w pracy, która najczęściej związana była z nadzorem procesu produkcyjnego.<sup>397</sup> Ów problem społeczny będzie dalej rozwijany w cyfrowym modelu ekonomicznym, który intencyjnie blokuje użytkownikom dostęp do wiedzy na temat działania oprogramowania, w którym przechowywane są strategie zarządzania ekonomicznego i społecznego. Rozważania zawarte w dalszej części rozdziału kontynuować będą problematykę związaną z produkcją, pracą i prawem do własności, ale w kontekście współczesnej kultury cyfrowej i bardziej zaawansowanej formy kapitalizmu – informacyjnego.

---

<sup>396</sup> R. Engelman, dz. cyt.

<sup>397</sup> Tamże.

## **Rozdział 5**

# **Ekonomie oprogramowania**

*Ci, którzy są w stanie kontrolować obieg produktu pracy innych ludzi, mogą narzucać prawa i instytucje społeczne zgodnie ze swoimi interesami. Ci, którzy nie są w stanie zachować kontroli nad produktem własnej pracy, nie są w stanie się temu przeciwstawić.*<sup>398</sup>

Dmitri Kleiner

Po wprowadzeniu historycznym, które przybliżało kontekst rewolucji przemysłowych, chciałabym w niniejszym rozdziale przejść do rozważań osadzonych we współczesności, aby pokazać, że wypracowane na początku XVIII wieku strategie doskonalenia procesu produkcyjnego i wiążącej się z nim eksploatacji pracowników, sprowadzania ich do statusu narzędzia, części procesu technologicznego, rozwija się dalej we współczesnej kulturze cyfrowej, a dzięki oprogramowaniu jako jej systemowi operacyjnemu, zyskuje nowe, udoskonalone formy wyrazu. Niektórzy badacze są zdania, że można wyznaczyć jeszcze trzecią rewolucję przemysłową, którą jest okres od połowy XIX wieku aż do teraźniejszości i jest to rewolucja oparta na komputeryzacji.<sup>399</sup> Pojawiają się już także hipotezy spekulujące na temat nadchodzącej czwartej rewolucji przemysłowej, która opierałaby się na technologii sztucznej inteligencji, inżynierii genetycznej i zaawansowanej robotyzacji.<sup>400</sup> Zwolennicy analizowania kolejnych rewolucji przemysłowych przedstawiają je jako złożony, ciągły, wieloetapowy proces związany z postępem technologicznym, który nigdy się nie skończył, ale raczej przechodzi w kolejne fazy rozwoju.

W odniesieniu do przywoływanych we wprowadzeniu do tego rozdziału zjawisk związanych z rozwojem kapitalizmu przemysłowego, chciałabym teraz odwołać się do współczesnej kultury cyfrowej i tego, w jaki sposób oprogramowanie dostarcza narzędzi do zarządzania pracą na szeroką skalę. Zasadniczą zmianą ekonomiczną, którą wprowadziła rewolucja przemysłowa jest proces, który Marks nazwał alienacją robotnika od efektu jego pracy. W niniejszym rozdziale będę analizować jak współcześnie separuje się użytkowników od efektów ich aktywności sieciowej, która często przybiera formę pracy –

---

<sup>398</sup> D. Kleiner, *The Telekommunist Manifesto*, Institute of Network Cultures, Amsterdam 2010, s. 8. Tłumaczenie A.P.

<sup>399</sup> Zob. C. Anderson, dz. cyt., s. 40-41; J. Rifkin, *The Third Industrial Revolution. How lateral power is transforming energy, the economy and the world*, Palgrave Macmillan, Nowy Jork 2011.

<sup>400</sup> Koncepcja czwartej rewolucji przemysłowej rozpowszechniona została przez Klausa Schwaba, niemieckiego inżyniera i ekonomistę. Zob. K. Schwab, *The Fourth Industrial Revolution*, Portfolio Penguin, Londyn 2017.

zarówno świadomej jak i nieświadomej. Celem tego rozdziału jest również prześledzenie tego, jakie formy może przybierać bunt społeczny wobec panującego modelu ekonomicznego.

### 5.1 Współczesny użytkownik jako robotnik cyfrowy

W drugiej połowie XX wieku podział klasowy i wynikające z niego różnice w warunkach i stylach życia zaczął zanikać, czy też przestał być oczywisty przez ogólne polepszenie się standardów życia członków społeczeństwa, na które wpłynęły przede wszystkim rozwój gospodarki i szkolnictwa.<sup>401</sup> Ulrich Beck, niemiecki socjolog, w klasycznej już dziś książce *Spoleczeństwo ryzyka – w drodze do innej nowoczesności* z 1986 roku, omawiał między innymi zjawisko indywidualizacji jako tendencji społeczeństw przechodzących od etapu przemysłowego do poprzemysłowego (nowoczesność). Odnosiło się ono do procesu, w ramach którego człowiek nie identyfikował się już z konkretną klasą, warstwą społeczną, ale na własną rękę starał się formować swój status ekonomiczno-społeczny poprzez określanie swojej pozycji na rynku pracy<sup>402</sup>. Beck pisał, że indywidualizacja i wiążący się z nią odwrót od kultury klasowej oraz od przywilejów wynikających z pochodzenia były możliwe dzięki takim czynnikom jak rosnąca powszechność i dostęp do zdobywania wykształcenia wyższego oraz wzrost mobilności, co z kolei przyczyniło się do budowania mniej silnych, dających się zerwać i budować na nowo relacji z miejscem zamieszkania czy pracy.<sup>403</sup> Opisywane zmiany dotyczyły zarówno kobiet, jak i mężczyzn. Globalizacja kapitału postępująca w XX wieku, wpłynęła na ukształtowanie się odmiennego modelu robotnika od tego, który został utrwalony na początku epoki przemysłowej. Samodzielnie definiuje on swoją relację pracowniczą. W tym kontekście Edwin Bendyk podkreślał, że zaproponowany przez Marska podział na klasy (proletariat i burżuazję) przestaje być aktualny na rzecz atomizacji społeczeństwa, która przekłada się na ukształtowanie się „indywidualizowanego społeczeństwa pracowników”.<sup>404</sup>

Dodatkowa różnica między epoką przemysłową a współczesną wykracza poza opisywany wyżej model robotnika i dotyczy kwestii ulokowania samego kapitału w skali

---

<sup>401</sup> U. Beck, *Spoleczeństwo Ryzyka. W drodze do innej nowoczesności*, tłum. S. Cieśla, Wydawnictwo Naukowe Scholar, Warszawa 2002, s. 117.

<sup>402</sup> Tamże, s. 118.

<sup>403</sup> Tamże, s. 119-123.

<sup>404</sup> E. Bendyk, *Antymatrix. Człowiek w labiryncie sieci*, Wydawnictwo W.A.B., Warszawa 2004, s. 22.

globalnej. Obecnie „niewielki odsetek ludności świata nadzoruje środki produkcji, za pomocą których wytwarza się znaczną część jego bogactwa.”<sup>405</sup> Z raportu opublikowanego na początku 2022 roku przez brytyjską fundację Oxfam, zajmującą się problemem nierówności społecznej na świecie, wynika, że dziesięciu najbogatszych ludzi zyskało dzięki kryzysowi wywołanemu przez pandemię COVID 19 i z bilionerów stali się trylionerami, podczas gdy przychody i majątek 99% ludzkości znacznie się zmniejszyły.<sup>406</sup> Oznacza to, że współcześnie tak samo jak na początku rewolucji przemysłowej, pracownicy nie posiadają zdolności ekonomicznej do gromadzenia majątku. W przypadku najbogatszych obecnie ludzi na świecie wzbogacenie się w tak znacznym stopniu było możliwe dzięki temu, że stworzyli oni firmy, które zyskiwały na aktywności ich użytkowników. Korzystanie z przeglądarki Google, komunikowanie się przez Facebooka, płacenie za pomocą PayPala czy zamawianie zakupów na Amazonie są powszechnymi czynnościami wykonywanymi codziennie przez miliony osób na świecie, a które przekładają się na pomnażanie się kapitału najbogatszych. W przeciwieństwie do czasu początku rewolucji przemysłowej, kiedy robotnik był świadomy tego, że jego praca, przekłada się na wytworzenie kapitału dla właściciela fabryki, co powodowało, że bunt mógł zostać ukierunkowany, obecnie kwestia ta jest mniej wyrazista. Bowiem z perspektywy nowego systemu ekonomicznego, każdy niemal użytkownik internetu jest bezwiednym pracownikiem, który mimo braku oficjalnego zatrudnienia i tak przyczynia się do zysku właścicieli platform cyfrowych, których celem jest doprowadzenie do kapitalizacji jak największego zakresu codziennych czynności.

W kontekście buntu luddystów w początkowym okresie rewolucji przemysłowej amerykański socjolog, Craig Calhoun argumentował za tym, że to właśnie identyfikacja z klasą robotniczą, pozwoliła przeciwnikom postępu technologicznego organizować się jako społeczność i to właśnie dzięki niej fenomen luddyzmu był w ogóle możliwy.<sup>407</sup> Współcześnie, w „zindywidualizowanym społeczeństwie pracowników”, możliwości budowania wspólnotowości i identyfikacji z określoną grupą społeczną wydają się być bardziej ograniczone. Sieciowa struktura internetu charakteryzuje się równoległymi, lecz sprzecznymi tendencjami, z jednej strony do tego, by sprzyjać szybkiej wymianie informacji

---

<sup>405</sup> C. Harman, *Kapitalizm zombie. Globalny kryzys i aktualność myśli Marksa*, tłum. H. Jankowska, Warszawskie Wydawnictwo Literackie Muza, Warszawa, 2011, s. 45.

<sup>406</sup> Zob. *Inequality Kills: The unparalleled action needed to combat unprecedented inequality in the wake of COVID-19*, Oxfam International, Oxford 2022.

<sup>407</sup> C. Calhoun, *The Question of Class Struggle. Social Foundations of Popular Radicalism during the Industrial Revolution*, The University of Chicago Press, Chicago 1952, s. 60.

i budowaniu więzi, z drugiej jednak do atomizacji, indywidualizacji i zamykania użytkowników w określonych bańkach informacyjnych.<sup>408</sup>

Użytkownik sieci funkcjonuje w skomercjalizowanej przestrzeni, w której podejmuje on samodzielne decyzje i wybory, jednak jego aktywność może przekładać się na zysk lub korzyści firm, platform czy podmiotów dostarczających treści w internecie. Użytkownik, najczęściej w sposób bezwiedny, staje się cyfrowym pracownikiem. W większości sytuacji czerpania korzyści z aktywności użytkowników fakt ten pozostaje ukryty i zachodzi bez świadomości wykorzystywanego. Przypadki te ukrywają się pod osłoną prozaicznych, codziennych czynności. Doskonałym przykładem jest tu test reCAPTCHA, którego zadaniem jest odróżnienie człowieka od bota poprzez weryfikację zadania przeznaczonego dla ludzkiej inteligencji – na przykład przepisania zdeformowanego tekstu lub odróżnienia konkretnych obrazów ze zbioru. Szacuje się, że dziennie test reCAPTCHA jest wykonywany około sto milionów razy.<sup>409</sup> Jest to ogromna próba rezultatów transkrypcji tekstu lub interpretacji obrazu, która okazała się być wykorzystywana przez Google w celu trenowania samouczących się algorytmów.<sup>410</sup> W rezultacie każde wykonanie testu reCAPTCHA, które zazwyczaj jest obowiązkowe w celu wykonania pewnych czynności, takich jak np. odzyskanie hasła, albo opublikowanie postu na forum, przekładało się na bezwiedne trenowanie algorytmów firmy Google.

O ile kapitalizm przemysłowy, opierał się na wynalezieniu i udoskonalaniu technik masowej produkcji, o tyle we współczesnym kontekście, panującą obecnie fazę ekonomiczną określa się mianem kapitalizmu informacyjnego. Socjolog, Manuel Castells na początku XXI wieku opisywał przemianę kapitalistycznego modelu ekonomicznego w następujący sposób:

Globalny system gospodarczy przechodzi wielką transformację – od kapitalizmu przemysłowego opartego na industrializmie, do kapitalizmu informacyjnego, którego zasadą jest informacjonizm. Industrializm jest zorientowany na wzrost gospodarczy, tj. na

---

<sup>408</sup> Termin „bańka informacyjna” wprowadziłam w poprzednim rozdziale, w podrozdziale 4.6 *Podsumowanie*.

<sup>409</sup> J. Havel, *reCAPTCHA: The Genius Who's Tricking the World Into Doing His Work*, 3.12.2023, online: <https://thehustle.co/the-genius-whos-tricking-the-world-into-doing-his-work-recaptcha>, dostęp: 06.11.2023.

<sup>410</sup> Tamże; zob. również: A.A. Casilli, *No CAPTCHA: Yet Another Ruse Devised by Google to Extract Free Digital Labor From You*, 05.12.2014, online: <https://www.casilli.fr/2014/12/05/no-captcha-is-google-jargon-for-mechanical-turk-for-free>, dostęp: 06.11.2023.

maksymalizację produkcji; informacjonizm jest zorientowany na rozwój technologiczny, tj. na akumulację wiedzy oraz na wyższy poziom złożoności w przetwarzaniu informacji.<sup>411</sup>

Badacz wskazywał na to, że to właśnie pojawienie się i ewolucja technologii cyfrowych przyczyniły się do transformacji modelu kapitalizmu. Oprogramowanie dostarczyło w tym przypadku dynamicznie rozwijających się narzędzi do zbierania i dzielenia się informacjami. W sieciowym społeczeństwie informacja szybko stała się podstawową wartością ekonomiczną, co badacz potwierdza słowami:

Stare pytanie społeczeństwa przemysłowego – w istocie kamień węgielny klasycznej ekonomii politycznej – a mianowicie: „czym jest wartość?”, nie ma jednoznacznej odpowiedzi w globalnym społeczeństwie sieciowym. Wartość jest tym, co jest przetwarzane w każdej dominującej sieci, w każdym czasie i w każdej przestrzeni, zgodnie z hierarchią zaprogramowaną w sieci przez aktorów w niej działających. Kapitalizm nie zniknął. W rzeczywistości jest bardziej wszechobecny niż kiedykolwiek.<sup>412</sup>

Castells jasno wyraża przekonanie na temat tego, że kapitalizm, jako zjawisko ekonomiczne rozpoczęte wraz z rewolucją przemysłową, cały czas istnieje, choć uległo licznym transformacjom, zależnym od rozwoju technologicznego. Inna socjolożka, Shosana Zuboff, podkreślała to, jak szybko kapitalizm informacyjny rozwijał się poprzez postępujący proces digitalizacji kultury:

W 1986 tylko 1 procent światowych informacji był zdigitalizowany, w roku 2000 zaś było to 25 procent. Do 2013 postęp digitalizacji i datafikacji danych (zastosowanie oprogramowania, które pozwala komputerom i algorytmom przetwarzać i analizować surowe dane) w połączeniu z nowymi i tańszymi technologiami przechowywania umożliwił przełożenie 98 procent światowych informacji na format cyfrowy.<sup>413</sup>

Odwołując się do opisu badaczki można podsumować, że w cyfrowej rzeczywistości, w której niemal każda informacja ma zdigitalizowaną postać i staje się podstawową wartością ekonomiczną, a oprogramowanie jest narzędziem, które umożliwia zbieranie, przechowywanie i przede wszystkim analizę danych w ogromnej skali.

---

<sup>411</sup> E. Bendyk, *Wstęp. Fałszywa alternatywa*, w: M. Castells, P. Himanem, *Spółeczeństwo informacyjne i państwo dobrobytu. Model fiński*, tłum. M. Sutkowski, M. Penkala, Wydawnictwo Krytyki Politycznej, Warszawa 2009, s. 7.

<sup>412</sup> M. Castells, *Communication Power*, Oxford University Press, Oxford 2009, s. 29.

<sup>413</sup> S. Zuboff, *Wiek kapitalizmu inwigilacji. Walka o przyszłość ludzkości na nowej granicy władzy*, tłum. A. Unterscheutz, Zysk i S-ka, Poznań 2020, s. 262.

Przejście od kapitalizmu przemysłowego do informacyjnego wiązało się z przemianą sposobów pracy i wytwarzania jej wartości. Współczesne społeczeństwo zaczęło funkcjonować w sposób zbliżony do zbiorowego intelektu a wymiana informacji stała się jednym z podstawowych procesów społecznych. Włoski filozof, Maurizio Lazzarato, wprowadził w 1996 roku pojęcie pracy niematerialnej w celu kontynuowania marksistowskiej myśli w kontekście współczesnym. Termin ten został przedstawiony w eseju *Immaterial Labor*. Zaproponowane przez filozofa pojęcie miało określać takiego rodzaju pracę, która wytwarza wartości informacyjne lub kulturowe.<sup>414</sup> Praca niematerialna miała więc odnosić się do dwóch osobnych aspektów pracy. W ramach pierwszego z nich wytwarzany towar miał charakter informacyjny, co wiązało się z przemianami procesu produkcyjnego i sposobu pracy, w związku z którymi bezpośrednia praca fizyczna zastępowana była przez aktywności związane z obsługą komputera, cybernetyką, zarządzaniem systemem i komunikacją. Drugim aspektem, do którego odnosi się pojęcie zaproponowane przez Lazzarato jest aktywność, która wcześniej nie była pojmowana w kategoriach pracy, a konkretnie wytwarzanie wartości kulturowych, a więc: utrwalanie nowych standardów artystycznych, modowych, gustów norm i trendów oraz kształtowanie opinii publicznej.<sup>415</sup>

Koncepcja pracy niematerialnej Lazzarato była później rozwijana chociażby przez Tizianę Terranovą, włoską aktywistkę i badaczkę, która już na początku XXI wieku pokazywała złożoność kategorii pracy i wytwarzania wartości w kontekście współczesnej ekonomii kapitalistycznej, wykorzystującej platformy cyfrowe. Terranova zwracała uwagę na to, że kapitalistyczne modele sieci rozwijają się w oparciu o monetyzowanie wartości związanych z wiedzą, kulturą i afektami.<sup>416</sup> Badaczka podkreślała również, że kapitalizacja cyfrowych sieci wymiany informacji i wartości kulturowych otwiera przestrzeń dla organizacji nieświadomej, niepłatnej pracy i, co za tym idzie, wielu niesprawiedliwości i nadużyć.<sup>417</sup>

Największe platformy sieciowe były do tej pory całkowicie darmowe w użyciu, jednak czerpały zyski z aktywności swoich użytkowników. W ich obszarze wszystkie społeczne interakcje stają się nieodpłatną pracą, która przekłada się na zysk dla właścicieli

---

<sup>414</sup> M. Lazzarato, *Immaterial Labor*, w: *Radical thought in Italy. A Potential Politics*, red. P. Virno, M. Hardt, University of Minnesota Press, Minneapolis, Londyn 2006, s. 133.

<sup>415</sup> Zob. Tamże.

<sup>416</sup> T. Terranova, *Free Labor: Producing Culture for the Digital Economy*, „Social Text” 2000, nr 18 (2), s. 38.

<sup>417</sup> T. Terranova, *Network Culture: Cultural Politics for the Information Age*. Pluto Press, Nowy Jork 2004, s. 73-94.

tego rodzaju platform. Trudno w tym kontekście wprowadzić rozróżnienie pomiędzy aktywnością sieciową a pracą.<sup>418</sup> Problematyka kapitalizacji aktywności użytkowników i niematerialnej pracy podejmowana jest przez dzieła artystyczne z dziedziny sztuki mediów, co doskonale widać na przykładzie dwóch prac z dziedziny sztuki mediów: *Facebook Algorithmic Factory* oraz *The Sheep Market*. Te dwa projekty artystyczne demaskują strategie obciążania użytkowników pracą, wykorzystywane przez dwie ogromne korporacje: Facebook i Amazon. Pierwsza instalacja skupiła się na temacie pracy nieświadomej, z kolei druga na świadomym podejmowaniu drobnych zleceń w ramach platformy Amazon Mechanical Turk. Obie wzięły sobie za cel ujawnianie nie dających się postrzegać w codziennym doświadczeniu zależności i relacji, kryjących się za interfejsem aplikacji. Na przykładzie obu instalacji widać, że tym, co chowa się po drugiej stronie usług internetowych, jest skala zjawisk, z którymi wchodzi w interakcję człowiek. Zindywidualizowanemu użytkownikowi, zamkniętemu w jednostkowej perspektywie umyka rozpiętość procesów ekonomiczno-społecznych, z którymi ma do czynienia.

## **5.2 Facebook Algorithmic Factory**

Poprzez instalację *Facebook Algorithmic Factory*,<sup>419</sup> Katarzyna Szymielewicz, Vladan Joler oraz Paweł Janicki, podjęli próbę przedstawienia tego, w jaki sposób działają algorytmy popularnego medium społecznościowego, jak zbierają dane o jego użytkownikach, określają ich profil, a następnie monetyzują ich aktywność.<sup>420</sup> Ten konkretny projekt został już przywołany w drugim rozdziale, jako odniesienie uzupełniające analizę kwestii politycznej manipulacji za pomocą mediów społecznościowych. W ramach niniejszej części dysertacji przykład ten ma otworzyć rozważania na temat ekonomiczno-społecznych aspektów aktywności sieciowej.

---

<sup>418</sup> N. Srnicek, *Platform Capitalism*, Polity Press, Malden 2017, EPUB, rozdz 2.

<sup>419</sup> Do tej instalacji odwoływałam się już w poprzednim rozdziale, gdzie przywoływałam okoliczności jej powstania, autorów datę i miejsce jej pierwszego wystawienia.

<sup>420</sup> Należy tu również podkreślić, że Facebook to nie tylko portal społecznościowy. Należą do niego także inne firmy, wśród których wymienić można najbardziej rozpoznawalne: Instagram, WhatsApp, Parse czy Oculus. W celu podkreślenia tej różnorodności i przeciwdziałania identyfikacji z pojedynczą platformą społecznościową, w 2021 CEO firmy, Mark Zuckerberg ogłosił początek procesu zmiany marki z Facebook na Meta. Pod nową nazwą marka ma mieścić się sieć powiązanych ze sobą produktów i usług określanych mianem metaverse. W czasie, gdy powstała instalacja jeszcze nie doszło do zmiany nazwy marki, ale już wtedy firma była siecią usług i produktów, wykupującą kolejne przedsiębiorstwa i włączającą do swojego cyfrowego imperium, stając się w ten sposób olbrzymim potentatem danych.

Zasadniczą częścią tego badawczo-artystycznego przedsięwzięcia były długotrwałe badania sposobu działania algorytmów Facebooka. Praca badawcza uwzględniała także analizę dostępnej dokumentacji, regulaminu platformy oraz oferty biznesowej dla partnerów zewnętrznych, którym Facebook sprzedaje dane użytkowników w ramach programów partnerskich. Efektem tych analiz była instalacja, która reprezentowała wyniki badań za pomocą różnych narracji: wizualnej, w postaci rozbudowanych, interaktywnych infografik; tekstowej, w formie esejów oraz głosowej, poprzez nagrania relacji trzech użytkowników platformy społecznościowej (Julii, Rafała i Magdy), którzy opowiadali o nadużyciach ze strony platformy, których doświadczyli.<sup>421</sup> Pokazali oni, w jaki sposób algorytmy interpretują dane zawarte w profilach użytkowników oraz jak określają relacje między innymi profilami, tworząc sieć powiązań. Instalacja udowodniła, że jako użytkownik można nie podać żadnych informacji na swój temat, a mimo to poprzez analizę aktywności najbliższych znajomych w tak skonstruowanej sieci, algorytmy są w stanie dostarczyć informacji na temat płci, stylu konsumpcji, poglądów politycznych czy zarobków. Jeśli użytkownik pozwoli aplikacji na dostęp do plików zapisanych na telefonie, algorytmy na podstawie analizy metadanych wykonanych zdjęć są w stanie dostarczyć informacji chociażby na temat lokalizacji w określonym czasie i połączyć te dane z zebranymi od innych użytkowników. W ten sposób, czego można było się dowiedzieć z jednej z narracji zawartej w instalacji, może dojść do sytuacji, kiedy fotografia wykonana telefonem podczas demonstracji politycznej pozwoliła określić czyjś profil polityczny, co przełożyło się w konsekwencji na utratę pracy osoby opowiadającej. Inna narracja, prezentowana w ramach projektu przedstawiała to, jak dwóm osobom kupującym bilet lotniczy zaprezentowana została różna cena na ten sam lot, kupowany w tym samym momencie, co dowodziło temu, że platforma sprzedająca bilety dostosowywała cenę do profili behawioralnych użytkowników.

Wymiar cyfrowej inwigilacji, odbywającej się bez świadomości użytkowników stanowi szczególnie problematyczne zagadnienie. Tak wnikliwa kontrola podporządkowana jest celom ekonomicznym i ma przysłużyć się temu, by zebrać jak najdokładniejsze informacje dotyczące użytkowników, po to, aby umożliwić podmiotom zewnętrznym, dostosowanie treści reklamowych do potencjalnych klientów. Te strategie są podstawowym narzędziem w ramach działania systemu ekonomicznego, który Shoshana Zuboff określa

---

<sup>421</sup> Instalacja była prezentowana na Biennale Sztuki Mediów WRO we Wrocławiu w 2017 roku zob. *Facebook Algorithmic Factory*. Katarzyna Szymielewicz, Vladan Joler, Paweł Janicki, 18.05.2017, online: <http://wro2017.wrocenter.pl/facebook-algorithmic-factory/>, dostęp: 06.11.2023.

mianem kapitalizmu inwigilacji.<sup>422</sup> Badaczka podkreślała, że charakterystyczne dla tego rodzaju systemu są metody akumulacji kapitału, oparte na mechanizmie eksploracji i predykcji.<sup>423</sup> W tym kontekście eksploracja polega na bezpłatnym pozyskiwaniu danych użytkowników, a następnie predykcja odnosi się do procesu przetwarzania danych behawioralnych do postaci „produktów predykcyjnych”<sup>424</sup>, które z kolei stanowią właściwy towar sprzedaży dla potencjalnych reklamodawców lub innych platform. Jak widać, w tym systemie nie same dane są przedmiotem transakcji, ale modele predykcyjne określające użytkowników, które tworzy się na ich podstawie. Zasadniczą cechą kapitalizmu inwigilacji jest to, że to właśnie metody nadzoru i kontroli generują zyski dla wykorzystujących je gigantów cyfrowych. Inwigilacja stała się podstawowym mechanizmem wytwarzania kapitału.

Facebook stał się bankiem ludzkich danych – w tym opisie określenie „bank” wydaje się być bardzo trafne, ponieważ tak samo jak kapitał w banku, który z jednej strony jest przechowywany i chroniony, ale z drugiej strony instytucja finansowa ma prawo obracać nim dla własnych korzyści, podobnie rzecz ma się z danymi osób prywatnych w internecie. Badacz cyfrowej ekonomii, Nick Srnicek wskazywał na to, że platformy cyfrowe wypracowały na przełomie XX i XXI wieku, a więc wraz ze zmianą modelu sieci z Web 1.0 na Web 2.0, strategie ekonomiczne zapewniające im utrzymanie się i przynoszenie przychodu, przy jednoczesnym pozostaniu darmowymi dla użytkowników.<sup>425</sup> Najbardziej powszechny model ekonomiczny opiera się na umieszczaniu reklam na platformach cyfrowych, a więc sprzedawaniu reklamodawcom możliwości docierania do określonych grup odbiorców, które da się określić za pomocą analizy danych, ciasteczek i aktywności sieciowej.<sup>426</sup> Srnicek podkreślał w 2017, że Facebook i Google są całkowicie zależne od tego modelu.<sup>427</sup>

Użytkownicy platformy społecznościowej logują się na nią w celach komunikacyjnych, rozrywkowych, informacyjnych czy relaksacyjnych, co oznacza, że robią to często i pozwalają tej czynności stać się częścią codziennego życia, a nawet jednym z nawyków. Interfejs aplikacji i sposób jej działania zaprojektowane zostały tak, aby oddziaływać na użytkownika, dostarczać mu przyjemności, nasilać jego zaangażowanie, po

---

<sup>422</sup> S. Zuboff, dz. cyt.

<sup>423</sup> Tamże, s. 294.

<sup>424</sup> Tamże, s.19.

<sup>425</sup> Jak wspominałam już wcześniej model ten właśnie ulega przekształceniom przez wzgląd na zaostrenie się praw ochrony danych osobowych. Z tego powodu większość platform społecznościowych obecnie wprowadza możliwość płatnej subskrypcji. Omówię jeszcze to zjawisko w zakończeniu pracy.

<sup>426</sup> N. Srnicek, dz. cyt., rozdział 2.

<sup>427</sup> Tamże.

to, aby platforma działała na najwyższych obrotach i przynosiła korzyści ekonomiczne. Na ekranie logowania jeszcze do 2019 roku widniało hasło „To jest i zawsze będzie darmowe”, które zapewniało rejestrujących się, że nie ponoszą oni żadnych kosztów związanych z korzystaniem z platformy. Zarząd Facebooka, bez uzasadniania swoich motywów, podjął decyzję o tym, żeby zmienić to motto na „To szybkie i proste”, istnieją spekulacje, że przyczyną zmiany były liczne skandale, związane ze wzbogacaniem się dzięki aktywności użytkowników.<sup>428</sup>

Przekaz instalacji *Facebook Algorithmic Factory* oparty został na bardzo wyrazistej metaforze – platformy społecznościowej jako fabryki. Przedstawia ona użytkowników Facebooka jako robotników, którzy bezwiednie, poprzez swoją codzienną aktywność dostarczają danych, które są podstawowym materiałem budulcowym do tworzenia cyfrowych profili, będących odzwierciedleniem osobowości internautów, które z kolei stają się głównym towarem wymiany na rynku ekonomii uwagi. W jednym z esejów uzupełniających treść projektu artystycznego, Joler zaprezentował wyniki badań nad strategią ekonomiczną platformy i wzbogacił je stworzonymi przez siebie infografikami ułatwiającymi wyobrażenie sobie sposobu działania algorytmów.<sup>429</sup> Pomiedzy grafikami autor tekstu umieścił również obrazy z historii i popkultury, takie jak na przykład scena z filmu *Matrix*, w której latające maszyny uprawiają farmę z zanurzonymi we śnie ludźmi. Jedną z tych kontekstowych fotografii przedstawiała halę pełną robotników pracujących na swoich stanowiskach w fabryce. Tak zbudowane odniesienie pomiędzy przemysłową nowoczesnością a cyfrową współczesnością zdaje się być spełnieniem przywoływanego przeze mnie we wprowadzającej części kontekstowej wyobrażenia Marksa na temat fabryki-automatu, która jest bytem hybrydycznym: po części technologicznym, po części organicznym. Autor instalacji opisując procesy technologiczne korzysta również z języka opisu odpowiadającemu mówieniu o procesach biologicznych. Joler pisze bowiem o pracy algorytmów na portalu społecznościowym, że są one procesem ekstrakcji danych z biomasy.<sup>430</sup> Określenie biomasa, użyte przez Jolera w kontekście oprogramowania oznaczać ma bezkształtny, surowy materiał cyfrowy, nie dający się interpretować zbiór

---

<sup>428</sup> Facebook usuwa hasło o tym, że zawsze będzie darmowy. Nadchodzi płatna wersja serwisu?, 08.09.2019, online: <https://www.wirtualnemedi.pl/artukul/facebook-usuwa-haslo-o-tym-ze-zawsze-bedzie-darmowy-nadchodzi-platna-wersja-serwisu>, dostęp: 06.11.2023.

<sup>429</sup> V. Joler, A. Petrovski, *Immaterial Labour and Data Harvesting. Facebook Algorithmic Factory*, 21.08.2016, online: <https://labs.rs/en/facebook-algorithmic-factory-immaterial-labour-and-data-harvesting>, dostęp: 06.11.2023.

<sup>430</sup> V. Joler, A. Petrovski, *Immaterial Labour...*, dz. cyt.

zapisów aktywności użytkowników, z którego dopiero po algorytmicznym przetworzeniu da się uzyskać wartościowe dane.

Jaką rolę odgrywa oprogramowanie w omawianych tu procesach ekonomiczno-społecznych? Przede wszystkim jest ono zapisem działania programu, a co za tym idzie, zawiera strategie inwigilacji i kontroli użytkowników, a przy tym pozostaje ono ukryte i niedostępne, schowane wewnątrz algorytmicznej czarnej skrzynki. Autorzy instalacji nie mieli dostępu do samego kodu, mogli jedynie wnioskować na temat jego działania na podstawie dostępnej dokumentacji, relacji użytkowników i przypadkowych wycieków danych. W tej sytuacji oprogramowanie nie mogło zostać udostępnione do powszechnego wglądu, ponieważ znajdowało się pod ochroną praw własności jako własność intelektualna firmy Facebook. Projekt badawczy i instalacja, która powstała w jego rezultacie są więc tu próbą wyjaśnienia i graficznego zobrazowania tego, jak działa oprogramowanie, do którego ani autorzy *Facebook Algorithmic Factory*, ani użytkownicy platformy nie mają prawa ani możliwości uzyskania dostępu. Joler, jako artysta i teoretyk mediów, w swojej działalności poświęcił się szukaniu sposobów obrazowania działania oprogramowania – w formie graficznej, ale także poprzez pojęcia fizyczne czy za pomocą metafor filozoficznych. W ramach wyjaśniania tego, jak rozumieć można ślad cyfrowy w postaci danych, który zostawia po sobie aktywność użytkowników w sieci, Joler odwoływał się do platońskiej koncepcji jaskini.<sup>431</sup> Zgodnie ze starożytną myślą, uwięzieni w jaskini mogą jedynie oglądać cienie prawdziwych obiektów i zjawisk, bez możliwości odwrócenia się i poznania ich faktycznego źródła. Artysta-badacz powoływał się na tę teorię, aby powiedzieć, że wyekstrahowany z biomasy zestaw danych, mający być reprezentacją profilu psychometrycznego danego użytkownika, w podobny sposób jest jego cieniem, związaną ze swoim oryginałem reprezentacją, ale w żadnym wymiarze nie jest dokładnym jego przedstawieniem. Myśl ta została zobrazowana przez wyrazistą animację, w której, zgarbiony i pochylony nad jasnym interfejsem człowiek jest oświetlany przez emitowane z urządzenia światło, przez co na ścianę za nim rzucany jest cień przedstawiający go w pozycji wyprostowanej. Metaforę Platona można też wykorzystać jako odniesienie do analizowania oprogramowania bez możliwości wglądu w kod źródłowy. Proces ten polega na konstruowaniu wniosków na podstawie skutków działania badanego obiektu bez możliwości bezpośredniego poznania przedmiotu analizy.

---

<sup>431</sup> Artysta przedstawił tę koncepcję i związaną z nią serię grafik i animacji podczas wykładu zorganizowanego przez Instytut Sztuki Współczesnej Axioma w Lublanie, w Słowenii, zob. Vlado Joler, *Dissecting New Extractivism*, 23.03.2022, online: [https://www.youtube.com/watch?v=NJ2itiA6uFE&t=1517s&ab\\_channel=Aksioma](https://www.youtube.com/watch?v=NJ2itiA6uFE&t=1517s&ab_channel=Aksioma), dostęp: 06.11.2023.

Podjmowanie tego rodzaju prób badania ukrytych zależności oprogramowania i wyjaśniania działania czarnoskrzynkowych technologii jest niezmiernie istotną misją społeczną. Dzięki tego rodzaju dociekaniami udało się ustalić i dopuścić do powszechnego obiegu informacji, że to samo prawo, które chroni oprogramowanie jako własność intelektualną ogromnej platformy, nie zapewnia ochrony jej użytkownikom przed tym, żeby ich dane poddawane były monetyzacji i stawały się przedmiotem transakcji na rynku ekonomii uwagi.<sup>432</sup> Oprogramowanie pośredniczy w wielu rozległych procesach ekonomiczno-społecznych, których mechanizmy tak długo pozostają ukryte, jak długo nie zdobędzie się wiedzy o działaniu software'u. Podstawowym postulatem dziedziny *software studies* jest wprowadzanie dyskursu dotyczącego oprogramowania na gruncie naukowym, po to, aby rozbudowywać wiedzę na jego temat, kształtować transdyscyplinarny język mówienia o problemach z nim związanych. Przywoływany tu przykład projektu artystycznego *Facebook Algorithmic Factory* jest doskonałym przykładem budowania dobrych, pożytecznych społecznie, metafor oprogramowania. Dzieło to wpisuje się w misję założonej przez Jolera fundacji i centrum badawczego SHARE Lab, którego głównym założeniem jest dzielenie się wiedzą na temat działania technologicznych czarnych skrzynek. Instalacja jest także doskonałą odpowiedzią na problem braku dobrych metafor oprogramowania, który opisywałam w poprzednim rozdziale. Shoshana Zuboff podkreślała, że sprawowanie władzy poprzez kontrolę w ramach kapitalizmu inwigilacji jest możliwe przede wszystkim dzięki asymetrii wiedzy. Brak świadomości użytkowników dotyczącej tego, jakie dane są zbierane w konsekwencji ich aktywności oraz, co dzieje się z nimi potem, umacnia skuteczność kapitalistycznych strategii nadzoru.<sup>433</sup> Tym bardziej istotne jest, aby powstawały tego rodzaju przedsięwzięcia artystyczno-badawcze, które przyczyniają się do ujawniania owej asymetrii wiedzy i edukowania użytkowników sieci.

### 5.3 *The Sheep Market*

Problematykę zależności pomiędzy produkcją, własnością i kapitałem w sferze cyfrowej oraz wynikających z relacji między nimi stosunków pracy podjął za pomocą projektu artystycznego również inny programista, Aaron Koblin, autor *The Sheep Market*. Jego praca

---

<sup>432</sup> Pojęcie ekonomii uwagi wprowadziłam i wyjaśniłam już w czwartym rozdziale pracy, w podrozdziale 4.3 *Cambridge Analytica: oprogramowanie jako inżynieria społeczna*.

<sup>433</sup> S. Zuboff, dz. cyt., s. 259-261.

była krytycznym komentarzem w kontekście działania platformy Amazon Mechanical Turk (funkcjonującej także pod skróconą nazwą MTurk). W odróżnieniu do przywoływanego wcześniej *Facebook Algorithmic Factory*, projekt artystyczny Koblina skupia się na kwestii świadomego podejmowania się pracy przez użytkowników internetu i analizuje on problem wykorzystywania pracowników cyfrowych poprzez rozwiązania crowdsourcingowe.

Amazon Mechanical Turk (MTurk) jest to jedna z usług internetowych udostępnianych przez przedsiębiorstwo Amazon. Jest to platforma o charakterze crowdsourcingowym służąca do zlecania i wykonywania prostych zadań, których obecnie nie można zrealizować automatycznie, ponieważ jest to utrudnione lub nie przynosi zadowalających efektów.<sup>434</sup> Swoją nazwą nawiązuje bezpośrednio do Mechanicznego Turka, automatu Kempelena z XVIII wieku. Użytkownicy zarejestrowani na platformie jako zleceniodawcy (*requester*) mogą umieszczać na niej zlecenia – każde z nich należy do tzw. HIT (*Human Intelligence Task*). Mimo daleko posuniętego rozwoju technologii cyfrowych, cały czas istnieją zadania wykraczające poza możliwości sztucznej inteligencji, a przy tym jednocześnie będące łatwe do wykonania dla człowieka. Mogą to też również być aktywności, które pozwalają się zautomatyzować, jednak które człowiek, jak dotąd jest w stanie wykonać o wiele skuteczniej. Należą do nich chociażby: moderowanie treści, weryfikacja tłumaczenia, etykietowanie zbioru obrazów czy też zadania wymagające abstrakcyjnego myślenia. Zleceniodawcy zarejestrowani na platformie MTurk ustalają wynagrodzenie za wykonanie wystawionych przez nich zadań. Kwoty za realizację pojedynczego zlecenia zwykle nie są wysokie (może to być przykładowo 0,04\$), ale pojawiają się również bardziej wymagające, za których wykonanie można otrzymać ponad 30\$.

Mimo że aktywność użytkowników MTurk sprowadza się do realizacji drobnych zleceń i otrzymywania za nie zapłaty, platforma jest promowana przy wykorzystaniu języka, który unika określania tych aktywności jako pracy. W ramach reklamowania serwisu internetowego, wyprodukowana została seria materiałów promocyjnych w postaci kubków z napisem „Po co pracować, jeśli możesz turkować?” („Why work if you can turk?”).<sup>435</sup> Inskrypcja umieszczona na kubkach określa pracę i wykonywanie zleceń na platformie MTurk jako przeciwieństwa lub wykluczające się aktywności. Realizowanie ofert dostępnych w ramach tej platformy crowdsourcingowej było promowane jako hobby, coś co nie jest męczące (jak praca), a przy okazji dostarcza pewnego dochodu. Na tym przykładzie

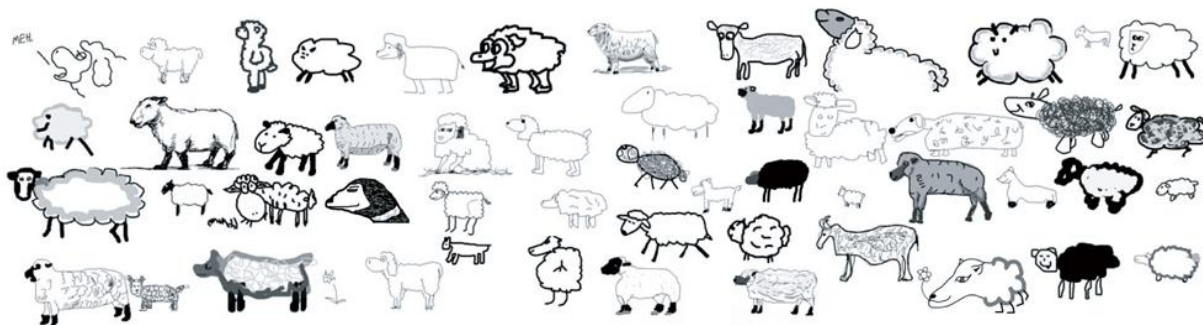
---

<sup>434</sup> Zob. <https://www.mturk.com>, dostęp: 06.11.2023.

<sup>435</sup> G. Cox, A. McLean, dz. cyt., s. 49.

wyraźnie widać strategię manipulacyjną, która realizowana jest za pomocą języka. Poprzez zmianę sposobu opisu aktywności w ramach platformy MTurk, starano się odciążyć działanie użytkowników ze znaczeń związanych z pracą. Tym samym, jest to budowanie nowych skojarzeń i metafor, których celem jest ukrywanie chociażby tego, że tanio wyceniona praca w ramach platformy crowdsourcingowej podporządkowana jest zasadom kapitalistycznej produkcji.

Aaron Koblin, artysta i programista, realizujący się w dziedzinie sztuki mediów i software artu, przyjął za cel ujawnienie i graficzne opracowanie właśnie tych ekonomiczno-społecznych zależności, które ukrywane są pod powierzchnią działania platformy cyfrowej MTurk. Od 2008 roku pracował on nad projektem *The Sheep Market*. Rozpoczął od umieszczenia na MTurk zadania o treści „narysuj owcę, która patrzy w lewo”. Zlecenie wycenił na 0,02\$ za jeden rysunek i szybko zebrał dziesięć tysięcy grafik, stworzonych przez zarejestrowanych użytkowników platformy (tak zwanych „Turkers”). Zgromadzony w ten sposób zbiór danych w postaci grafik stał się podstawą narracji artystycznej, dotyczącej tanio wycenionej pracy cyfrowej. *The Sheep Market* jako projekt artystyczny składał się z kilku elementów: ogromnego kolażu, przedstawiającego wszystkie zebrane grafiki (Il. 14; 15), animacji odtwarzających powstawanie każdej z nich oraz strony internetowej.<sup>436</sup>



Il. 14. *The Sheep Market*. Zbliżenie na fragment kolażu. Źródło: online: <http://www.aaronkoblin.com/work/thesheepmarket>, dostęp: 06.11.2023.

<sup>436</sup> Strona internetowa projektu nie jest już funkcjonalna, przez to, że wtyczka Adobe Flash Player została wycofana: <http://www.thesheepmarket.com>.



Il. 15. *The Sheep Market*. Kolaż składający się z 10 000 prac stworzonych przez użytkowników MTurk.  
Źródło: online: <http://www.aaronkoblin.com/work/thesheepmarket>, dostęp: 06.11.2023.

Artyście za pomocą tak nietypowej techniki wizualizacji danych udało się zobrazować ogromną skalę, na jaką rozciąga się zjawisko taniej siły roboczej, zarządzanej za pomocą platformy crowdsourcingowej. Zwieńczenie pracy w postaci ogromnego kolażu złożonego z owiec narysowanych przez użytkowników MTurk oddawało swoim formatem, ile danych można zebrać za pomocą jednego, tanio wycenionego zlecenia. Kolaż był wielokrotnie wystawiany<sup>437</sup>, zazwyczaj zajmował całą ścianę w prezentowanej przestrzeni, znacznie przewyższając zwiedzających. Miało to na celu odwrócenie perspektywy, z jakiej można postrzegać działanie w ramach platformy. Od strony użytkownika zlecenie wystawione przez Koblina nie wymagało dużego wysiłku ani czasu. Jednak przedstawienie w postaci monumentalnego kolażu, pozwalało zwizualizować nakład pracy, którą udało się zorganizować, poprzez wystawienie prostego polecenia za niewielką cenę. Kolaż, który zajmował całą ścianę galerii i rozciągał się ponad oglądających pozwalał w sposób fizyczny doświadczać i postrzegać skalę omawianego zjawiska. Tego rodzaju percepcja i perspektywa są niemożliwe w codziennym doświadczeniu użytkownika internetu. Biorąc pod uwagę ilość stworzonych w ten sposób grafik owiec, rozwiązanie to, przez swoją wydajność, przypominało zautomatyzowany proces, tak jakby wykonany został przez algorytmy. Jednak przez fakt, że wytwórcami rysunków byli ludzie, każda z owiec była unikalna i wyjątkowa. Indywidualny i jednostkowy charakter pracy, mimo ogromnej ilości wykonawców, artysta wydobyl poprzez kreatywny charakter samego zlecenia. W ten sposób Koblinowi udało się z jednej strony podkreślić to, jak MTurk pozwala organizować na masową skalę pracę użytkowników, artysta nazywa platformę określeniem „narzędzie do

<sup>437</sup> Praca była prezentowana w następujących galeriach i w ramach następujących festiwali: Laboral Centro de Arte, Gijon, Hiszpania, Japan Media Arts Festival, Tokyo, Japonia, Apex Gallery, Nowy Jork, USA ElectroFringe, New Castle, Australia, Media Art Friesland, Leeuwarden Holandia.

dystrybucji pracy”<sup>438</sup>. Z drugiej zaś strony, w geście artystycznym dokonał on upodmiotowienia nawet pozornie mało znaczącej i tanio wycenionej pracy, ponieważ obok kolażu prezentowane były animacje przedstawiające powstawanie każdej z owiec osobno, które artysta zarejestrował podczas ich tworzenia, o czym zleceniobiorcy byli informowani przyjmując zlecenie. Wybór owcy jako tematu pracy również był nieprzypadkowy. Za jego pomocą artysta odnosił się do procesu klonowania, ale także do rolnictwa i hodowli owiec, które przynoszą rolnikowi korzyść w postaci wełny, rosnącej samoistnie, bez specjalnego wysiłku ze strony zwierząt.

W przeciwieństwie do *Facebook Algorithmic Factory*, projekt Aarona Koblina skupia się na przedstawieniu świadomie podejmowanej i płatnej pracy niematerialnej, zapośredniczonej cyfrowo. Nie oznacza to jednak, że jest ona wolna od nadużyć w obszarze etyki pracy. Technologia cyfrowa umożliwiła ukształtowanie się nowych form pracy, opartej na wyzysku taniej siły roboczej. Dzieło Koblina skupione było przede wszystkim na pokazaniu ilościowego wymiaru pracy, którą można zorganizować za pomocą narzędzia do dystrybucji zleceń, jakim jest MTurk. Dzięki *The Sheep Market* mogła powstać narracja, która przeciwstawia się tej formułowanej przez materiały promocyjne firmy Amazon, które w kontekście aktywności użytkowników przesuwają na dalszy plan aspekty związane z pracą i zakrywają je metaforami dotyczącymi hobby i codziennych działań. Oprócz problemów, które zwizualizowała praca Koblina, istnieje jeszcze wiele innych kontrowersyjnych pod względem etycznym kwestii, dotyczących pracy w ramach Amazon Mechanical Turk. Należy do nich chociażby problem fałszywej estymacji, kiedy wykonanie zlecenia zajmuje o wiele więcej czasu niż było to zadeklarowane i zleceniobiorca zamiast poświęcić chwilę, spędza godzinę, lub dłużej na mało płatnym zadaniu. Innym problemem jest fakt, że użytkownicy platformy spędzają nieodpłatnie czas na niezbędnych aktywnościach, takich jak oczekiwanie na pojawienie się zleceń, przeszukiwanie ich, wybieranie odpowiednich. Dochodzi również do wielu nadużyć związanych z odrzucaniem zleceń czy to przez Turkera, spowodowanych niekompletnym, źle przygotowanym, niemożliwym do ukończenia zadaniem, czy to przez zleceniodawcę, przez to, że nie jest zadowolony z wykonanej pracy. W tym drugim przypadku, osoba, która wykonała odrzucone zlecenie, nie dostaje żadnego wynagrodzenia, mimo, że zleceniodawca posiada dostęp do wykonanej przez nią pracy. Przez tak sformułowane zasady działania otwiera się miejsce dla nadużyć, związanych ze strony zleceniodawców, których ofiarami padają

---

<sup>438</sup> Zob. Data Art: *The Sheep Market*, 07.03.2008, online: <https://www.youtube.com/watch?v=3Mmb5aSscck>, dostęp: 06.11.2023.

przyjmujący zlecenia użytkownicy, będący okradani z własnej pracy.<sup>439</sup> Oprócz nadużyć wynikających z zasad działania platformy, odnotowano również wiele przypadków nieetycznych, kontrowersyjnych zleceń, związanych z pornografią nieletnich, transkrypcją danych personalnych czy też wymagających od Turkera podzielenia się traumatycznym wydarzeniem z życia za upokarzająco niską cenę 50 centów.<sup>440</sup>

W ramach platformy Amazon Mechanical Turk, użytkownicy przyjmujący zlecenia do wykonania nie są chronieni przed nadużyciami ze strony zleceniodawców. Prawo pracy, które zwyczajowo jest egzekwowane w przypadkach nadużyć nie ma tu zastosowania, ponieważ zarejestrowani Turkerzy nie są pracownikami, ale niezależnymi zleceniobiorcami. W ten sposób w ramach działania Mturk udało się osiągnąć skrajną formę kapitalistycznej produkcji, ponieważ pozwala ona dysponować siłą roboczą i produktem pracy, ale bez konieczności opieki nad pracownikiem.<sup>441</sup>

## 5.4 Cyfrowe fabryki niematerialnej pracy

Oprogramowanie dostarczyło narzędzi do tego, aby na ogromną skalę zarządzać nisko płatną lub całkowicie darmową, bo nieświadomą, pracą. W cyfrowej rzeczywistości produkcja nie odbywa się już w fizycznych fabrykach, ale w sieci, a robotnikami są użytkownicy, którzy przez swoją aktywność przyczyniają się do wzbogacania się właścicieli poszczególnych platform. Kapitalizacji poddane zostały niemal wszystkie obszary życia codziennego, dlatego użytkownik podejmuje pracę niematerialną prawie cały czas. Ian Bogost w jednym ze swoich esejów postawił czytelnikowi prowokacyjne pytanie o to, czy nie czuje się on zmęczony będąc online? Przecież wykonuje tam dziesiątki prac.<sup>442</sup> Badacz starał się podkreślić przede wszystkim to, że aktywność sieciowa przypomina pracę o charakterze sortowania i analizy danych. Codzienne czynności, takie jak odbieranie maili, zarządzanie pocztą, newsletterami, przeglądanie powiadomień, oprócz tego, że są

---

<sup>439</sup> A. Semuels, *The Internet Is Enabling a New Kind of Poorly Paid Hell*, 23.01.2018, online: <https://www.theatlantic.com/business/archive/2018/01/amazon-mechanical-turk/551192>, dostęp: 06.11.2023.

<sup>440</sup> Kontrowersyjne zlecenia, które miały miejsce w ramach platformy opisane zostały w niniejszym artykule: Dhruv Mehrotra, *Horror Stories From Inside Amazon's Mechanical Turk*, 01.28.2020, online: <https://gizmodo.com/horror-stories-from-inside-amazons-mechanical-turk-1840878041>, dostęp: 06.11.2023.

<sup>441</sup> Więcej na temat nadużyć pracowniczych związanych z platformą MTurk można przeczytać w: T. Scholz, *Overworked and Underpaid. How Workers Are Disrupting the Digital Economy*, Polity Press, Cambridge, UK 2017.

<sup>442</sup> I. Bogost, *Hyperemployment, or the Exhausting Work of the Technology User*, 08.11.2013, online: <https://www.theatlantic.com/technology/archive/2013/11/hyperemployment-or-the-exhausting-work-of-the-technology-user/281149>, dostęp: 06.11.2023.

elementami gry w ramach ruchu na rynku ekonomii uwagi, wymagają również od użytkowników skupienia, uważności i podejmowania decyzji.

Zasadniczym problemem, wynikającym z przywoływanych przeze mnie przykładów pracy niematerialnej jest osłabiona możliwość protestu lub przeciwdziałania temu modelowi ekonomicznemu. Brak poczucia wspólnotowości, będący konsekwencją indywidualizacji społeczeństwa sieciowego, sprawia, że trudniej niż chociażby w epoce przemysłowej, identyfikować się z konkretną klasą lub wspólnotą oraz zorganizować się oddolnie, w celu wywołania jakiejś zmiany. Na przykładzie przedsiębiorstwa Amazon widać jak krytyka systemu jest uniemożliwiana – użytkownicy MTurk, udzielając wywiadów chcą pozostać anonimowi, ponieważ istnieje przekonanie (nie potwierdzone oficjalnie przez przedstawicieli Amazona), że firma banuje konta tych, którzy zniesławiają ją publicznie.<sup>443</sup> Tego rodzaju działanie przeciwdziała wspólnotowości, organizowaniu się, by wspólnie walczyć o swoje prawa. Jest to jeden z wielu przykładów na to, jak współcześnie opisywane przez Marksa zjawisko alienacji zyskuje nowe znaczenie w odniesieniu do niematerialnej pracy zapośredniczonej cyfrowo (*digital alienation*)<sup>444</sup>. Ujawnia się tutaj charakterystyczna dla historycznego modelu kapitalizmu przemysłowego eksploatacja możliwości produkcyjnych pracownika, której przeciwdziałały oddolne ruchy społeczne ostatniego stulecia, a która może ponownie przejawiać się w nowych formach, w ramach dominującego współcześnie kapitalizmu informacyjnego.

Zgodnie z koncepcją „fabryki społecznej” sformułowaną przez włoskiego filozofa, Mario Tronti’ego w 1971 roku i rozwijaną później przez Antonio Negriego, fabryka nie ogranicza się tylko do określonego systemu produkcyjnego, ale jest zjawiskiem społecznym, w ramach którego kapitał stał się nową, rozproszoną formą władzy.<sup>445</sup> Do tej koncepcji odnosili się współcześni krytycy kapitalizmu informacyjnego, wśród których wymienić można chociażby Tizianę Terranovą, która podkreślała to, że strategie organizacji pracy przeniosły się z fabryki na poziom społeczeństwa, co przekłada się na kapitalizację codziennych aktywności zapośredniczonych przez cyfrowe sieci komunikacyjne.<sup>446</sup> Odniesienie do koncepcji fabryki w szerokim kontekście społecznym cały czas jest podejmowane w dyskursie badawczym, co widać chociażby na przykładzie tekstów zebranych w publikacji pod redakcją Trevora Scholza, o tytule *Digital Labor. Internet as*

---

<sup>443</sup> A. Semuels, dz. cyt.

<sup>444</sup> M. Andrejevic, *Estranged Free Labor*, w: *Digital Labor. The Internet as a Playground and Factory*, red. T. Scholz, Routledge, Nowy Jork 2013, s. 192-212.

<sup>445</sup> Ç. Çıdam, *A politics of love? Antonio Negri on revolution and democracy*, „Contemporary Political Theory” 2013, nr 12, s. 29-31.

<sup>446</sup> T. Terranova, *Free Labor...*, dz. cyt. s. 33.

*a Playground and a Factory*<sup>447</sup>. Jej tematem przewodnim jest analiza tego, jak we współczesnej sieci internetowej przenika się przestrzeń rozrywki i kreatywności z kontrolą i nieodpłatną pracą. Omawiane przeze mnie w niniejszym rozdziale przykłady projektów artystyczno-badawczych również miały na celu przedstawienie tego jak powszechne platformy cyfrowe z jednej strony inwigilują i kontrolują swoich użytkowników na rzecz generowania zysków, a z drugiej zaś, jak ogromny wymiar może przybrać zgromadzenie efektu pracy, wykonywanej przez użytkowników w postaci prozaicznych zadań, co nie daje się zauważyć z perspektywy jednostki. W internecie jako fabryce społecznej relacje produkcji zawsze są relacjami władzy, które przenikają głęboko życie codzienne i związane z nim aktywności i zacierają granicę pomiędzy rekreacją, komunikacją a wytwarzaniem kapitału.

Przywoływany już przeze mnie badacz, Nick Srnicek, analizując ekonomiczne strategie działania platform cyfrowych, wyraźnie wycofywał się z określania mianem pracy aktywności sieciowych użytkowników, mimo że, jak sam podkreślał, to właśnie na ich podstawie generowany jest przychód dla platform.<sup>448</sup> Ekonomista podkreślał, że w tym kontekście nie można stosować określenia „praca”, ponieważ brakuje tutaj typowych standardów czasu pracy oraz presji konkurencyjnej, motywującej pracowników do tego, aby robili więcej, co, według niego, ma być podstawą procesów produkcyjnych. Jego stanowisko sytuje się w kontrze do krytyczek kapitalizmu informacyjnego, takich jak Tristana Terranova, która podkreślała to że mechanizmy organizacji darmowej pracy na platformach cyfrowych mają postać dyskretną, a nawet utajoną,<sup>449</sup> czy też Shoshana Zuboff, która w ramach koncepcji kapitalizmu inwigilacji pokazywała, że to właśnie dogłębna kontrola aktywności użytkowników jest podstawowym źródłem generowania zysków w tym modelu ekonomicznym.<sup>450</sup> Ponadto, teza Srnicka, dotycząca tego, że aktywności użytkowników nie można nazwać pracą, ponieważ brakuje jej presji motywującej do działania kłóci się z postulatami badaczek i badaczy zajmujących się chociażby analizą tego, jak pasja społeczności fanowskich jest kapitalizowana na platformach blogowych<sup>451</sup>, czy też tego, w jaki sposób gry komputerowe są mechanizmami podtrzymywania zaangażowania graczy.<sup>452</sup>

---

<sup>447</sup> *Digital Labor...*, dz. cyt.

<sup>448</sup> N. Srnicek, dz. cyt., rozdz. 2.

<sup>449</sup> T. Terranova, *Free Labor...*, dz. cyt.

<sup>450</sup> S. Zuboff, dz. cyt.

<sup>451</sup> A. De Kosnik, *Fandom as Free Labor*, w: *Digital Labor: Internet as a Playground and a Factory*, Routledge, Nowy Jork 2013, EPUB, rozdz. 6.

<sup>452</sup> C. Silpasuwanchai, X. Ren, *A Quick Look at Game Engagement Theories*, w: *The Wiley Handbook of Human Computer Interaction*, red. K.L. Norman, John Wiley & Sons, Nowy Jork, s. 657-679.

Dlatego też dalsza część rozdziału poświęcona jest ruchom społecznym, których misją było i jest poszukiwanie sposobów produkcji i modusów komunikacji alternatywnych wobec powszechnego, kapitalistycznego modelu ekonomicznego, który realizuje swoje cele za pośrednictwem oprogramowania i narzędzi cyfrowych.

### 5.5 W poszukiwaniu alternatyw: wolne oprogramowanie

Charakterystyczne dla epoki przemysłowej podejście do robotnika i wytwarzania kapitału, nastawione na eksploatację siły produkcyjnej i maksymalizację zysków, cały czas jest żywe w kontekście współczesnym, w odniesieniu do użytkownika mediów cyfrowych. Konflikt klasowy i przekonanie, że wytwórcy, robotnicy sami powinni posiadać środki produkcji stał się punktem wyjścia dla wszelkich ruchów lewicowych – zarówno tych, które opisałam w części wprowadzającej, powstałych jako bunt wobec rozwijającej się rewolucji przemysłowej, jak i współczesnych ruchów, starających się wprowadzić formy produkcji i dystrybucji oprogramowania alternatywne względem panującego systemu kapitalistycznego.

Współcześnie wśród badaczek i badaczy cyfrowej kultury można wskazać i takie stanowiska, które podkreślają prospołeczny charakter cyfrowych technologii komunikacyjnych i wychodzą z założenia, że pomimo kapitalistycznych zależności, podstawową tendencją rozwoju sieci jest współpraca między jej użytkownikami. Podkreślali to chociażby autorzy książki *Collaborative Society*, a więc Dariusz Jemielniak i Aleksandra Przegalińska<sup>453</sup> czy też badacz cyfrowej ekonomii, Clay Shirky, który w książce *Cognitive Surplus: How Technology Makes Consumers into Collaborators* zaznaczał, że to właśnie technologie cyfrowe pozwalają użytkownikom wyzwolić się z roli konsumentów do pozycji aktywnych współtwórców.<sup>454</sup> Założenia dotyczące tego, że sam fakt bycia połączonym przez sieć cyfrową w pewien naturalny sposób zapewnia demokratyzację wiedzy i nadaje sieci prospołeczny charakter stoją w kontrze do tez krytyków kapitalizmu informacyjnego. Wspominana już przeze mnie Zuboff jasno podkreślała, że owe sieciowe połączenie „jest obecnie środkiem do realizacji komercyjnych celów osób trzecich”<sup>455</sup>, a platformy komunikacyjne nie dążą do demokratyzacji relacji społecznych i wartości kulturowych, ale

---

<sup>453</sup> D. Jemielniak, A. Przegalińska, *Collaborative Society*, The MIT Press, Cambridge, Mass. 2020.

<sup>454</sup> C. Shirky, *Cognitive Surplus: How Technology Makes Consumers into Collaborators*, Penguin Books, Nowy Jork 2010.

<sup>455</sup> S. Zuboff, dz. cyt., s. 21.

do sprawowania kontroli poprzez inwigilację użytkowników. Podobne stanowisko przyjmuje chociażby Evgeny Morozov, badacz społecznych i politycznych oddziaływań cyfrowych technologii, który w książce *The Net Delusion: The Dark Side of Internet Freedom* opisywał mechanizmy tłumienia wolności słowa, inwigilacji, rozpowszechniania ideologicznej propagandy czy też uspokajania ludności za pomocą cyfrowej rozrywki.<sup>456</sup> Platformizacja staje się centralną kategorią współczesnej politycznej ekonomii, co zostało postawione jako główny problem w publikacji *The Politics of Platformization. Amsterdam Dialogues on Platform Theory*, która ma postać zbioru wywiadów przeprowadzonych przez Gianmarco Cristofari'ego z badaczami zorganizowanymi wokół *University of Amsterdam* (UvA), którzy wpisują się w rozwijający się w ostatnim czasie nurt badań *platform studies*.<sup>457</sup>

Poszukiwanie alternatyw wobec kapitalizmu platform cyfrowych może przybierać różne formy ekspresji i realizacji, jak chociażby może sprowadzać się do próby ucieczki i separacji od dominującego modelu polityczno-ekonomicznego, tak jak proponuje to Jaron Lanier, autor książki *Ten Arguments for Deleting Your Social Media Accounts*.<sup>458</sup> Amerykański informatyk argumentuje za tym, że metodą wyzwolenia się spod uzależniającego wpływu kapitalistycznych platform jest usunięcie wszystkich kont, a więc wycofanie się z uczestnictwa w obiegu informacji zapośredniczonego przez media społecznościowe. Geoff Cox określał akt usuwania kont społecznościowych mianem „wirtualnego samobójstwa”, podkreślając przy tym, że rezygnuje się w ten sposób z uczestnictwa w znaczącej części życia społecznego.<sup>459</sup> Badacz zaznaczał tym samym, że tego rodzaju radykalne wycofywanie się posiada miano rozstrzygającego aktu politycznego. Z kolei duński teoretyk mediów, Geert Lovink, podkreślał, że usunięcie kont i aplikacji, związanych z portalami społecznościowymi nie jest ani proste, ani do końca możliwe, ponieważ poszukując metod na komunikację i współuczestnictwo w kulturowej wymianie informacji najprawdopodobniej zostanie się z powrotem wciągniętym do kapitalistycznego modelu platformowego.<sup>460</sup> Zasadnicza teza Lovinka opiera się na przekonaniu, że internet potrzebuje restrukturyzacji, która polegałaby na zachowaniu wszystkiego, co jest społecznie użyteczne w obecnych mediach społecznościowych, ale zorganizowaniu ich na nowo

---

<sup>456</sup> E. Morozov, *The Net Delusion: The Dark Side of Internet Freedom*, Public Affairs, Nowy Jork 2012.

<sup>457</sup> *The Politics of Platformization. Amsterdam Dialogues on Platform Theory*, red. G. Cristofari, Institute of Network Cultures, Amsterdam 2023.

<sup>458</sup> J. Lanier, *Ten Arguments for Deleting Your Social Media Accounts*, Vintage, Londyn 2019.

<sup>459</sup> G. Cox, *Virtual Suicide as Decisive Political Act*, w: *Activist Media and Biopolitics*, red. W. Sützl, T. Hug, University of Innsbruck Press, Innsbruck 2011, s. 105.

<sup>460</sup> G. Lovink, *Stuck on the Platform. Reclaiming the Internet*, Valiz, Amsterdam 2012.

w sposób kolektywny i zdecentralizowany.<sup>461</sup> Jest to bardzo utopijny postulat, szczególnie, że badacz odnosi się do tego, że podobnie woda i energia są organizowane w sposób kolektywny, a nie sprywatyzowany, co jednak nie jest przecież wyrazem prospołecznych przedsięwzięć, ale państwowo zarządzanej gospodarki.

W dalszej części rozdziału będę przedstawiała alternatywne strategie wytwarzania i dystrybucji oprogramowania względem dominujących, kapitalistycznych modeli ekonomicznych, jednak nie będą one sytuować się po stronie radykalnych propozycji czy to odseparowania się od dominujących sieci komunikacyjnych, czy też tworzenia struktur całkowicie od nowa. Skupię się bowiem na ruchu związanym z wolnym oprogramowaniem, który stara się wytwarzać nowe, zdecentralizowane formy dystrybucji informacji i wiedzy w ramach systemu kapitalistycznego, korzystając z wytworzonych w jego obrębie narzędzi kontroli (takich jak chociażby licencje) po to, aby je zredefiniować i ustawić w nowym kontekście, przyczyniając się tym samym do tworzenia nowych relacji społecznych.

### **5.5.1 Richard Stallman, *Free Software*, GNU, GPL i *copyleft***

Jednym z pionierów rewolucyjnego ruchu społecznego, mającego na celu wytworzenie przeciwnych względem kapitalizmu metod wytwarzania oprogramowania, był amerykański haker i jeden z twórców ruchu wolnego oprogramowania, Richard Stallman. Od 1971 roku pracował jako programista w Laboratorium Sztucznej Inteligencji MIT, gdzie przede wszystkim projektował i rozwijał systemy operacyjne. Porzucił jednak to stanowisko w 1984 roku, aby poświęcić się pracy nad rozwijaniem projektu wolnego systemu operacyjnego GNU. Sam Stallman opisując swoją decyzję związaną z wyborem drogi życiowej, wskazuje na to, że widział dwie możliwości: pierwszą z nich było dołączenie do komercyjnego świata oprogramowania prawnie zastrzeżonego (co musiałoby się wiązać z zerwaniem relacji z kulturą hakerską, do której należał), natomiast drugą opcją było na własną rękę rozwijanie modelu tworzenia i dzielenia się oprogramowaniem, zgodnym z jego własną etyką. Haker przyznaje, że pierwsza opcja, związana ze stabilnością na rzecz klauzuli poufności kusila, go, jednak nie miałby wówczas czystego sumienia:

---

<sup>461</sup> Zob. G. Lovink, *We must reclaim the internet, before it is too late*, 12.10.2022, online: <https://www.uva.nl/en/shared-content/faculteiten/en/faculteit-der-geesteswetenschappen/news/2022/10/we-must-reclaim-the-internet-before-it-is-too-late.html>, dostęp: 17.12.2023.

Mogłem w ten sposób zarobić pieniądze i być może zabawić się pisanie kodu. Wiedziałem jednak, że pod koniec mojej kariery spojrzę wstecz na lata budowania murów dzielących ludzi i będę miał poczucie, że spędziłem życie, czyniąc świat gorszym miejscem.<sup>462</sup>

Swoje przekonania etyczne, związane z produkcją oprogramowania wyraził poprzez stworzenie wolnego systemu operacyjnego GNU, którego powstanie ogłosił w 1983 roku. System był złożony wyłącznie z wolnego oprogramowania i był pierwszym projektem założonej przez Stallmana *Free Software Foundation*. Zasady wytwarzania i dzielenia się wolnym oprogramowaniem zostały spisane w postaci *General Public License* (GPL), która była pierwszą licencją typu *copyleft* i stanowi fundamentalny zapis zasad etyki wolnego oprogramowania. GPL zastrzega prawo do czterech wolności: wolności uruchamiania, używania programu w dowolnym celu, wolności modyfikowania oprogramowania w celu dostosowywania do swoich potrzeb, wolności redystrybucji kopii za darmo lub komercyjnie oraz wolności redystrybucji zmodyfikowanego programu tak, by społeczność skorzystała z jego udoskonaleń.<sup>463</sup>

Celem GNU i GPL była wolność użytkowników i ochrona oprogramowania przed prywatyzacją. Główną metodą, zapewniającą osiągnięcie tych celów było wykorzystanie prawa autorskiego (*copyright*), które pozwala twórcom decydować o tym, na jakich zasadach można korzystać z jego dzieła. W przypadku wielkich firm oferujących usługi cyfrowe pozwala ono uniemożliwiać użytkownikom dostęp do oprogramowania, i co za tym idzie, ukrywać specyfikę działania systemu. Stallman wykorzystał to samo narzędzie, jakim jest prawo autorskie, aby sformułować licencję *copyleft*, która pozwala korzystać z dzieł nią objętych tylko, jeśli zachowa się zawarte w nich prawo do wolności i nie wprowadzi się żadnych restrykcji.

Stallman stał się ikoną ruchów wolnościowych związanych z dystrybucją oprogramowania i zainspirował powstawanie kolejnych tego rodzaju oddolnych, zdecentralizowanych przedsięwzięć, m.in. ruch otwartego oprogramowania (*open-source*) zainicjowany przez fińskiego programistę, Linusa Torvaldsa, twórcę systemu operacyjnego Linux. Podstawowym postulatem Stallmana była idea, że oprogramowanie nie powinno posiadać żadnych „właścicieli” tak, aby było powszechnie dostępne dla każdego.<sup>464</sup> Dzięki temu, że nie będzie należało do nikogo, każdy będzie mógł mieć do niego dostęp.

---

<sup>462</sup> *Free Software, Free Society, selected essays of Richard Stallman*, red. J. Gay, Free Software Foundation, Boston, Mass. 2002, s. 19.

<sup>463</sup> Tamże, s. 20.

<sup>464</sup> Zob. R. Stallman, *Why software should have no owners*, online: <https://www.gnu.org/philosophy/why-free.html>, dostęp: 06.11.2023.

### 5.5.2 Dmytri Kleiner, *venture communism* i *copyfarleft*

Gustav Landauer, niemiecki działacz anarchizmu, walczący o prawa robotników na przełomie XIX i XX wieku, powiedział, że „Państwo jest stosunkiem społecznym; pewnym sposobem odnoszenia się ludzi do siebie. Można go zniszczyć przez tworzenie nowych stosunków społecznych; tzn. przez to, że ludzie zawierają inne relacje, zachowują się inaczej”<sup>465</sup>. Na cytata ten powoływał się Dimitri Kleiner, autor manifestu *Telekommunist Manifesto* (2010), argumentował na rzecz tego, że sieci komputerowe, podobnie jak systemy ekonomiczne, można postrzegać jako wyraz relacji społecznych.<sup>466</sup> Z tej perspektywy, przekształcanie zasad dystrybucji oprogramowania i praw jego własności w ramach sieci komputerowych przekłada się bezpośrednio na stworzenie nowego porządku społecznego, który umożliwia ukształtowanie się nowego rodzaju społeczeństwa. W ramach propozycji sformułowanej przez badacza, jest to społeczeństwo oparte na wspólnocie równych sobie dystrybutorów i użytkowników oprogramowania. Kleiner pisał, że „Jedynym sposobem na zmianę społeczeństwa jest produkowanie i dzielenie się w inny sposób.”<sup>467</sup> Programista podkreślał, że relacje społeczne zawierają się również w oprogramowaniu, a więc w tym, co technologiczne, ponieważ pośredniczy ono w procesie produkcji i komunikacji. Pogląd ten reinterpretuje przekonania Marksa, który lokował społeczne relacje związane z produkcją poza narzędziami i maszyną.

Jak sugeruje Kleiner, mimo że sam podział na klasy przestał być oczywisty we współczesności, pewne wątki z nim związane cały czas pozostają aktualne. Należą do nich kwestia statusu osób wytwarzających kapitał i zagadnienie tego, kto nim ostatecznie dysponuje. Badacz nie zrezygnował z określenia „klasa robotnicza” w kontekście współczesnym, w odniesieniu do użytkowników sieci. W napisanym przez siebie manifestie, w którym sformułował swoje postulaty dotyczące polityczności i ekonomii internetu, wskazywał na to, że:

Kondycja klasy robotniczej w społeczeństwie to w dużej mierze bezsilność i bezradność; nie inaczej jest z kondycją klasy robotniczej w internecie. Kontrola i przywileje, których wymaga kapitalizm, są narzucane w internecie, zmieniając topologię sieci z takiej, w której komunizm typu peer-to-peer jest wpisany w jej architekturę, na taką, w której aplikacje typu klient-serwer stały się centralne i w coraz większym stopniu pośredniczą i kontrolują wszystkie relacje.<sup>468</sup>

---

<sup>465</sup> G. Landauer, *Revolution and Other Writings: A Political Reader*, red. tłum. G. Kuhn, PM Press, Oakland 2010, s. 214.

<sup>466</sup> D. Kleiner, *The Telekommunist Manifesto*, dz. cyt., s. 8; 50.

<sup>467</sup> Tamże.

<sup>468</sup> D. Kleiner, *The Telekommunist Manifesto...*, dz. cyt., s. 14. Tłumaczenie A.P.

Przywoływany tu *Telekommunist Manifesto* powstał w inspiracji manifestem Partii Komunistycznej utworzonym przez Marksa i Engelsa.<sup>469</sup> Celem jego powstania było sformułowanie propozycji dla użytkowników funkcjonowania w sieci internetowej w sposób alternatywny wobec kapitalistycznego i dominującego układu klient-serwer. Architektura systemu typu klient-serwer charakteryzuje się tym, że komunikacja wewnątrz niej dzieli się na stronę klienta, po której leży wysłanie żądanie do serwera, i która oczekuje odpowiedzi oraz na stronę serwera, który odbiera żądania, przetwarza je i wysyła odpowiedź. W ramach tego układu serwer jest jednostką nadrzędną, kontrolującą przebieg komunikacji i decydującą o przyznaniu określonych dostępów w ramach działania systemu. Alternatywnym układem wobec wyżej opisanego jest architektura typu *peer-to-peer*, reprezentowana przez sieci rówieśnicze, które łączą w sobie wiele komputerów użytkowników, stanowiących węzły w transferze danych. Każdy host jest sobie równy i może pełnić rolę zarówno serwera, jak i klienta. Najpopularniejszym zastosowaniem architektury *peer-to-peer* są systemy oparte na wymianie plików (np. BitTorrent) i komunikatory (np. Skype). Autor *Telekommunist Manifesto* podkreśla to, że obecnie w internecie dominują kapitalistyczne systemy typu klient-serwer, które albo wymagają opłat za korzystanie z nich, albo monetyzują aktywność swoich użytkowników w ramach ekonomii uwagi i personalizacji reklam.

Kleiner jest założycielem berlińskiej grupy *Telekomunisten Collective*, powstałej w Berlinie w 2005 roku. Celem inicjatywy jest wypracowanie sposobów funkcjonowania w sieci będących alternatywą wobec dominujących modeli kapitalistycznych. Grupa działała w oparciu o dwie główne idee: *venture communism*<sup>470</sup> oraz *copyfarleft*<sup>471</sup>, obie zostały szczegółowo opisane w ramach *Manifestu Telekomunistycznego*. Stanowią one

---

<sup>469</sup> Ciekawym fragmentem wieloczęściowego manifestu jest rozdział, w którym Kleiner reinterpretuje i uaktualnia drugą sekcję Manifestu Komunistycznego Marksa i Engelsa poprzez podmienienie niektórych wyrażeń, np. „zniesienie własności prywatnej” zostało zastąpione przez „uwspólnienie”, zamiast „publiczny” autor użył określenia „wspólny”, wyrażenie „państwo” zostało zastąpione przez „wspólnoty”, itd. W rozdziale zawarty został oryginalny tekst Marksa i Engelsa z wykreślonymi fragmentami i zaproponowanymi alternatywnymi wyrażeniami. Zob. Tamże, s. 26-27.

<sup>470</sup> Tamże, s. 8.

<sup>471</sup> Tamże, s. 40-43.

dekonstrukcje dwóch idei kapitalistycznych – *venture capitalism*<sup>472</sup> oraz *copyright*<sup>473</sup>, stojących w kontrze do lewicowej kultury sieciowej. Zaproponowany przez Kleinera model *venture communism* opiera się na organizowaniu komun, połączonych ze sobą siecią typu *peer-to-peer*. W porównaniu do modelu kapitalistycznego, owe komuny można rozumieć jako firmy, posiadają one jednak inaczej zdefiniowane cele, nastawione na przywrócenie wartości pracy, a nie na jej wyzysku.<sup>474</sup> Tego rodzaju struktura różni się od kolektywu, czy spółdzielni, w ramach których własność jest posiadana indywidualnie przez członków. W komunie typu „venture” dostęp do dóbr można uzyskać jedynie przez wkład pracy, a nie przez własność.<sup>475</sup> Jest to przeciwny rodzaj inwestowania od tego reprezentowanego przez *venture capitalism*, w którym inwestowanie ma charakter kapitałowy. W kapitalizmie praca nie jest należycie opłacana – zasadniczym sposobem bogacenia się jest wynagradzanie pracy w mniejszym stopniu, niż jest tego warta, a więc poniżej wartości wytwarzanego produktu. *Venture communism* uznaje pracę za najwyższą wartość inwestycyjną i nadaje jej wspólnototwórczy charakter. Kwestie własności i stosunku do produkcji Kleiner wyjaśnia następująco:

Komuna typu „venture” jest właścicielem wszystkich aktywów produkcyjnych które składają się na wspólny zasób wykorzystywany przez zróżnicowaną i rozproszoną geograficznie sieć kolektywnych i niezależnych rówieśniczych producentów (*peer producers* – dop. A.P.). Komuna nie koordynuje produkcji, społeczność producentów produkuje zgodnie z własnymi potrzebami i pragnieniami. Rolą komuny jest jedynie zarządzanie wspólnym zasobem, udostępnianie producentom własności, takiej jak mieszkania i narzędzia, których potrzebują.<sup>476</sup>

Jak daje się odczytać z tego opisu, *venture communism* pozwala organizować wspólnoty przedsięwzięć, w których wkładem jest praca, i które dzielą się własnością, nie pozwalając przy tym, aby gromadziła się ona w rękach jednego człowieka lub niewielkiej liczby osób.

---

<sup>472</sup> *Venture capitalism* to powszechny, szczególnie w sektorze technologicznym, model ekonomiczny, opierający się na poszukiwaniu inwestorów, którzy przekazują firmie prywatne środki na rzecz jej rozwoju, w zamian za co zyskują procent udziału w jej zyskach. Model ten jest najbardziej popularny wśród startupów, a więc nowych, małych firm, które opierają się na konkretnym pomysle na produkt, czy usługę, licząc na uzyskanie zainteresowania inwestorów. Model ten jest związany z ryzykiem, ponieważ nie ma gwarancji, że firma, której inwestor przekazuje prywatne środki odniesie sukces ekonomiczny i wybije się na rynku. Zob. A. Ganti, *Venture Capitalists Definition: Who Are They and What Do They Do?*, 2022, online: <https://www.investopedia.com/terms/v/venturecapitalist.asp>, dostęp: 06.11.2023.

<sup>473</sup> *Copyright*, a więc system prawa autorskiego jest zaprzeczeniem lewicowych idei współdzielenia i równego dostępu.

<sup>474</sup> D. Kleiner, *The Telekommunist Manifesto*, dz. cyt., s. 23.

<sup>475</sup> Tamże, s. 24.

<sup>476</sup> Tamże, s. 23.

W odniesieniu do produkcji wartości intelektualnych i pracy niematerialnej Kleiner opracował typ licencji *copyfarleft*. Stanowi ona reinterpretację stworzonej wcześniej licencji *copyleft*, która jako pierwsza wykorzystywała prawo własności, będące zasadniczym narzędziem kontroli zysku i dostępu w ramach kapitalistycznej ekonomii, do tego aby stworzyć antykapitalistyczny, lewicowy model dzielenia się i udostępniania oprogramowania. Oprogramowanie objęte licencją *copyleft* jest udostępniane do powszechnego wglądu, wykorzystywania i przetwarzania pod jednym warunkiem – wszystkie pochodne tych procesów muszą być również dystrybuowane na zasadach licencji *copyleft*. W tym wypadku praca staje się zbiorowym zasobem, dostępnym, dla wszystkich twórców wolnego oprogramowania. Nasuwa się tu pytanie o to, dlaczego model o ideach tak bliskich założeniom *venture communism* potrzebował, w ujęciu Kleinera, reinterpretacji. Programista nie neguje zasadności tej licencji, a wręcz podkreśla jej wartość słowami:

*Copyleft* skutecznie przejmuje istniejący aparat, który egzekwuje przywileje dotyczące dóbr intelektualnych, wykorzystując uprawnienia przyznane przez licencję praw autorskich do zagwarantowania dostępu dla wszystkich i wymagając, by ta wolność była przekazywana dalej.<sup>477</sup>

Jednak autor *Manifestu Telekomunistycznego* słusznie zauważa, że choć *copyleft* skutecznie spełnia zadanie tworzenia wspólnego dobra w postaci oprogramowania, to okazuje się jednak ono niewystarczające w szerszej skali, np. w obszarze tworzenia wspólnego dobra w postaci dzieł kultury.<sup>478</sup> Niewystarczalność *copyleft* wynika z tego, że pozwala na to, aby dzięki niej kapitalistyczne firmy i organizacje poprzez korzystanie z dostępnych zasobów wolnego oprogramowania realizowały cele komercyjne i bogaciły się, według zasady, że każdy ma dostęp do oprogramowania. Podstawowym celem sformułowanej przez Kleinera licencji *copyfarleft* była odmowa dostępu do wytwarzanego na jej zasadach oprogramowania organizacjom, które posiadają własne zasoby poza tymi dzielonym w ramach licencji. Przywoływany już przeze mnie badacz oprogramowania Geoff Cox, określił *copyfarleft* jako spełnienie Hegłowskiej koncepcji podwójnej negacji, zgodnie z którą aby skutecznie uśmiercić określony system, nie wystarczy go raz zanegować, ale trzeba potem zanegować jego negację.<sup>479</sup> Podstawą tego procesu, co również tłumaczył między innymi Slavoj Žižek, jest fakt, że pojedyncza negacja opiera się zazwyczaj na

---

<sup>477</sup> Tamże.

<sup>478</sup> Tamże, s. 28.

<sup>479</sup> Zob. G. Cox, A. McLean, dz. cyt., s. 54.

stworzeniu prostego przeciwieństwa (dla przykładu – robotnicy wyobrażający sobie świat bez kapitalistów), a dopiero dekonstrukcja tak wytworzonej negacji pozwala we właściwy sposób przemodelować oryginalny system (dla przykładu – uświadomienie sobie, że bez kapitalistów nie istnieją również robotnicy, w efekcie poszukiwania jeszcze innego modelu).<sup>480</sup> W odniesieniu do *copyfarleft* nie można powiedzieć, że absolutnie neguje ono *copyleft*, ale że poszerza ją i umożliwia skuteczniejsze zanegowanie licencji *copyright*.

Założony przez Kleinera Telekommunisten Collective realizował idee zarówno *venture communism* jak i *copyfarleft*. Grupa podejmowała partyzanckie działania, wśród których wymienić można stworzenie własnej sieci internetowej i telefonicznej Telekommunisten.net. Innym przykładem budowania platform komunikacyjnych o wspólnotowym i rówieśniczym charakterze był *Thimbl* (2010-2011)<sup>481</sup>. Był to projekt platformy blogowej *open-source*. Jej serwer został zbudowany za pomocą oprogramowania o nazwie Finger, dostosowanego do zasad otwartej sieci rówieśniczej typu *peer-to-peer*. W przeciwieństwie do popularnych platform społecznościowych, *Thimbl* pozwalał na jednoczesne publikowanie własnych treści oraz współposiadanie samego serwisu, razem ze wszystkimi innymi jego użytkownikami.

Kolektyw angażował się również w działalność w obszarze sztuki nowych mediów, jedną z najsłynniejszych prac grupy jest *deadSwap* (2009) oraz wznowienie projektu w postaci *deadSwap 2.0* (2015), prezentowane w ramach festiwalu Transmediale w Berlinie. Obie inicjatywy były eksperymentalną próbą tworzenia kanałów komunikacji i wymiany, które stanowiłyby antagonistyczną propozycję względem skomercjalizowanych sieci. Projekty grupy Telekommunisten Collective zapewniały swoim użytkownikom swego rodzaju azyl. W pierwszej odsłonie *deadSwap* polegał na tym, że uczestnicy zaproszeni zostali do tego, aby przekazywać sobie pamięć USB, a sposób i kierunek przekazywania koordynowany był przez anonimową bramkę SMS-ową. W ten sposób powstał system dzielenia się danymi, w którym transfer plików odbywał się całkowicie poza zasięgiem internetu. Eksperyment społeczny sprawdzał możliwości funkcjonowania zdecentralizowanej sieci wymiany, działającej offline, w której zasadnicza informacja, będąca przedmiotem przekazu zapisana została na materialnym nośniku, a cały proces nie wymagał połączenia internetowego. W ten sposób eksperyment omijał centralnie działające, skomercjalizowane sieci komunikacyjne na rzecz stworzenia własnej alternatywy. W drugiej

---

<sup>480</sup> Więcej o koncepcji negacji negacji można przeczytać w rozdziale poświęconym Heglowi *The Hegelian Ticklish Subject*, w: S. Žižek, *The Ticklish Subject. The Absent Centre of Political Ontology*, Verso, Londyn, Nowy Jork 2000, s. 70-75.

<sup>481</sup> *Thimbl* był dostępny pod adresem: <http://www.thimbl.net>.

odslonie projektu *deadSwap 2.0*, koordynacja przez system SMS-owy zastąpiona została aplikacją dedykowaną systemom Android, za pomocą której uczestnicy mogli koordynować obieg pamięci USB.

W kontekście aktywności grupy Telekommunisten Collective, sam Kleiner podkreślał jednak, że nie zakłada, że jego działania wykraczają poza system kapitalistyczny, ale określał je raczej jako wytwarzanie zakłóceń wewnątrz istniejącego systemu. Podkreślał to w stwierdzeniu, że: „nie istnieje pojęcie zewnątrz kapitalizmu, są tylko sprzeczności wewnątrz systemu”.<sup>482</sup> Jednocześnie haker zakładał, że nie istnieje również absolutny kapitalizm, ponieważ sprzeczne systemy ekonomiczne mogą funkcjonować równolegle w tym samym czasie.<sup>483</sup>

Dmytri Kleiner nie uważa, że rewolucja systemu ekonomicznego miałaby przebiegać poprzez wybranie innej władzy, jak na przykład partii neoliberalnej czy lewicowej. Jego zdaniem jedyne słuszne rozwiązanie nie ma charakteru systemowego, ale wspólnotowy. Kluczowym problemem jest fakt, że pracownicy/użytkownicy nie kontrolują tego, co wytwarzają. W myśl jego teorii rewolucja ekonomiczna, która pozwoliłaby faktycznie oprzeć się dominacji kapitalizmu miałaby polegać na przejęciu kontroli nad kapitałem przez użytkowników sieci. Zaproponowany przez Kleinera model *venture communism* pozwala pracownikom na zakładanie sieci o charakterze wspólnotowym, przejmowanie środków produkcji i definiowanie zasad wzajemnej wymiany.

Struktura opisanego przez Kleinera modelu *venture communism* przypomina swoją architekturą wczesny internet. Na początku swojego rozwoju internet działał dzięki chałupniczej branży niewielkich, rozproszonych dostawców, którzy później zastąpieni zostali przez centralnie zarządzane, monolityczne firmy. Autor *Telekommunist Manifesto* również odwoływał się do okresu początków internetu, podkreślając przy tym, że nie miał on od zawsze tak skomercjalizowanego charakteru, jak ma to miejsce współcześnie. Internet miał być wręcz swego rodzaju zaskoczeniem dla świata korporacyjnego, ponieważ jego projekt został opracowany w ramach finansowanej ze środków publicznych uczelni wyższej, na zlecenie organów wojskowych.<sup>484</sup> Jego genezę można by więc umiejscowić bardziej po stronie wartości społeczeństwa obywatelskiego, reprezentowanych przez uniwersytet, niż kapitalistycznego świata dużych firm. Użytkownicy wczesnego internetu posiadali wybór

---

<sup>482</sup> D. Kleiner, *Nie istnieje coś takiego, jak zewnątrz kapitalizmu*, rozm. przepr. E. Drygalska, 19.10.2015, online: <https://popmoderna.pl/nie-istnieje-cos-takiego-jak-zewnetrze-kapitalizmu-wywiad-z-dmytri-kleinerem/#>, dostęp: 06.11.2023.

<sup>483</sup> Tamże.

<sup>484</sup> D. Kleiner, *The Telekommunist Manifesto*, dz. cyt., s. 14.

czy korzystać z usług wchodzących na rynek dużych dostawców (takich jak np. CompuServe) i korzystać z narzędzi i platform proponowanych przez nich, czy też zdecydować się na mniejszego, niezależnego dostawcę usług lub nawet uruchomić swój własny serwer. Wraz ze swoim rozwojem, sieć ulegała stopniowym przekształceniom, a w ich rezultacie bezpośrednia komunikacja i wymiana danych między użytkownikami, bliska modelowi *peer-to-peer*, zastępowana była służącej celom kapitalistycznym, centralnie zarządzanej strukturze klient-serwer. Działania Kleinera w ramach grupy Telekommunisten Collective ukierunkowane są na to, by odzyskać ów tłumiony przez kapitalistyczne wpływy wspólnotowy charakter sieci.

Mimo że ruchy społeczne, które starają się wprowadzić dysonans w działaniu kapitalistycznego modelu ekonomicznego są bardzo ważne i przyczyniają się do znaczących zmian społeczno-kulturowych, to jednak nie należy do nich podchodzić bezkrytycznie. Tak jak podkreślał Kleiner, tego rodzaju rewolucyjne działania nie sytuują się gdzieś poza modelem kapitalistycznym, ale są jego częścią. Konsekwencją takiego układu jest to, że może dochodzić do nadużyć, w związku z którymi oprogramowanie dystrybuowane pod znakiem wolnej licencji może i tak zostać przejęte przez prywatne firmy lub też może dochodzić do nieporozumienia, w związku z którym przedsięwzięcie deklarujące się jako wolnościowe jest w istocie zamaskowanym modelem kapitalistycznym.

### **5.5.3 Przejmowanie wartości wolnego oprogramowania przez kapitalizm**

Jednym z przykładów tego, w jaki sposób ogromne, kapitalistyczne konglomeraty przejmują wartości wytwarzane przez wolne oprogramowanie jest przypadek Usenetu. Jest to system przesyłania wiadomości i dzielenia się informacjami oraz plikami, działający od 1979 roku, a więc od samego początku rozwoju internetu. Jego założycielami byli trzech doktoranci: Tom Truscott i James Ellis z Uniwersytetu Duke oraz Steve Bellovin z Uniwersytetu Północnej Karoliny. Stworzyli oni ten system, aby usprawnić komunikację pomiędzy swoimi uczelniami. Współcześnie rola platformy straciła na znaczeniu, chociaż wiele grup dyskusyjnych wciąż pozostaje aktywnych. Usenet jest systemem rozproszonym, nie stanowi niczyjej własności ani nie jest przez nikogo kontrolowany. Właśnie brak centralnej kontroli i własności odróżnia go od powszechnych konsumpcyjnych platform komunikacyjnych. Jednak przedstawiciele Google znaleźli sposób, aby częściowo podporządkować Usenet własnemu monopolowi poprzez wdrożenie funkcjonalności Google Groups, dzięki której

można przeglądać grupy dyskusyjne Usenetu. Google Groups stało się najczęściej używanym klientem sieciowym Usenetu, a firma Google weszła w posiadanie największego archiwum tejże platformy. System komunikacyjny nie został jednak całkowicie zawłaszczony przez ogromne przedsiębiorstwo, ze względu na rozproszoną strukturę tego pierwszego. Równolegle nadal istnieją środki dostępu do sieci komunikacyjnej Usenet inne niż poprzez Google Groups.

Przejmowanie wartości wolnego oprogramowania przez model kapitalistyczny może przybierać również bardziej dyskretną postać i rozgrywać się na płaszczyźnie ideologicznej. Podmioty realizujące model kapitalistyczny mogą manipulować znaczeniami i reprezentować się jako projekt wolnościowy, przez co wprowadzać użytkowników w błąd. Jest to działanie szkodliwe na poziomie społecznym, ponieważ doprowadza do dezinformacji i mylnego obrazu ruchów wolnościowych dotyczących oprogramowania. Doskonałym przykładem takiego mechanizmu jest funkcjonowanie platform w ramach szeroko pojętego Web 2.0. W opisie idei Web 2.0 użytkownicy określani są mianem „współprogramistów” w odniesieniu do tego, jak „współtworzone” są produkty oprogramowania w ramach projektów otwartego oprogramowania (*open-source*), przy jednoczesnym zaznaczeniu, że Web 2.0 nie funkcjonuje w ramach tejże licencji.<sup>485</sup> Cechą charakterystyczną najbardziej wpływowych firm realizujących model Web 2.0 jest to, że umożliwiają społeczności otwarty wkład w udostępnioną platformę cyfrową, po to by następnie wykorzystywać ten wkład w ramach działania zastrzeżonego systemu, którego właścicielem (i co za tym idzie również owych tworzonych treści) jest właściciel platformy. W rezultacie w ramach modelu Web 2.0 pozwala się użytkownikowi współtworzyć platformę i o tyle nazywany jest współtwórcą, jednak nie pozwala mu się posiadać tego, co wytworzył. Uderza to w zasadniczy problem związany z brakiem kontroli nad i dostępu do wytwarzanych treści. W rozdziale czwartym opisywałam przejście od modelu sieci Web 1.0 do Web 2.0 z podkreśleniem tego, że deklaratywnie ten drugi model miał być siecią demokratyczną, siecią użytkownika, w ramach której społeczny charakter internetu miał wziąć górę nad komercyjnym. Jednak społecznościowy charakter Web 2.0 z tej perspektywy jest iluzją, przekazanie użytkownikowi możliwości tworzenia treści, wytworzone w ten sposób mechanizmy współdzielenia, współtworzenia i współpracy są centralnie zarządzane i skomercjalizowane. Web 2.0 nie może być postrzegane jako pewna druga faza

---

<sup>485</sup> T. O'Reilly, *What is Web 2.0. Design Patterns and Business Models for the Next Generation of Software*, 30.09.2005, online: <https://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html?page=4>, dostęp: 06.11.2023.

technicznego czy społecznego rozwoju internetu, raczej można ją porównać do drugiej, bardziej wyrafinowanej, fali kapitalistycznego zawłaszczania wspólnego dobra, jakim jest informacja.<sup>486</sup> Dmitri Kleiner podkreślał w tym kontekście, że:

Misją Web 2.0 jest zniszczenie aspektu P2P w internecie i uzależnienie Ciebie, Twojego komputera i Twojego połączenia internetowego od połączenia ze scentralizowanymi usługami, które kontrolują Twoje możliwości komunikacyjne. Web 2.0 to ruina wolnych systemów peer-to-peer i powrót monolitycznych usług internetowych.<sup>487</sup>

Web 2.0 podając się za ruch oddolny, społecznościowy dostarczyło narzędzi do tego, aby przekształcać wspólne dobro w prywatny zysk i zarządzać na masową skalę wytwarzaniem treści przez użytkowników.

Przejmowanie wartości związanych z wolnym oprogramowaniem przez korporacje i systemy ekonomii kapitalistycznej może posiadać też przeciwny kierunek niż opisane wyżej podszywanie się komercyjnych przedsięwzięć pod społecznościowe. Dochodzi czasem do sytuacji, w której inicjatywy, które z założenia mają realizować postulaty etyki wolnego oprogramowania tracą jednak w zależnościach kapitalistycznych, przez co przestają być rewolucyjne, choć wciąż powszechnie są za takie uznawane. Za przykład posłużyć tu może ciekawy przypadek *Creative Commons*, które nawet w swojej nazwie odwołują się do idei wspólnego tworzenia i dzielenia się ze sobą. Twórca licencji *Creative Commons*, prawnik, Lawrence Lessig starał się uzyskać kompromis pomiędzy pełną ochroną praw autorskich a całkowitym współdzieleniem. W rezultacie *Creative Commons* pozwala na elastyczne dostosowywanie zasad współdzielenia wytwarzanych wartości i posługuje się hasłem „pewne prawa zastrzeżone” (w kontrze do charakterystycznego dla *copyright* „wszystkie prawa zastrzeżone”). Przez swoją elastyczność licencja pozwala współtworzyć zgodnie z etyką współdzielenia, charakterystyczną dla wolnego oprogramowania, jednak może być ona również wykorzystana do tego, aby przemycić pod jej osłoną prywatyzację wartości. Zasadniczą różnicą między licencją *Creative Commons* a *copyleft*, jest to, że nie wymaga ona, aby wraz z wykorzystywaniem stworzonych na jej zasadach dzieł, przekazywać dalej idee otwartości i dostępu. Z tych powodów została w całości odrzucona i skrytykowana przez działaczy związanych z ruchem wolnego oprogramowania m.in. przez Stallmana i Kleinera. Stallman wydał specjalne oświadczenie, w którym odrzuca w całości *Creative Commons*, powołując się na to, że jedno z jego licencji

---

<sup>486</sup> D. Kleiner, *The Telekommunist Manifesto*, dz. cyt., s. 20.

<sup>487</sup> Tamże, s. 19. Tłumaczenie A.P.

są wolne, a inne nie. Zaznaczył on również, że tego rodzaju przedsięwzięcia, które mieszają w sobie część etyki wolnego oprogramowania z mechanizmami charakterystycznymi dla modelu kapitalistycznego, wyrządzają szkodę społeczną, przez to, że użytkownicy myślą, że korzystają z licencji o charakterze wolnościowym, podczas gdy w rzeczywistości nie została ona oparta na żadnym standardzie ani stanowisku etycznym.<sup>488</sup> Z kolei Kleiner podkreślał fakt, że tak jak licencja *copyleft* korzysta z prawa autorskiego w sposób subwersywny, po to aby wykorzystać podstawowe narzędzie kapitalizmu do walki przeciw niemu, tak *Creative Commons* czerpie korzyści z prawa autorskiego i daje twórcom możliwość ograniczania wolności użytkowników:

Podczas gdy *copyleft* rości sobie prawo do własności z prawnego punktu widzenia, by następnie zrzec się go w praktyce, odwołania do własności w *Creative Commons* są autentyczne, a nie ironiczne. Licencje *Creative Commons* pozwalają na arbitralne ograniczanie wolności użytkowników w zależności od upodobań i preferencji danego twórcy. W tym sensie *Creative Commons* jest bardziej rozbudowaną wersją prawa autorskiego. Nie kwestionuje ono systemu praw autorskich jako całości, ani nie zachowuje jego prawnej otoczki, aby wywrócić do góry nogami praktykę stosowania prawa autorskiego, jak to czyni *copyleft*.<sup>489</sup>

W innym miejscu Kleiner ucieka się do obrazowej metafory poprzez porównanie *Creative Commons* do rady robotniczej, która powstaje jako ruch związkowy negocjujący umowy prawne z właścicielami korporacji, ale ostatecznie wchodzi z nimi w układ i zaprzecza swojej podstawowej funkcji, do której została powołana, zamiast tego staje się kolejnym narzędziem ucisku.<sup>490</sup> Podstawową różnicą pomiędzy ruchami realizującymi postulaty etyki wolnego oprogramowania a ruchami kapitalistycznymi jest inny rodzaj wolności, którą starają się osiągnąć. Ci pierwsi dążą do uzyskania wolności konsumenta do używania i produkowania, z kolei ci drudzy do wolności producenta do kontrolowania użytkowników i wytwarzanych przez nich wartości. Na tak wyznaczonej osi tendencji *Creative Commons* zdaje się sytuować po stronie wolności producentów.

Wartością charakterystyczną dla ruchów społecznych związanych z wolnym oprogramowaniem, którą trudno jest przejąć przez inicjatywy kapitalistyczne jest zaangażowanie użytkowników i zamiłowanie do tego, co robią. W ramach pracy na rzecz firm kapitalistycznych pracownicy wykonują swoje obowiązki w ciągu ośmiogodzinnego dnia pracy (lub innej określonej w umowie jednostki czasowej). Można jedynie snuć

---

<sup>488</sup> DaBlade, *Richard Stallman Interview*, 06.02.2006, online: <http://www.p2pnet.net/story/7840>, dostęp: 06.11.2023.

<sup>489</sup> D. Kleiner, *The Telekommunist Manifesto*, dz. cyt., s. 34-35.

<sup>490</sup> Tamże.

przypuszczenia odnośnie tego, jak zmieniłaby się produktywność, gdyby oddać pracownikom całkowitą wolność wyboru tego, kiedy i gdzie chcą pracować. Choć częścią dyskursu kapitalistycznego jest rozpowszechnianie idei, że dobrze jest wykonywać pracę, którą się lubi, to wybór pracy w obrębie oferty firm często podyktowany jest zarobkami, potrzebą utrzymania się, a nie osobistą pasją. Jest to coś przeciwnego w stosunku do ruchu wolnego oprogramowania, który w żaden sposób nie narzuca nikomu przymusu rozwijania go, ani nie kontroluje metod i czasu pracy na to poświęcanego. Jest to inicjatywa zrzeszająca miliony entuzjastów i osób, które z własnej woli przyczyniają się do rozbudowywania projektów wolnego oprogramowania. Z pewnością przynosi to widoczne rezultaty w kontekście efektywności i jakości pracy. Wolne oprogramowanie rozwija się też o wiele szybciej niż produkty komercyjne, co zawdzięcza otwartemu dostępowi, dzięki któremu projekty mogą być przeglądane i poprawiane przez ogromne ilości użytkowników, rozproszonych na całym świecie. Jest to coś, na co kapitalistyczne firmy nie mogą sobie pozwolić, ponieważ muszą chronić swoją własność intelektualną i nad jej rozwojem pracuje zawężony zespół osób.

## **5.7 Podsumowanie**

Przejście od początku rozwoju kapitalizmu przemysłowego (przywoływanego we wprowadzeniu do niniejszego rozdziału) do współczesnych form zarządzania pracą w cyfrowej kulturze miało na celu przedstawienie tego, że mimo znacznego postępu technologicznego, który nastąpił w tym przedziale czasowym, utrwalone na początku XVIII wieku formy eksploatacji pracowniczey i tendencje do maksymalizacji produkcji zostały zachowane, a nawet zyskały nowe formy wyrazu dzięki oprogramowaniu, które pozwala na automatyzację tych procesów w szerokiej skali. Opisany we wstępie ruch luddystyczny był pierwszą, mimo że kontrowersyjną to jednak historycznie znaczącą, reakcją na zachodzącą zmianę modelu ekonomicznego. Rzemieślnicy i wytwórcy bezskutecznie starali się bronić własnego kapitału, wraz z dotychczasowymi strategiami jego wytwarzania. Jednak rewolucja przemysłowa wprowadziła odseparowanie robotnika od produktu własnej pracy, co później Marks określił mianem alienacji. W konsekwencji tego procesu robotnik przestał posiadać to, co sam wytworzył. Właśnie brak kontroli nad produktem własnej pracy stał się podstawą postulatów Marksa i potem kolejnych ruchów społecznych, starających się

przywrócić robotnikom, pracownikom czy użytkownikom kontrolę nad wytwarzaną wartością.

Istotną zmianą społeczną i kulturową, jaką wprowadziła rewolucja przemysłowa było zrównanie człowieka i technologii w procesie produkcyjnym – zarówno pod względem funkcjonalnym, jak i w odniesieniu do ich wartości. Robotnik stał się częścią procesu, ogromnego systemu zależności, w którym ludzie i maszyny wspólnie działają na rzecz wytwarzania wartości. Wyraża się on w postaci fabryki, którą Marks porównał do automatu. Konsekwencje procesu zrównywania człowieka i technologii w obszarze produkcji są zauważalne współcześnie, chociażby na przywoływanych przeze mnie przykładach zarządzania pracą w ramach takich platform jak Facebook czy Amazon, w których użytkownik staje się zasadniczym elementem funkcjonowania serwisu.

Zasadniczą różnicą pomiędzy funkcjonowaniem społeczeństwa na początku kapitalizmu przemysłowego i w rozwiniętym kapitalizmie informacyjnym jest indywidualizacja społeczeństw, która postępowała wraz z rozwojem cywilizacyjnym. Przyczyniła się ona do atomizacji społecznej, w konsekwencji której identyfikacja z określoną klasą, która była możliwa kiedy organizowały się pierwsze ruchy lewicowe, okazała się być znacznie bardziej utrudniona w odniesieniu do współczesnej kultury cyfrowej. Niezmiennie, tak samo na początku rewolucji przemysłowej, jak i w ramach panującego obecnie kapitalizmu informacyjnego, robotnik/pracownik pozbawiony jest możliwości gromadzenia majątku na rzecz utrzymywania się z pensji wyznaczonej przez pracodawcę. Kolejną ważną różnicą pomiędzy tymi dwoma etapami rozwoju przemysłu są formy wykonywanej pracy. Podstawową formą pracy w kapitalizmie przemysłowym była praca fizyczna, podczas gdy kapitalizm informacyjny, dzięki rozwojowi cyfryzacji i temu, że oprogramowanie stało się kluczowym elementem większości czynności życia codziennego pozwolił na zastąpienie w znaczącym stopniu pracy fizycznej przez pracę niematerialną, polegającą na wytwarzaniu wartości intelektualnych. W konsekwencji rozpowszechnienia się pracy niematerialnej, coraz trudniej jest odróżnić pracę najemną od bezpłatnej. W podobny sposób zaciera się granica pomiędzy czasem wolnym, a czasem pracy przez to, że kapitalizacji poddana została większość aktywności życia codziennego, która zapośredniczona jest przez technologię cyfrową.

Od początku lat 80 XX wieku zaczęły powstawać ruchy społeczne, dążące do wytworzenia własnych zasad produkowania i dzielenia się wartościami intelektualnymi, takimi jak oprogramowanie czy informacje. Metody te miały być alternatywne wobec dominującego modelu kapitalistycznego. Ich inicjatorem był Richard Stallman, który

opracował zasady etyczne związane z wolnym oprogramowaniem i był twórcą licencji GPL, która stanowiła pierwszą licencję typu *copyleft*. Zawarty w niniejszym rozdziale opis rozwoju tej inicjatywy i jej kontynuatorów miał na celu przedstawienie tego, że w społeczeństwie cały czas kształtują się formy sprzeciwu wobec panującego modelu ekonomicznego.

Jednak, jak starałam się pokazać w powyższym rozdziale, każda produkcja wolnego oprogramowania, choć jest istotną tendencją, nie może być zawsze uważana w sposób jednoznaczny za alternatywę dla kapitalizmu, ponieważ może ona sama stanowić wyraz nowych form pracy w kapitalizmie. Zaznaczał to już Cox w słowach:

To, co jest uważane za „wolne”, czy to praca, oprogramowanie, czy mowa, jest wyraźnie oparte na infrastrukturze rynkowej, która wykorzystuje prawa własności intelektualnej do realizacji własnych interesów.”<sup>491</sup>

Stanowisko to pokrywa się z założeniem Kleinera, który podkreślał, że nie istnieje zewnątrz kapitalizmu i wszelkie rewolucyjne działania, starające się wprowadzić alternatywny porządek, będą miały charakter sprzeczności wewnątrz systemu – tu z naciskiem na „wewnątrz”.

Kończąc rozważania zawarte w tym rozdziale, chciałabym podkreślić, że oprogramowanie odgrywa kluczową rolę w procesach ekonomiczno-społecznych cyfrowej kultury. Po pierwsze, to właśnie ono umożliwiło rozwój i rozpowszechnienie się pracy niematerialnej. Sam software w świetle prawa autorskiego jest uznawany za wartość niematerialną, intelektualną.<sup>492</sup> Po drugie, dostarczyło ono narzędzi do wspomnianej kapitalizacji życia codziennego, przez to, że pośredniczy w wykonywaniu czynności z nim związanych. Po trzecie, stanowi on wyraz stosunków i relacji społecznych. Ogromne znaczenie dla Kleinera miało to, że jeśli przyjmie się, że sieci komputerowe, tak samo jak systemy ekonomiczne wyrażają relacje społeczne, to redefiniowanie sposobów dystrybucji oprogramowania i praw jego własności przekłada się bezpośrednio na kreowanie nowego porządku społecznego.

---

<sup>491</sup> G. Cox, A. McLean, dz. cyt., s. 51. Tłumaczenie A.P.

<sup>492</sup> Inaczej jest on ujmowany przeze mnie i w badaniach nad kulturowym znaczeniem oprogramowania, które mają na celu zaznaczenie materialności, i, co za tym idzie, sprawczości kulturowej software’u. Nerozerwalność oprogramowania i materialnego hardware’u omawiałam w pierwszym rozdziale, przy okazji przedstawiania mojej definicji software’u.

## **Podsumowanie**

## **Wnioski kończące**

W pracy tej starałam się przedstawić oprogramowanie jako system operacyjny kultury, a więc udowodnić, że software odgrywa kluczową rolę w procesach współczesności opartych o szybki rozwój technologii cyfrowych. Jest bowiem sposobem zarządzania praktykami społecznymi, politycznymi i ekonomicznymi, rejestruje, przetwarza, aktualizuje i kontroluje przepływ wartości w kulturze cyfrowej. Rozprawa ta kontynuuje podejmowane już w szerokim dyskursie dyscypliny *media studies* problematyki, takie jak: interakcja z technologią cyfrową czy funkcjonowanie użytkownika w sieci, temat inżynierii społecznej oraz politycznego wykorzystywania mediów. Jednak zamiast klasycznego, medioznawczego podejścia proponuje ona zaczerpnięte z *software studies* metody problematyzowania cyfrowej kultury z uwzględnieniem roli jaką samo oprogramowanie odgrywa w tych procesach kulturowych. Poniżej wypunktuję wnioski, które wydają się najbardziej kluczowe dla podsumowania wyników pracy.

Po pierwsze, w odniesieniu do drugiego i trzeciego rozdziału tej pracy, należy wyszczególnić tezę o nierozzerwalności i wzajemnym warunkowaniu się procesów wytwarzania i interakcji z technologią cyfrową. Zaproponowana przeze mnie perspektywa badawcza skupia się na oprogramowaniu jako jednym z możliwych narzędzi testowych, analitycznych i poznawczych dla prowadzenia badań nad uchwyceniem realizacji zależności między cyfrowymi technologiami a kulturą. Takie podejście pozwala zejść na poziom programowania jako modelu obecnego w epistemicznym sprzężeniu zwrotnym między historią rozwoju technologii i współczesnymi jej wersjami, oraz pokazać, jak model ten potrafi przekształcać, kontrolować i zarządzać komunikacją oraz relacjami społecznymi i ekonomicznymi tych, którzy w nim funkcjonują. W samym wytwarzaniu oprogramowania dominującą rolę odgrywa modelowanie, a więc translacja procesów myślowych na sposób działania systemu. Na strategię rozwoju oprogramowania mają też znaczący wpływ badania nad sposobami, w jaki użytkownicy wchodzą w interakcję z interfejsami (*Human-Computer Interaction*). Z tego powodu trzeba zaznaczyć, że oprogramowanie musi być analizowane w kontekście społecznym. Nie funkcjonuje ono jako odseparowany fenomen wyłącznie technologiczny. Społeczne uwarunkowanie jest tak głęboko wpisane w strategię wytwarzania oprogramowania oraz w jego językowy charakter, że należałoby właściwie myśleć o nim jako o technologiczno-ludzkiej hybrydzie. Oprogramowanie spełnia bowiem swoją rolę tylko w kontekście interakcji z użytkownikiem. Z tego powodu w rozdziale czwartym zaproponowałam metaforę oprogramowania jako mechanicznego Turka, który

z pozoru wydaje się automatycznym, czysto mechanicznym urządzeniem, ale w środku skrywa jednak człowieka. To on właśnie jest zasadniczą, fundamentalną częścią tego układu, a który w ramach tej metafory ma symbolizować całokształt społecznych uwarunkowań oprogramowania, a także ukrytych strategii inżynierii społecznej, nie dających się łatwo postrzegać w codziennym jego użytkowaniu.

Po drugie, opierając się na refleksji przedstawionej w rozdziale czwartym i piątym, które skupiały się na strategiach kontroli użytkowników, ale też pracy i wytwarzania wartości w sieci, trzeba podkreślić, że tak zwana sieć Web 2.0 jest pozornie społecznościowym przedsięwzięciem, umożliwiającym użytkownikom swobodę twórczą i komunikacyjną, ale w istocie jest dobrze zakamuflowanym kapitalistycznym modelem ekonomicznym. W tym obszarze bezustannie ścierają się dwie siły, które można określić jako tendencja do kontroli wyrażana przez jednostki posiadające wpływy w sieci oraz tendencja do uzyskania wolności przez jej użytkowników. Nie można jednoznacznie usytuować komercyjnych gigantów po stronie kontroli a ruchy wolnościowe po stronie wolności. Pomimo tego że najbogatsi potentaci technologiczni, sprawują największą kontrolę w sieci internetowej, nie można zignorować faktu, że platformy takie jak Facebook, Google czy Amazon zrewolucjonizowały modele komunikacyjne i zapewniły dostęp do wiedzy, informacji i możliwości tworzenia wartości kulturowych dla swoich użytkowników. Tak samo z drugiej strony, niektóre ruchy wolnościowe powstające jako alternatywa dla obecnego systemu ekonomicznego, które przedstawiałam w rozdziale piątym, pomimo idealistycznych założeń często mimowolnie wspierają system kapitalistyczny lub realizują te same założenia. Paradoksem kondycji współczesnej sieci jest to, że tak samo pełna kontrola, jak i pełna wolność są niemożliwe do zrealizowania, ale też nie byłyby opłacalne, przestałyby utrzymywać dotychczasowy system. Nadmierna kontrola ze strony rządu czy firm, które posiadają największe wpływy w sieci ogranicza twórczość i aktywność użytkowników, która stała się współcześnie przedmiotem kapitalizacji w ramach modelu ekonomicznego opartego na wartości informacji i danych. W odniesieniu do rozważań zawartych w rozdziale czwartym i piątym należy podkreślić sposób, w jaki egzekwowana jest kontrola w sieci cyfrowej. Otóż działa ona dyskretnie i nie dąży do całkowitego ograniczenia użytkownika, ponieważ pozornie wolny użytkownik, twórczy i aktywny w ramach zaprojektowanych dla niego ograniczeń interfejsu, przynosi najwięcej zysku z perspektywy systemu.

Po trzecie, kluczowym wnioskiem dla prowadzonych przeze mnie rozważań jest też teza o tym, że sieci komputerowe i działanie systemów wyrażają relacje społeczne, a co za

tym idzie zmiany na poziomie technologicznym skutkują tworzeniem się nowych porządków społecznych i nowej rzeczywistości kulturowej. Z tej perspektywy oprogramowanie jest technologią dynamiczną, ciągle rozwijającą się. Tak samo, jak kultura cyfrowa, która jest efektem jego działania, podlega bezustannym przekształceniom. Innymi słowy, nawiązując do koncepcji efektu interfejsu Gallowaya, można zaryzykować stwierdzenie, że kultura jest dynamicznym interfejsem oprogramowania, w ciągłym statusie swojego wytwarzania. Z drugiej zaś strony należy zaznaczyć, jak ważne jest prowadzenie krytycznego dyskursu na temat technologii cyfrowych, ponieważ redefinicja strategii wytwarzania oprogramowania i sposobu jego oddziaływania na kulturę jest w stanie wytworzyć nowe porządki społeczne i nową rzeczywistość w niezwykle dynamicznych i rozwijających się formach.

Po czwarte, w ramach *software studies*, które wciąż pozostaje dynamicznie kształtującą się dyscypliną, daje się zaobserwować zwrot w kierunku praktykowania programowania jako metody badawczej. Znamiennym przejawem tej zmiany jest skład zespołu redaktorskiego drugiej serii wydawniczej *Software Studies*, który, w przeciwieństwie do pierwszej serii, nie zawiera samych teoretyków dyskursu nad oprogramowaniem. Nad wznowioną serią, której pierwszych publikacji można się spodziewać w ciągu najbliższych lat, pracować będą teoretycy oprogramowania Noah Wardrip-Fruin, Wendy Hui Kyong Chun oraz hakerka, artystka i badaczka Winnie Soon i artystka, designerka Jichen Zhu. Sylwetki tych badaczek i badacza przywoływałam już w pierwszym rozdziale. Redaktorzy, przy okazji ogłoszenia nowej serii wydawniczej, jasno wskazywali na to, że mają na celu zmierzenie się z „przepaścią pomiędzy praktyką a teorią dotyczącą oprogramowania”<sup>493</sup> Winnie Soon w tym kontekście mówiła, że:

Jest wielu makerów, koderów i designerów, ale w moim odczuciu wciąż brakuje powiązania między praktykami oprogramowania a dziedziną studiów nad oprogramowaniem. Jednym ze sposobów zniwelowania tego braku jest oduczenie się obecnych założeń i normatywnych praktyk w tworzeniu i rozwijaniu oprogramowania (...). Można zacząć od zabaw, eksperymentów, estetycznych i krytycznych dociekań, które będą nie tylko proponować funkcjonalne rozwiązania, ale także umożliwią wyłonienie się nowych form wiedzy.<sup>494</sup>

---

<sup>493</sup> W. Hui Kyong Chun, W. Soon, N. Wardrip-Fruin, J. Zhu. *Software Studies, Revisited. A Roundtable on the Software Studies Series at MIT Press*. „Computational Culture” 2022, nr 9, online: <http://computationalculture.net/software-studies-revisited/>. Tłumaczenie A.P.

<sup>494</sup> Tamże. Tłumaczenie A.P.

Z przywołanego fragmentu wypowiedzi wynika jasno, jaki kierunek przybiera obecnie rozwój badań nad oprogramowaniem. Można zaobserwować zwrot w kierunku praktycznego zdobywania wiedzy i wypracowywania nowych metod badawczych za pomocą narzędzi programistycznych. W tym też sensie zestawiam w pracy przykłady teorii i praktyk badawczo-artystycznych, pokazując, że te drugie współcześnie zyskują sprawczość krytyczną właśnie jako działania na konkretnym oprogramowaniu, z nim i przeciw niemu. To one tłumaczą poprzez kodowanie i projektowanie interakcji z odbiorcami, co kryje oprogramowanie, platformy cyfrowe i korporacje nimi zarządzające. Jest to wiedza praktyczna i niezbędna do poszerzenia akademickiego pola badań w *software studies*, i szerzej, w *media studies*.

Trzeba zaznaczyć, iż jest to pogłębiona kontynuacja założeń ukutych u początków powstania samej dyscypliny, a więc faktu, że *software studies* nigdy nie było wyłącznie teoretyczną dyscypliną. Od samego początku bowiem w ramach *software studies* starano się łączyć badaczy i badaczki zarówno prowadzących dyskurs teoretyczny, jak i stosujących praktykowanie programowania jako metodę poszukiwań. Widać to na przywoływanych już przeze mnie przykładach współpracy Geoffa Coxa z Alexem McLeanem nad publikacją *Speaking Code. Coding as Aesthetic and Political Expression* czy dorobku Aleksandra R. Gallowaya, który jest jednocześnie artystą, programistą i naukowcem. Doskonałym przykładem łączenia dyskursu teoretycznego i praktycznego jest Dmitri Kleiner, którego działalność intelektualną i praktyczną opisywałam w ostatnim rozdziale, a także, również przywoływany już w treści pracy, Vladan Joler, który z własnej praktyki programistycznej wywodzi wnioski badawcze dotyczące cyfrowej rzeczywistości. Ja również, co starałam się przedstawić w rozdziale drugim, wykorzystywałam praktykę programistyczną w celu naukowej eksploracji zagadnienia rekurencji oprogramowania i jego związku z językiem naturalnym.

Jednak, jak wynika z wypowiedzi redaktorów nowej serii wydawniczej, można się spodziewać, że to właśnie eksperymentalne praktyki programowania będą mieć kluczowe znaczenie w nadchodzących badaniach na temat oprogramowania. W dziedzinie *software studies* praktyka i teoria potrzebują siebie nawzajem i tylko właściwe ich połączenie pozwala wypracować metody badawcze umożliwiające zrozumienie dynamicznie rozwijających się metod działania oprogramowania w kulturze. W ramach nowego otwarcia potrzeba coraz więcej projektów opartych o praktyki software'owe, wytwarzania oporu i krytyki jako działania, co pozwoli zbliżyć się do samych funkcji oprogramowania jako systemu operacyjnego kultury, nie pozostając jedynie w ramach czysto akademickiego

dyskursu, który często ma tendencję do wpisywania nowych, bardzo dynamicznie rozwijających się zjawisk kultury cyfrowej w ugruntowane i ustabilizowane modele poznawcze oraz uznane i wypracowane w obszarze badań nad kulturą czy językiem metody i metodologie. Tego rodzaju przedsięwzięcia, które polegają na wytwarzaniu wiedzy poprzez działanie i aktywne poszukiwanie rozwiązań, często wywołują dyfrakcję i dysonans poznawczy, szczególnie gdy dotyczą zjawisk wielowymiarowych i transdyscyplinarnych, takich jak działanie oprogramowania w warstwach kultury. Dlatego też *software studies* nie może pozostać jedynie domeną badań interdyscyplinarnych, ale powinno otwierać się poza dyscypliny akademickie, w stronę praktyk koderskich, artystycznych czy aktywistyczno-badawczych.

Po szóste, co należy podkreślić raz jeszcze, praca pokazuje aspekt działania *software'u* w ramach kultur i społeczeństw, które są infrastrukturalnie wpisane w model cyfrowego i platformizacyjnego kapitalizmu. W tym sensie, jak starałam się wykazać, sytuuję się w opisie i analizie modelu zarządzanego przez kategorię, którą krytycy współczesnych metod wytwarzania i dystrybucji wiedzy określają bardzo dosadnie mianem *whiteness supremacy*.<sup>495</sup> W niniejszej dysertacji przedstawiałam więc znaczenie oprogramowania dla kultur i społeczeństw, których funkcjonowanie i rozwój ufundowany jest na modelach wytwarzanych przez grupy społeczne uprzywilejowane pod względem dostępu do wiedzy, umiejętności czy posiadanego kapitału. Opracowane w ten sposób strategie ekonomiczno-polityczne czerpią korzyści z przede wszystkim z asymetrii społecznych.

### **Dalsze perspektywy**

Pisząc tę pracę musiałam wybrać pewien ograniczony zakres tematyczny dotyczący procesów kulturowych, w którym oprogramowanie odgrywa znaczącą rolę. Zdaję sobie sprawę, że przedstawiona przeze mnie dysertacja nie wyczerpuje tematu kulturotwórczej roli oprogramowania.

Wśród dalszych kierunków, które można podjąć, aby kontynuować zapoczątkowaną przeze mnie refleksję należy przede wszystkim zauważyć, że w momencie, gdy kończę pracować nad niniejszą rozprawą, Meta (a więc Facebook i Instagram) przestaje być całkowicie darmową platformą i od listopada 2023 roku wprowadzona zostaje możliwość

---

<sup>495</sup> B. Zehner, *Machines of Subjection...*, dz. cyt., s. 49.

subskrypcji aplikacji, w celu dostępu do jej wersji bez reklam. Koszt takiej subskrypcji wynosi około dziesięć dolarów miesięcznie. Ta transformacja modelu ekonomicznego jest odpowiedzią na zaostrenie praw obrony danych osobowych, które miało miejsce w ostatnich latach (o czym pisałam w rozdziale czwartym). Wykorzystywanie danych biometrycznych do dostosowywania reklam przestaje być legalne w świetle europejskiego prawa, z tego powodu polityka ekonomiczna platform społecznościowych, opierających się na tego rodzaju strategiach, będzie musiała ulec zmianie.<sup>496</sup> Dochodzi tu więc do skapitalizowania prywatności poprzez wyznaczenie jej ceny, przemienienia jej w dobro luksusowe, na które nie każdy będzie mógł sobie pozwolić. Należy przy tym podkreślić, że ze strony Meta nie udało się uzyskać żadnej deklaracji dotyczącej tego, że dane osób posiadających płatną subskrypcję nie będą wykorzystywane przez firmę.<sup>497</sup> Monitorowanie dalszych przemian i ich konsekwencji stanowi wyzwanie, które wolny rynek stawia przed współczesną nauką i osobami, które próbują poddawać krytycznej refleksji strategii inżynierii społecznej.

Kolejną istotną kwestią, którą można by dalej rozwijać jest zagadnienie interakcji na poziomie człowiek-komputer. Jak już wskazywałam, jest ono kluczowe dla zrozumienia procesów wytwarzania i interakcji z technologią cyfrową. W ramach badań na temat HCI można by podjąć się dogłębnej analizy strategii projektowania interakcji człowieka z komputerem poprzez kreowanie interfejsów, filozoficznych i ideologicznych proveniencji tego procesu oraz tego, jak ewoluował on wraz z rozwojem technologii cyfrowych. Aktualne badania dotyczące metod projektowania cyfrowych interfejsów są bardzo potrzebne i niezbędne do prowadzenia krytycznego dyskursu dotyczącego kultury cyfrowej, szczególnie, że formy interfejsu dynamicznie zmieniają się wraz z rozwojem oprogramowania. Podkreślali to już Christian Ulrik Andersen wraz z Sørenem Bro Poldem, a więc autorzy książki *The Metainterface. The Art of Platform, Cities, and Clouds*, którzy wskazywali na to, że przez technologie chmurowe, usługi sieciowe, transfer danych współczesne interfejsy rozmywają się, stają się coraz mniej uchwytnie, wtopione w kulturową warstwę komunikacji społecznej.<sup>498</sup> Badacze argumentują jednak, że pomimo prób sprawienia, by interfejs zniknął w płynnej, cyfrowej rzeczywistości, powraca on jako wyparty element kultury w postaci metainterfejsu. Praktyki artystyczne są dla autorów

---

<sup>496</sup> M. Meaker, *Facebook Finally Puts a Price on Privacy: It's \$10 a Month*, 1.11.2023, online: <https://www.wired.com/story/meta-facebook-pay-for-privacy-europe>, dostęp 27.11.2023.

<sup>497</sup> Tamże.

<sup>498</sup> C.U. Andersen, S.B. Pold, *The Metainterface. The Art of Platform, Cities, and Clouds*, The MIT Press, Cambridge, Mass. 2018.

punktem wyjścia do analizy tego, jak współcześnie doświadczają się interfejsów, za pomocą jakich procedur i retoryk się wyrażają.

Rozważania dotyczące ruchów związanych z wolnym oprogramowaniem i przeciwstawieniem się zastanym strukturom sieci warto poszerzyć o kontekst kultury hakerskiej. Inicjatywy hakerskie cały czas rozbrajają wszechobecną dominantę cyfrowych gigantów kapitalistycznych.<sup>499</sup> Współcześnie domaga się analiz kwestia tego, jak zmienia się obecnie rola hakera, szczególnie że coraz częściej dochodzi do prywatyzacji działań hakerskich, zmienianie działalności społecznej na usługi, poprzez opłacanie przez podmioty rządowe hakerów na przykład w celach przeprowadzania operacji wojennych.<sup>500</sup> Tematem wartym kontynuacji w kontekście poszukiwania alternatyw wobec systemu kapitalistycznego jest również kultura makerska zorganizowana wokół idei *do-it-yourself*, która dostarcza użytkownikom narzędzi do tego, aby samodzielnie tworzyli narzędzia cyfrowe, umożliwiające prowadzenie własnego dyskursu w stechnicyzowanej kulturze.<sup>501</sup> Tego rodzaju inicjatywy z pewnością dostarczają nowych rozwiązań technologicznych i strategii dystrybucji oprogramowania, a więc, zgodnie z wcześniejszymi ustaleniami, przyczyniają się one do tworzenia nowych relacji społecznych, co z pewnością jest warte analizy naukowej.<sup>502</sup>

Problematyką wartą kontynuacji również jest kwestia ujawniania się oprogramowania poza interfejsem. Przykładem takiego manifestowania się oprogramowania w kulturze są praktyki związane z ruchem *live-coding*, których wartość pozostaje nieoceniona przez wzgląd na ich szczególny charakter związany z uwidacznianiem kodu.<sup>503</sup> Refleksja nad ujawnianiem się oprogramowania w kulturze może również dotyczyć zjawisk związanych z błędami. Kiedy dochodzi do usterki i system przestaje funkcjonować tak, jak powinien, jest to moment niespodziewanej manifestacji oprogramowania spod warstwy interfejsu, która w innych okolicznościach pozostawałaby ukryta. W tym kontekście jest to też bardzo istotny moment załamania się modelu ideologicznego, ukrytego w działaniu systemu, co już

---

<sup>499</sup> M. Webb, *Coding Democracy. How Hackers Are Disrupting Power, Surveillance, and Authoritarianism*, The MIT Press, Cambridge, Mass. 2021; J. Söderberg, Maxigas, *Resistance to the Current. The Dialectics of Hacking*, The MIT Press, Cambridge, Mass. 2022.

<sup>500</sup> Więcej na ten temat zob. L. Follis, A. Fish, *Hacker States*, The MIT Press, Cambridge, Mass. 2020.

<sup>501</sup> O kulturze makerów i jej rewolucyjnym działaniu pisał już Chris Anderson w książce *C. Anderson, Makers...*, dz. cyt.

<sup>502</sup> Wgląd w teorię *do-it-yourself* w obszarze artystycznych poszukiwań opartych na tworzeniu nowych technologii w celu prowadzenia badań i krytycznych poszukiwań zawiera publikacja H. Garnet, *Art + DIY Electronics*, The MIT Press, Cambridge, Mass. 2023.

<sup>503</sup> W tym kontekście cenną i wydaną niedawno publikacją jest A.F. Blackwell, E. Cocker, G. Cox, A. McLean, T. Magnusson, *Live Coding...*, dz. cyt., która jest pierwszym tego rodzaju zebraniem i uporządkowaniem dotychczasowych praktyk i kierunków rozwoju programowania na żywo.

zostaje zauważone w dyskursie badawczym skupionym na cyfrowej kulturze, czego przykładem jest zapowiedziana na rok 2024 publikacja *More than a Glitch. Confronting Race, Gender, and Ability Bias in Tech*, autorstwa Meredith Broussard.<sup>504</sup> W nadchodzącej książce autorka ma przedstawiać to, że rasizm, seksizm i inne uprzedzenia nie są zwyczajnymi błędami działania systemu, ale manifestacjami kryjącego się pod powierzchnią interfejsu oprogramowania i danych.

Ostatnią, ważną kwestią, wyraźnie odznaczającą się w mojej pracy i domagającą się kontynuacji, jest kategoria prawdy, która stała się bardzo problematycznym pojęciem od czasu skandalu związanego z Cambridge Analytica. Wiarygodność informacji w sieci spadła, przez co wzrosła potrzeba opracowywania strategii weryfikacji rzetelności informacji oraz głębokiej problematyzacji zjawiska fałszywych informacji i społecznej inżynierii za pomocą konstruowania post-prawdy.<sup>505</sup> Co więcej, na podstawie moich rozważań dotyczących wytwarzania modeli, odgrywających kluczową rolę w tworzeniu oprogramowania, daje się dostrzec, że naukowe strategie wytwarzania wiedzy i ustalania „prawd” również wymagają krytyki i redefinicji. Sean Cubitt, jeden z pionierów *media studies*, wskazywał na to, że od lat 40. XX wieku cybernetyka zdominowała ideologicznie strategie wytwarzania wiedzy.<sup>506</sup> W opublikowanej niedawno książce *Truth. Aesthetic Politics*, badacz nawołuje o poszukiwanie alternatywnych form utrwalania prawd dotyczących rzeczywistości i sugeruje, aby poszukiwać ich w obszarach, które były wykluczone przed dotychczasowe modele naukowe.

Kierunków kontynuowania refleksji nad znaczeniem oprogramowania w kulturze można wyznaczyć jeszcze więcej, ale przede wszystkim należy podkreślić, że każda próba prowadzenia krytycznego dyskursu na temat kultury cyfrowej i roli, jaką odgrywa w niej oprogramowanie jest ważną inicjatywą, potrzebną dla zrozumienia współczesnej rzeczywistości. Praca niniejsza jest też poniekąd zapisem ograniczonym czasem prowadzenia przeze mnie badań w latach 2017-2023, dotyczących działania oprogramowania jako systemu operacyjnego kultury. Jest to też jedna z form zapisu i swoistej archiwizacji tego, co w różnych punktach, miejscach, dyskursach i praktykach w globalnym i wielowymiarowym aspekcie działania technologii cyfrowych wówczas zaszło. Jednak procesy działania oprogramowania w sprzężeniu zwrotnym z kulturą

---

<sup>504</sup> M. Broussard, *More than a Glitch. Confronting Race, Gender, and Ability Bias in Tech*, The MIT Press, Cambridge, Mass. 2024.

<sup>505</sup> Dowodem na potrzebę opracowań dotyczących tego zjawiska kulturowego jest chociażby publikacja: *Fake News, Understanding Media and Misinformation in the Digital Age*, red. M. Zimdars, K. McLeod, The MIT Press, Cambridge, Mass. 2020.

<sup>506</sup> S. Cubitt, *Truth. Aesthetic Politics*, Goldsmiths Press, Cambridge, Mass., London 2023, EPUB, rozdz. 1.

współczesną nie kończą się, ale coraz wyraźniej i szybciej rozwijają, zmieniają i przekształcają. W ostatnim czasie do przyspieszenia rozwoju technologicznego i poszukiwania nowych rozwiązań technologicznych znacznie przyczyniły się pandemia wirusa COVID-19, trwająca od 2019 do 2022 roku, oraz inwazja Rosji na Ukrainę, która nastąpiła 24 lutego 2022 roku. Prowadzenie badań i praca nad stworzeniem tekstu naukowego wobec tak szybko przekształcającej się rzeczywistości stanowi swego rodzaju wyzwanie, ponieważ należy zmierzyć się z faktem, że rozwój technologiczny postępuje znacznie szybciej od rozwoju nauki zajmującej się jego krytyką. Z tego powodu tak istotne jest wytwarzanie krytycznego dyskursu dotyczącego współczesnej rzeczywistości cyfrowej nie tylko akademickiego, ale również w postaci eksperymentalnych poszukiwań wytwarzania wiedzy w praktyce w postaci projektów artystyczno-badawczych, hakerskich i koderskich oddolnych przedsięwzięć.

## Bibliografia

### Literatura:

- Adorno Theodor W., *Przemysł kulturalny. Wybrane eseje o kulturze masowej*, tłum. M. Bucholc. Warszawa, Narodowe Centrum Kultury 2019.
- Alashbi Abdulaziz Ali Saleh, Sunar Mohd Shahrizal, *Occluded Face Detection, Face in Niqab Dataset w: Emerging Trends in Intelligent Computing and Informatics*, IRICT, Malaysia 2019, s. 209-215.
- Allen G. E., *Mechanism, Vitalism and Organicism in Late Nineteenth and Twentieth Century Biology. The Importance of Historical Context*, „Studies in History and Philosophy of Biological and Biomedical Sciences” 36 (2005) nr 2, s. 261-283.
- Althusser Louis, *Ideological State Apparatuses*, w: *Mapping Ideology*, red. S. Žižek, Verso, Londyn, Nowy Jork 1994, s. 100-140.
- Andersen Christian Ulrik, Pold Søren Bro, *The Metainterface. The Art of Platform, Cities, and Clouds*, The MIT Press, Cambridge, Mass. 2018.
- Anderson Chris, *Makers. The New Industrial Revolution*, Crown Business, Nowy Jork 2012.
- Andrejevic Mark, *Estranged Free Labor*, w: *Digital Labor. The Internet as a Playground and Factory*. red. T. Scholz, Taylor & Francis, Nowy Jork 2013, s. 192-212.
- Argasiński Jan K., *Software Studies – co to takiego?*, rozm. przepr. A. Marjankowska, E. Drygalska, 10.2016, online: <https://www.miesiecznik.znak.com.pl/software-studies-co-to-takiego>.
- Argasiński Jan K., *Software Studies. Kultura, sztuka i oprogramowanie*, 2014, online: [https://ruj.uj.edu.pl/xmlui/bitstream/handle/item/41743/Argasinski\\_Software\\_studies\\_Kultura\\_sztuka\\_i\\_oprogramowanie\\_2014.pdf?sequence=1&isAllowed=y](https://ruj.uj.edu.pl/xmlui/bitstream/handle/item/41743/Argasinski_Software_studies_Kultura_sztuka_i_oprogramowanie_2014.pdf?sequence=1&isAllowed=y).
- Austin John L., *Jak działać słowami*, w: *Tenże, Mówienie i poznawanie. Rozprawy i wykłady filozoficzne*, tłum. B. Chwedeńczuk, J. Woleński, Wydawnictwo Naukowe PWN, Warszawa 1993.
- Baeza Cristóbal, *The degendering of voice assistants, Artificial Intelligence*, BBVA, 28.08.2020, online: <https://www.bbva.com/en/the-degendering-of-voice-assistants>.
- Bansal Gagan, Nushi Besmira, Kamar Ece, Lasecki Walter S., Weld Daniel S., Hirvitz Eric, *Beyond Accuracy. The Role of Mental Models in Human-AI Team Performance*,

- „National Conference on Artificial Intelligence” 2019; online: <http://erichorvitz.com/gbansal-hcomp19.pdf>.
- Banka dotcomów i jej wpływ na rynek*, 12.09.2019, online: <https://histmag.org/Banka-dotcomow-i-jej-wplyw-na-rynek-19351>.
- Baudrillard Jean, *The Ecstasy of Communication*, tłum. B. Schütze, C. Schütze, Semiotext(e), Brooklyn 1988.
- Beck Ulrich, *Spółczesność Ryzyka. W drodze do innej nowoczesności*, tłum. S. Cieśla, Wydawnictwo Naukowe Scholar, Warszawa 2002.
- Bendyk Edwin, *Antymatrix. Człowiek w labiryncie sieci*, Wydawnictwo W.A.B., Warszawa 2004.
- Bernal Paul, *Internet Privacy Rights. Rights to Protect Autonomy*, Cambridge University Press, Cambridge 2014.
- Biriukow Boris W., Geller Jefim S., *Cybernetyka w naukach humanistycznych*, tłum. J. Sarna, Wydawnictwo Ossolineum, Warszawa, Wrocław, Kraków, Gdańsk, Łódź 1983.
- Blackwell Alan F., Cocker Emma, Cox Geoff, McLean Alex, Magnusson Thor, *Live Coding: A User's Manual*, The MIT Press, Cambridge, Mass. 2022.
- Bogost Ian, *Hyperemployment, or the Exhausting Work of the Technology User*, 08.11.2013, online: <https://www.theatlantic.com/technology/archive/2013/11/hyperemployment-or-the-exhausting-work-of-the-technology-user/281149>.
- Bogost Ian, *Persuasive Games. The Expressive Power of Videogames*, The MIT Press, Cambridge, Mass. 2007.
- Bratton Benjamin H., *The Stack: On Software and Sovereignty*, The MIT Press, Cambridge, Mass. 2016.
- Brecht Bertolt, *The radio as an apparatus of communication*, w: *New Media. Theories and Practices of Digitextuality*, red. A. Everett, J. T. Caldwell, Nowy Jork 2003, s. 29-32.
- Briant Emma L., *Propaganda and Counter-Terrorism: Strategies for Global Change*, Manchester University Press, Manchester 2015.
- Briant Emma L., *Propaganda Machine: Inside Cambridge Analytica and the Digital Influence Industry*, 2020, online: <https://www.propagandamachine.tech/ca-map>
- Briant Emma L., *Transcripts: Oakes, Wigmore, Patten and Gunster Submitted to DCMS Inquiry into Disinformation and Fake News*, 2018, online:

- <https://www.parliament.uk/documents/commons-committees/culture-media-and-sport/Dr-Emma-Briant-Audio-File-Transcripts-with-links.pdf>
- Broussard Meredith, *Artificial Unintelligence*, MIT Press, Cambridge, Mass. 2018.
- Cadwalladr Carole, *'I made Steve Bannon's psychological warfare tool': meet the data war whistleblower*, 18.03.2018, online: <https://www.theguardian.com/news/2018/mar/17/data-war-whistleblower-christopher-wylie-faceook-nix-bannon-trump>.
- Calhoun Craig, *The Question of Class Struggle. Social Foundations of Popular Radicalism during the Industrial Revolution*, The University of Chicago Press, Chicago 1952.
- Casilli Antonio A., *No CAPTCHA: Yet Another Ruse Devised by Google to Extract Free Digital Labor From You*, 05.12.2014, online: <https://www.casilli.fr/2014/12/05/no-captcha-is-google-jargon-for-mechanical-turk-for-free>.
- Cassimatis Nicholas L., *Artificial Intelligence and Cognitive Modeling Have the Same Problem*, w: *Theoretical Foundations of Artificial General Intelligence*, red. P. Wang, B. Goertzel, Atlantis Press, Amsterdam 2012, s. 11-24.
- Castells Manuel, *Communication Power*, Oxford University Press, Oxford 2009.
- Castells Manuel, Himanem Pekka, *Spółeczeństwo informacyjne i państwo dobrobytu. Model fiński*, tłum. M. Sutkowski, M. Penkala, Wydawnictwo Krytyki Politycznej, Warszawa 2009.
- Celiński Piotr, *Biomedialne dane w rękach władzy politycznej. Aplikacje, Bazy danych i biometria*, „Kultura Współczesna” 2019, nr 1(104), s. 15-23.
- Celiński Piotr, *Postmedia. Cyfrowy kod i bazy danych*, Wydawnictwo Uniwersytetu Marii Curie-Skłodowskiej, Lublin 2013.
- Certyfikowany tester. Sylabus poziomu podstawowego ISTQB. Wersja 3.1*, tłum. L. Stapp, Stowarzyszenie Jakości Systemów Informatycznych 2018.
- Chomsky Noam, *Syntactic Structures*, Mouton de Gruyter, Berlin, Nowy Jork 2002.
- Chun Wendy Hui Kyong, *Control and Freedom: Power and Paranoia in the Age of Fiber Optics*, The MIT Press, Cambridge, Mass. 2006.
- Chun Wendy Hui Kyong, *Discriminating Data: Correlation, Neighborhoods, and the New Politics of Recognition*, The MIT Press, Cambridge, Mass. 2021.
- Chun Wendy Hui Kyong, *On Software, or the Persistence of Visual Knowledge*, „Grey Room” 18, 2004, s. 26-51.

- Chun Wendy Hui Kyong, *Programmed Visions: Software and Memory*, The MIT Press, Cambridge, Mass. 2011.
- Chun Wendy Hui Kyong, Soon, Winnie, Wardrip-Fruin, Noah, Zhu, Jichen. *Software Studies, Revisited. A Roundtable on the Software Studies Series at MIT Press*. „Computational Culture” 2022, nr 9, online: <http://computationalculture.net/software-studies-revisited>.
- Cockburn Alistair, *Using natural Language as a Metaphoric Base for OO*, „ACM SIGPLAN OOPS Messenger” 1993, nr 4(2), s. 187-189.
- Cox Geoff, McLean Alex, *Speaking Code. Coding as Aesthetic and Political Expression*, The MIT Press, Cambridge 2013.
- G. Cox, *Virtual Suicide as Decisive Political Act*, w: *Activist Media and Biopolitics*, red. W. Sützl, T. Hug, University of Innsbruck Press, Innsbruck 2011.
- Cybernetyka*, w: *Uniwersalny Słownik Języka Polskiego*, red., S. Dubisz, Wydawnictwo Naukowe PWN, Warszawa 2008, s. 489.
- Çıdam Çiğdem, *A politics of love? Antonio Negri on revolution and democracy*, „Contemporary Political Theory” 2013, nr 12, s. 26-45.
- DaBlade, *Richard Stallman Interview*, 06.02.2006, online: <http://www.p2pnet.net/story/7840>.
- Darczewska Jolanta, *Diabeł tkwi w szczegółach. Wojna informacyjna w świetle doktryny wojennej Rosji*, Ośrodek Studiów Wschodnich im. Marka Karpia, Warszawa 2015.
- Descartes René, *Medytacje o pierwszej filozofii*, tłum. M. i K. Ajdukiewiczowie, S. Swieżawski, I. Dąmbaska, Wydawnictwo ANTYK, Kęty 2001, s. 49-50.
- Dillon Sarah, *The Eliza effect and its dangers: from demystification to gender critique*, „Journal for Cultural Research” 2020, t. 24, nr 1, s. 1-15.
- Dijkstra Edsger W., *A Review of the 1977 Turing Award Lecture by John Backus*, 23.07.2006, online: <http://www.cs.utexas.edu/users/EWD/transcriptions/EWD06xx/EWD692.html>.
- Dury Horace B., *Scientific management; a history and criticism*, Nowy Jork, Columbia University 1919, s. 195-198, online: <https://archive.org/details/cu31924002406647/page/n23/mode/2up>.
- Dym Clive, *Principles of Mathematical Modeling*, Academic Press, Nowy Jork 2004.
- Embury-Dennis Tom, *Man who invented 'Like' button deletes Facebook app over addiction fears*, 06.10.2017, online:

- <https://www.independent.co.uk/life-style/gadgets-and-tech/facebook-inventor-deletes-app-iphone-justin-rosenstein-addiction-fears-a7986566.html>.
- Engelman Ryan, *The Second Industrial Revolution, 1870-1914*, online: <https://ushistoryscene.com/article/second-industrial-revolution>.
- Enzensberger Hans M., *Constituents of a Theory of the Media*, w: *Video culture. A critical investigation*, red. J. Hanhardt, Visual Studies Workshop Press, Nowy Jork 1986, s. 96-123.
- Europa naprawia Internet. Czy to koniec dyktatu cyberkorporacji?*, 27.07.2021, online: <https://panoptykon.org/jak-dsa-zmieni-internet>.
- Facebook Algorithmic Factory*. Katarzyna Szymielewicz, Vladan Joler, Paweł Janicki, 18.05.2017, online: <https://wro2017.wrocenter.pl/facebook-algorithmic-factory>.
- Facebook CEO Mark Zuckerberg testifies before Congress on data scandal*, 2018, online: <https://www.youtube.com/watch?v=u-FIWZ1BOcA>.
- Facebook usuwa hasło o tym, że zawsze będzie darmowy. Nadchodzi płatna wersja serwisu?*, 08.09.2019, online: <https://www.wirtualnemedial.pl/artykul/facebook-usuwa-haslo-o-tym-ze-zawsze-bedzie-darmowy-nadchodzi-platna-wersja-serwisu>.
- Faster Than Thought. A Symposium on Digital Computing Machines*, red. B.V. Bowden, Sir Isaac Pitman & Sons, London 1953.
- Felczak Mateusz, *Obiekt i kod. Badanie gier wideo w perspektywie software studies oraz realizmu spekulatywnego*, „Przegląd Kulturoznawczy” 2016, nr 3(29), s. 305-317.
- Filiciak Mirosław, *Zajrzeć pod powierzchnię ekranu*, „Kultura Popularna” 2008, nr 4(22), s. 15-18.
- Fisk Peter, *Sophia, the intelligent humanoid robot... built on the traits of creativity, empathy and compassion*, 27.10.2017, online: <https://www.peterfisk.com/2017/10/sophia-intelligent-humanoid-robot-built-traits-creativity-empathy-compassion>.
- Foucault Michel, *O archeologii nauk. W odpowiedzi Kołu epistemologicznemu*, tłum. A. Bandura, w: *Język, dyskurs, społeczeństwo*, red. L. Rasiński, Wydawnictwo Naukowe PWN SA, Warszawa 2009, s. 307-332.
- Free Software, Free Society, selected essays of Richard Stallman*, red. J. Gay, Free Software Foundation, Boston, Mass. 2002.
- French Robert M., *Subcogniton and the Limits of the Turing Test*, „Mind” 1990, nr 99(393), 53-65.

- Freud Zygmunt, *Niesamowite*, w: tegoż, *Pisma psychologiczne*, tłum. R. Reszke, Wydawnictwo KR, Warszawa 1997, s 233-262.
- Freud Zygmunt, *Poza zasadą przyjemności*, tłum. J. Prokopiuk, Wydawnictwo Naukowe PWN, Warszawa 1976.
- Freud Zygmunt, *Psychologia nieświadomości*, tłum. R. Reszke, Wydawnictwo KR, Warszawa 2010.
- Freud Zygmunt, *Psychopatologia życia codziennego. Marzenia senne*, tłum. L. Jekels, H. Inka, W. Szewczuk, Wydawnictwo PWN, Warszawa 1997.
- Fuller Matthew, *Behind The Blip. Essays on Culture of Software*, Autonomedia, Nowy Jork 2003.
- Fuller Matthew, *Software Studies Methods*, w: *The Routledge Companion to Media Studies and Digital Humanities*, red. J. Sayers, Routledge, Nowy Jork 2018, s. 250-257.
- Galanter Philip, *What is generative art? Complexity Theory as a Context for Art Theory*, 28.12.2007, online: [http://www.philipgalanter.com/downloads/ga2003\\_paper.pdf](http://www.philipgalanter.com/downloads/ga2003_paper.pdf).
- Garber Megan, *Why We Prefer Masculine Voices (Even in Women)*, 18.12.2012, online: <https://www.theatlantic.com/sexes/archive/2012/12/why-we-prefer-masculine-voices-even-in-women/266350>.
- Garnet Hertz, *Art + DIY Electronics*, The MIT Press, Cambridge, Mass. 2023.
- Goldsmith Jack, Wu Tim, *Who Controls the Internet? Illusions of Borderless World*, Oxford University Press, Nowy Jork 2006.
- Galloway Alexander R., *Język chce, by go nie dostrzegać*, tłum. M. Szota, „Kultura Popularna” 2008, nr 4(22), s. 23-36.
- Galloway Alexander R., *Protocol: how control exists after decentralization*, MIT Press Cambridge, Mass., 2004.
- Galloway Alexander R., *Ręce ubrudzone kodem*, rozm. przepr. M. Halawa, „Kultura Popularna” 2008, nr 4(22) s. 19-22.
- Galloway Alexander R., *The Interface Effect*, Polity Press, Cambridge/Malden 2012.
- Galloway Alexander R., Thacker Eugene, *The Exploit. A Theory of the Networks*, University of Minnesota Press, Minneapolis 2007.
- Ganti Akhilesh, *Venture Capitalists Definition: Who Are They and What Do They Do?*, 2022, online: <https://www.investopedia.com/terms/v/venturecapitalist.asp>
- Głowacka Dorota, Szymielewicz Katarzyna, *DSA i DMA w pigułce – co warto wiedzieć o prawie, które ma poskromić cybergigantów*, 03.06.2022, online: <https://panoptykon.org/akt-o-uslugach-cyfrowych-dsa>.

- Guattari Félix, *Chaosophy. Texts and interviews 1972-1977*, red. S. Lotringer, MIT Press, Cambridge, Mass. 2009.
- Harman Chris, *Kapitalizm zombie. Globalny kryzys i aktualność myśli Marksa*, tłum. H. Jankowska, Warszawskie Wydawnictwo Literackie Muza, Warszawa, 2011.
- Hartmann Jan, Angeli Antonella De, Sutcliffe Alistair, *Framing the User Experience. Information Biases on Website Quality Judgement*, „Proceeding of the Twenty-Sixth Annual CHI Conference on Human Factors in Computing Systems” 2008, s. 855-864.
- Havel John, *reCAPTCHA: The Genius Who's Tricking the World Into Doing His Work*, 3.12.2023, online: <https://thehustle.co/the-genius-whos-tricking-the-world-into-doing-his-work-recaptcha>
- Historyjka użytkownika*, w: *Słownik Terminów Testowych ISTQB. Wersja 3.4*, tłum. L. Stapp, Stowarzyszenie Jakości Systemów Informatycznych 2020, s. 30.
- Hofstadter Douglas R., *Fluid Concepts & Creative Analogies. Computer Models of the Fundamental Mechanism of Thought*, Basic Books, Nowy Jork 1995.
- Hofstadter Douglas R., *Gödel, Escher, Bach: An Eternal Golden Braid*, Penguin, London 1994.
- Horkheimer Max, Adorno Theodor W., *Dialektyka oświecenia. Fragmenty filozoficzne*, tłum. M. Łukasiewicz, Wydawnictwo Instytutu Filozofii i Socjologii PAN, Warszawa 1994.
- Huberman Bernardo A., *Social Computing and the Attention Economy*, „Journal of Statistical Physics” 2013, nr 1(151), s. 329-339.
- Huhtamo Erkki, *Z ust lalki, niejasno. O sztuce Kena Feingolda*, w: *Ken Feingold: Figury mowy*, red. R. W. Kluszczyński, Centrum Sztuki Współczesnej Łaźnia, Gdańsk 2014, s. 36-75.
- Ilnicki Rafał, *Od programu komputerowego do kultury jako programu. Perspektywa studiów nad oprogramowaniem (software studies)*, „Media, Kultura, Społeczeństwo” 2011, nr 1(6), s. 27-39.
- Inequality Kills: The unparalleled action needed to combat unprecedented inequality in the wake of COVID-19*, Oxfam International, Oxford 2022.
- Jakesch Maurice, French Megan, Ma Xiao, Hancock Jeffrey T., Naaman Mor, *AI-Mediated Communication: How the Perception that Profile Text was Written by AI Affects Trustworthiness*, „Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems Conference” 2019, nr 239, s. 1-13.

- Janicki Paweł, *Efekt widzialności*, rozm. przepr. Michał Szota, „Kultura Popularna” 2008, nr 4(22), s.41-43.
- Jelevska Agnieszka, *Ekotopie. Ekspansja technokultury*, Wydawnictwo Naukowe UAM, Poznań 2014.
- Jelevska Agnieszka, Michał Krawczak, *Trudne relacji sztuki, nauki i technologii w Polsce*, w: *Sztuka i technologia w Polsce. Od cyberkomunizmu do zwrotu cyfrowego*, Wydawnictwo Naukowe UAM, Poznań 2014.
- Jemielniak Dariusz, Przeglasińska Aleksandra, *Collaborative Society*, The MIT Press, Cambridge, Mass. 2020.
- Językowy obraz świata, red. J. Bartmiński, Uniwersytet Marii Curie-Skłodowskiej, Lublin 1999.
- Joler Vladan, Petrovski Andrej, *Immaterial Labour and Data Harvesting. Facebook Algorithmic Factory*, 21.08.2016, online: <https://labs.rs/en/facebook-algorithmic-factory-immaterial-labour-and-data-harvesting>.
- Kant, w: Władysław Tatarkiewicz: *Historia Filozofii*. T. 2. Warszawa: PWN, 1988, s. 161–184.
- Ken Feingold: *Figury mowy*, red. R. W. Kluszczyński, Centrum Sztuki Współczesnej Łaźnia, Gdańsk 2014.
- Khadpe Pranav, Krishna Ranjay, Fei-Fei Li, Hancock Jeffrey T., Bernstein Michael S., *Conceptual Metaphors Impact Perceptions of Human-AI Collaboration*, „Proceeding ACM Human-Computer Interaction”, tom 4, nr CSCW2, artykuł 163, s. 1-26.
- Khashman Adnan, *Modeling cognitive and emotional processes: A novel neural network architecture*, „Neural Networks” 2010, nr 23(10), s. 1155-1163.
- Kirk Robert, *Mechanism*, w: *The Oxford Companion to Philosophy*, red. T. Honderich, Oxford University Press, Oxford 2005, s. 543.
- Kitchin Rob, Dodge Martin, *Code/Space. Software and Everyday Life*, The MIT Press, Cambridge, Mass. 2011.
- Kittler Friedrich, *There is No Software*, w: *Literature, Media, Information Systems: Essays*. red. J. Johnston, Overseas Publishers Association, Amsterdam 1997, s.147-155.
- Dmytri Kleiner, *Nie istnieje coś takiego, jak zewnątrz kapitalizmu*, rozm. przepr. E. Drygalska, 19.10.2015, online: <https://popmoderna.pl/nie-istnieje-cos-takiego-jak-zewnetrze-kapitalizmu-wywiad-z-dmytri-kleinerem/#>.

- Kleiner Dmitri, *The Telekommunist Manifesto*, Institute of Network Cultures, Amsterdam 2010.
- Kobierzycki Tadeusz, *Filozofia osobowości*, Eneteia, Warszawa 2001.
- Konkol Sebastian, *Algorytmy rządzą światem?*, 24.10.2016, online: <https://blog.twistersolve.pl/2016/10/algorytmy-rzadza-swiatem>.
- Koss Adele Mildred, *Programming on the Univac I: A woman's account*. „IEEE Annals of the History of Computing” 2003, nr 25(1), s. 48-59.
- Krawczak Michał, *Wetware*, w: *Mały słownik sztuki technologicznej*, online: <https://culture.pl/pl/artykul/maly-sloownik-sztuki-technologicznej>.
- La Mettrie, J. O. de, *Człowiek – maszyna*, tłum. S. Rudniański, Warszawa 1984.
- Landauer Gustav, *Revolution and Other Writings: A Political Reader*, red. tłum. G. Kuhn, PM Press, Oakland 2010.
- Landauer Thomas K., *Behavioral Research Methods in Human-Computer Interaction*, w: *Handbook of Human-Computer Interaction*, red. M.G. Helander, T.K. Landauer, P.V. Prabhu, North Holland, Nowy Jork 1997, s. 203-227.
- Lanier Jaron, *Ten Arguments for Deleting Your Social Media Accounts*, Vintage, Londyn 2019.
- Latil Pierre de, *Thinking by Machine. A Study of Cybernetics*, The Riverside Press, Cambridge, Mass. 1957.
- Latour Bruno, *Science In Action. How to Follow Scientists and Engineers Through Society*, Harvard University Press, Cambridge, Mass. 2015.
- Latour Bruno, *Pandora's Hope: Essays on the Reality of Science Studies*, Harvard University Press, Cambridge, Mass., Londyn 1999.
- Lazzarato Maurizio, *Immaterial Labor*, w: *Radical thought in Italy. A Potential Politics*, red. P. Virno, M. Hardt, University of Minnesota Press, Minneapolis, Londyn 2006, s. 132-146.
- Le Bon, Gustave, *Psychologia tłumy: studium powszechnego umysłu*, tłum. C. Matkowski, Wydawnictwo Helion, Gliwice 2012.
- Levy Pierre, *Cyberculture*, University Minnesota Press, Minneapolis 2001.
- Lightfoot David W., *Introduction*, w: Noam Chomsky, *Syntactic Structures*, Mouton de Gruyter, Berlin, Nowy Jork 2002, s. V-XVIII.
- Lovink Geert, *Stuck on the Platform. Reclaiming the Internet*, Valiz, Amsterdam 2012.

- Lovink Geert, *We must reclaim the internet, before it is too late*, 12.10.2022, online: <https://www.uva.nl/en/shared-content/faculteiten/en/faculteit-der-geesteswetenschappen/news/2022/10/we-must-reclaim-the-internet-before-it-is-too-late.html>.
- Łupkowski Paweł, *Test Turinga. Perspektywa sędziego*, Wydawnictwo Naukowe UAM, Poznań 2010.
- McIntyre Lee, *Post-Truth*, The MIT Press, Cambridge, Mass. 2018.
- Mackenzie Adrian, *Cutting code: Software and Sociality*, Peter Lang, Nowy Jork 2015.
- Malicious Life Podcast: Tay: A Teenage Bot Gone Rogue*, 06.12.2021, online: <https://www.cybereason.com/blog/malicious-life-podcast-tay-a-teenage-bot-gone-rogue>.
- Malinowska Anna, *Obiekty i technofeelie*, „Teksty Drugie” 2019, nr 5, s.18-34.
- Malinowska Anna, Miller Toby, *Sensitive Media*, „Open Cultural Studies” 2017, nr 1, s. 660-665.
- Manifesto for Agile Software Development*, online: <https://agilemanifesto.org>.
- Manovich Lev, *Język nowych mediów*, tłum. P. Cypryański, Wydawnictwa Akademickie i Profesjonalne, Warszawa 2006.
- Manovich Lev, *Software Takes Command*, Bloomsbury, Nowy Jork 2013.
- Marino Mark C., *Critical Code Studies*, The MIT Press, Cambridge, Mass. 2020.
- Marks Karol, *Capital, vol. I*, tłum. B. Fowkes, Penguin, Harmondsworth 1990.
- Marks Karol, *Rękopisy ekonomiczno-filozoficzne z 1844 r. w: Tenże, Pisma wybrane. Człowiek i socjalizm*. PWN, Warszawa 1979, s. 53-207.
- Maślanka Tomasz, Wiśniewski Rafał, *Wstęp do wydania polskiego*, w: Theodor W. Adorno, *Przemysł kulturalny. Wybrane eseje o kulturze masowej*, tłum. M. Bucholc, Narodowe Centrum Kultury Warszawa 2019, s. 9-29.
- Mazur Marian, *Modelowanie cybernetyczne i jego przydatność w modelowaniu procesu dydaktycznego*, w: *Problemy modelowania procesów dydaktycznych*, red. E. Berezowski, PWN, Warszawa 1978, online: [http://www.autonom.edu.pl/publikacje/mazur\\_marian/modelowanie\\_cybernetyczne\\_i\\_jego\\_przydatnosc\\_w\\_modelowaniu\\_procesow\\_dydaktycznych.php](http://www.autonom.edu.pl/publikacje/mazur_marian/modelowanie_cybernetyczne_i_jego_przydatnosc_w_modelowaniu_procesow_dydaktycznych.php).
- McLeod Mary, *Architecture or Revolution: Taylorism, Technocracy, and Social Change*, „Art Journal” 2014, nr 43(2), s. 133-147.
- Mehrotra Dhruv, *Horror Stories From Inside Amazon's Mechanical Turk*, 01.28.2020, online: <https://gizmodo.com/horror-stories-from-inside-amazons-mechanical-turk-184087804>

- Miller George A., *The cognitive revolution: a historical perspective*, „Trends in Cognitive Science” 2003, tom 7, nr. 4 s. 141-144.
- Miłkowski Marcin, *Kognitywistyka i modele umysłu*, „Dwutygodnik.com” 2016, nr 176, online:  
<https://www.dwutygodnik.com/artykul/6322-kognitywistyka-i-modele-umyslu.html>.
- Mokyr Joel, *Editor's Introduction. The New Economic History and the Industrial Revolution*, w: *The British Industrial Revolution. An Economic Perspective*, red. J. Mokyr, Westview Press, Oxford 1999, s. 1-127.
- Mori Masahiro, *The Uncanny Valley*, tłum. K. F. MacDorman, N. Kageki, „IEEE Robotics & Automation Magazine” 2012, nr 19(2), s. 98–100.
- Evgeny Morozov, *The Net Delusion: The Dark Side of Internet Freedom*, Public Affairs, Nowy Jork 2012.
- Mozur Paul, *A Genocide Incited on Facebook, With Posts From Myanmar's Military*, 15.10.2018, online:  
<https://www.nytimes.com/2018/10/15/technology/myanmar-facebook-genocide.html>.
- Murray Andrew D., *The Regulation of Cyberspace: Control in the Online Environment*, Routledge-Cavendish, Milton Park, Abingdon, UK, Nowy Jork 2006.
- Muslim Women Ask Apple To Make Face Recognition Work With Burqa Too*, 09.02.2022, online:  
<https://thefauxy.com/muslim-women-ask-apple-to-make-face-recognition-work-with-burqa-too>.
- Muzammil K., *Why Indian Programmers are Most Preferred?*, 07.08.2023, online:  
<https://www.aalpha.net/articles/why-indian-programmers-are-most-preferred>.
- Myoo Sidey, *Ontoelektronika*, Wydawnictwo Uniwersytetu Jagiellońskiego, Kraków 2013.
- Nacher Anna, *Cyfrowa humanistyka na styku sztuki, nauki i technologii*, „Czas Kultury” 2015, nr 2, s. 20-26.
- Newell Allen, Simon Herbert A., *Computer Science as Empirical Inquiry: Symbols and Search*, „Communications of the ACM” 1976, tom 19, nr 3, s. 113-126.
- O’Neil Cathy, *Broń matematycznej zagłady. Jak algorytmy zwiększają nierówności i zagrażają demokracji*, tłum. M. Z. Zieliński, Wydawnictwo PWN, Warszawa 2017.
- O’Neil Cathy, *The Weapons of Math Destruction. How Big Data Increases Inequality and Threatens Democracy*. Crown, Nowy Jork 2016.

- O'Reilly Tim, *What is Web 2.0. Design Patterns and Business Models for the Next Generation of Software*, 30.09.2005, online: <https://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html?page=4>.
- Osler Margaret J., *Mechanical Philosophy*, w: *New Dictionary of Ideas*, red. M.C. Horowitz, Charles Scribner's Sons, Nowy Jork 2005, s. 1389-1392.
- Opowieść, w: *Słownik Terminów Testowych ISTQB. Wersja 3.4*, tłum. L. Stapp, Stowarzyszenie Jakości Systemów Informatycznych 2020, s. 65.
- Nuechterlein Jonathan E., Weiser Philip J., *Digital Crossroads. American Telecommunications Policy in the Internet Age*, The MIT Press, Cambridge, Mass. 2005.
- Paluchowski Władysław Jacek, *Spór metodologiczny czy spór koncepcji. Badania ilościowe vs jakościowe*, „Roczniki psychologiczne” 2010, tom 10, nr 1, s. 7-22.
- Papakyriakopoulos Orestis, Hegelich Simon, Shahrezaye Morteza, Serrano Juan Carlos Medina, *Social media and microtargeting: Political data processing and the consequences for Germany*, „Big Data & Society”, nr 5(2) 2018, s. 1-15.
- Paprzycka Anna, *Znaczenie praktyki dla badacza kultury cyfrowej*, „Acta Universitatis Lodziensis. Folia Litteraria Polonica” 2021, nr 60(1), s. 79–91.
- Parry Encounters the DOCTOR*, online: <https://datatracker.ietf.org/doc/html/rfc439>.
- Patton Jeff, *User Story Mapping*, O'Reilly, Beijing, Köln, Cambridge 2014.
- Penny Simon, *Art and Artificial Life – a Primer*, 23.07.2011, online: [http://simonpenny.net/texts/Resources/a\\_life.pdf](http://simonpenny.net/texts/Resources/a_life.pdf).
- Perlis Alan J., *Foreword*, w: Abelson Harold, Sussman Gerald Jay, *Structure and Interpretation of Computer Programs*, The MIT Press, Cambridge, Mass. 1996, s. XI-XIII.
- Philips Anna Lena, *Crowdsourcing Gender Equity. Ada Lovelace Day, and its companion website, aims to raise the profile of women in science and technology*, 04.10.2016, online: <http://web.archive.org/web/20161004185338/https://www.americanscientist.org/issues/pub/crowdsourcing-gender-equity>.
- Pickering Andrew, *The Cybernetic Brain: Sketches of Another Future*, University of Chicago Press, Chicago 2011.
- Piekarski Karol, *Od modernizmu do medialabów. Krytyczne projektowanie w czasach nadmiaru danych*, „Formy” 2019, nr 2, online:

- <https://formy.xyz/arttykul/od-modernizmu-do-medialabow-krytyczne-projektowanie-w-czasach-nadmiaru-danych>.
- Pietrowicz Krzysztof, *Metafora a zmiana kulturowa*, „Doctrina. Studia społeczno-polityczne” 2004, nr 1, s. 45-55.
- Pomerantsev Peter, Weiss Michael, *The Menace of Unreality: How the Kremlin Weaponizes Information, Culture and Money*, The Institute of Modern Russia, Nowy Jork 2014.
- Porter Theodore M., *Trust in numbers. The pursuit of objectivity in science and public life*, Princeton University Press, Princeton, New Jersey 1995.
- Potter Claire Bond, *Political junkies: from talk radio to twitter, how alternative media hooked us on politics and broke our democracy*, Nowy Jork, Basic Books, 2020.
- Qezavati Hoda, Majidi Babak, Manzuri Mohammad Taghi, *Partially Covered Face Detection in Presence of Headscarf for Surveillance Applications*, w: *4th International Conference on Pattern Recognition and Image Analysis (IPRIA)*, IEEE 2019, s. 195-199.
- Randall Adrian J., *The Philosophy of Luddism. The Case of the West of England Woolen Workers, ca 1790-1809*, „Technology and Culture” 1986, nr 27(1), s. 1-17.
- Rheingold Howard, *A slice of life in my virtual community*, w: *Collective Intelligence. Creating a Prosperous World at Peace*, red. M. Tovey, Earth Intelligence Network, Oakton, 2008, s. 173-196.
- Rifkin Jeremy, *The Third Industrial Revolution. How lateral power is transforming energy, the economy and the world*, Palgrave Macmillan, Nowy Jork 2011.
- Riskin Jessica, *Eighteenth-Century Wetware*, „Representations” 2003, tom 83, nr 1, s. 97-125.
- Riskin Jessica, *The Defecating Duck, or the Ambiguous Origins of Artificial Life*, „Critical Inquiry” 2003, tom 29, nr 4, s. 599-633.
- Rogers Adam, *What Deep Blue And AlphaGo Can Teach Us About Explainable AI*, 09.05.2019, online: <https://www.forbes.com/sites/forbestechcouncil/2019/05/09/what-deep-blue-and-alpha-go-can-teach-us-about-explainable-ai/?sh=7c78a8b952fd>.
- Royce Winston W., *Managing the development of large software systems: concepts and techniques*, „IEEE CS Press, 1970, s. 328-339, online: <https://web.archive.org/web/20160318002949/http://www.cs.umd.edu/class/spring2003/cm3838p/Process/waterfall.pdf>.

- Rosen William, *The Most Powerful Idea in the World: A Story of Steam, Industry, and Invention*, Random House, Nowy Jork 2010.
- Rzqdzq nami algorytmy, a wiemy o nich niewiele, 06.02.2018 online: <https://p.dw.com/p/3Cqf7>.
- Sack Warren, *The Software Arts*, The MIT Press, Cambridge, Mass. 2019.
- Scholz Trebor, *Uberworked and Underpaid. How Workers Are Disrupting the Digital Economy*, Polity Press, Cambridge, UK 2017.
- Samuels Alana, *The Internet Is Enabling a New Kind of Poorly Paid Hell*, 23.01.2018, online: <https://www.theatlantic.com/business/archive/2018/01/amazon-mechanical-turk/551192>.
- Sapir Edward, *Kultura, język, osobowość. Wybrane eseje*, tłum. R. Zimand, Państwowy Instytut Wydawniczy, Warszawa 1978.
- Schaff Adam, *Wstęp do polskiego wydania*. w: Benjamin L. Whorf, *Język, myśl i rzeczywistość*, Państwowy Instytut Wydawniczy, Warszawa 1982, s. 5–31.
- Schmidt Eric, *Internet World Trade Show*, Nowy Jork 1999, online: <https://www.oxfordreference.com/view/10.1093/acref/9780191826719.001.0001/q-oro-ed4-00017947>.
- Schwab Klaus, *The Fourth Industrial Revolution*, Portfolio Penguin, Londyn 2017.
- Shirkly Clay, *Cognitive Surplus: How Technology Makes Consumers into Collaborators*, Penguin Books, Nowy Jork 2010
- Sikandar Tasriva, Ghazali Kamarul H., Mohd Izzeldin I., Rabbi M. F. , *Skin Color Pixel Classification for Face Detection with Hijab and Niqab*, w: *Proceedings of the International Conference on Imaging, Signal Processing and Communication, ICISPC 2017*, s. 1–4.
- Silpasuwanchai Chaklam, Ren Xiangshi, *A Quick Look at Game Engagement Theories*, w: *The Wiley Handbook of Human Computer Interaction*, red. K.L. Norman, John Wiley & Sons, Nowy Jork, s. 657-679.
- Simpson Christopher, *Science of Coercion: Communication Research and Psychological Warfare 1945-1960*, Oxford University Press, Oxford 1994.
- Składanek Marcin, *Sztuka generatywna. Metoda i praktyki*. Wydawnictwo Uniwersytetu Łódzkiego, Łódź 2017.
- Składanek Marcin, *Od Isotype do Many Eyes. Społeczna odpowiedzialność designu w praktykach wizualizacji danych*, „Przegląd Kulturoznawczy” 2020, 9(1), s. 37-54.

- Slotten Hugh Richard, *Radio and television regulation: broadcast technology in the United States, 1920-1960*, Johns Hopkins University Press, London 2003.
- Smith C. U., *Descartes' Pineal Neuropsychology*, „Brain and Cognition” 1998, tom 36, nr 1, s. 57-72.
- Smykowski, Mikołaj, *Szachy – (nie)ludzka gra. Ludzie, maszyny i antycypacje przyszłości*, „Kultura Współczesna” 2018, nr 2(101), s. 132-142.
- Software w: *Encyclopedia Britannica*, online: <https://www.britannica.com/technology/software>.
- Software Studies: A Lexicon*, red. M. Fuller, MIT Press, Cambridge, Mass. 2008.
- Srnicek Nick, *Platform Capitalism*, Polity Press, Malden 2017.
- Stallman Richard, *Why software should have no owners*, online: <https://www.gnu.org/philosophy/why-free.html>.
- Standage Tom, *The Turk. The Life and Times of The Famous Eighteenth-Century Chess-Playing Machine*, Walker & Company, Nowy Jork 2002.
- Stanowisko Fundacji Panoptykon w sprawie Aktu prawnego o usługach cyfrowych (Digital Services Act)*, online: [https://panoptykon.org/sites/default/files/stanowiska/panoptykon\\_kprm\\_uwagi\\_do\\_dsa\\_20.01.2021.pdf](https://panoptykon.org/sites/default/files/stanowiska/panoptykon_kprm_uwagi_do_dsa_20.01.2021.pdf).
- Steele Chandra, *The Real Reason Voice Assistants Are Female (and Why it Matters)*, PC News, 4.01.2018, online: <https://docs.google.com/document/d/1f62Z4uz9RGXHUVmM-MSGG2iu3V5MuCgzNGQCymku5gI/edit>.
- Steckflow Steve, *Why Facebook is losing the war on hate speech in Myanmar*, 15.10.2018, online: <https://www.reuters.com/investigates/special-report/myanmar-facebook-hate>.
- Sterne Jonathan, *MP3: The Meaning of a Format*, Duke University Press, Durham 2012.
- Szafraniec Asja, *Deleuze and New Technology*, „Journal of Philosophy” 2019, nr 19(3), s. 484-490.
- Szota Michał, *Magiczna sztuka programowania komputerów*, „Kultura Popularna” 2008, nr 4(22), s. 37-40.
- Szpyt-Wiktorowska Joanna, *Strategie mediów wobec baniek informacyjnych*, „Zarządzanie Mediami” 2018, nr 6(1), s. 41-50.
- Taylor John G., Fragopanagos Nickolaos F., *The interaction of attention and emotion*, „Neural Networks” 2005, nr 18(4), 353–369.

- Terranova Tiziana, *Free Labor: Producing Culture for the Digital Economy*, „Social Text” 2000, nr 18 (2), s. 33-58.
- Terranova Tiziana, *Network Culture: Cultural Politics for the Information Age*. Pluto Press, Nowy Jork 2004.
- The Politics of Platformization. Amsterdam Dialogues on Platform Theory*, red. G. Cristofari, Institute of Network Cultures, Amsterdam 2023.
- Toseeb Umar, Bryant Eleanor J., Keeble David R. T., *The Muslim Headscarf and Face Perception: „They All Look the Same, Don't They?”*, „PLOS one” 2014, nr 9(2), s. 1-9, online: <https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0084754&type=printable>, dostęp: 06.11.2023.
- Turing Alan M., *Digital Computers Applied to Games*, w: *Faster Than Thought. A Symposium on Digital Computing Machines*, red. B.V. Bowden, Sir Isaac Pitman & Sons, London 1953, s. 286-311.
- Turing Alan M., *Intelligent Machinery, a Heretical Theory*, „Philosophia Mathematica” 1951, nr 4(3), s. 256-260.
- Turing Alan M., *Maszyna licząca a inteligencja*, tłum. M. Szczubińska, w: *Filozofia umysłu*, red. B. Chwedeńczuk, Fundacja ALETHEIA — Wydawnictwo Spacja, Warszawa 1995, s. 271-300.
- Turkle Sherry, *Alone Together. Why We Expect More from Technology and Less from Each Other*, Basic Books, Nowy Jork 2010.
- Turkle Sherry, *The Second Self: Computers and the Human Spirit*, The MIT Press, Cambridge, Mass. 2004.
- Turque Bill, *‘Creative...motivating’ and fired*, 06.03.2012, online: [https://www.washingtonpost.com/local/education/creative--motivating-and-fired/2012/02/04/gIQAwzZpvR\\_story.html](https://www.washingtonpost.com/local/education/creative--motivating-and-fired/2012/02/04/gIQAwzZpvR_story.html).
- Turque Bill, *Rhee Spells Out Teaching Expectations With 200-Page 'Learning Framework'*, 23.08.2009, online: <https://www.washingtonpost.com/wp-dyn/content/article/2009/08/22/AR2009082202251.html>.
- Veet Annette, *Coding Literacy: How Computer Programming Is Changing Writing*, The MIT Press, Cambridge, Mass. 2017.

- Vincent James, *Twitter taught Microsoft's AI chatbot to be a racist asshole in less than a day*, 24.03.2016, online: <https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist>.
- Virilio Paul, *Speed and Information: Cyberspace Alarm!*, online: <http://www.tacticalmediafiles.net/n5m2/media/texts/virilio.htm>.
- Walters Karly, *Law, 'terror' and the Frame-Breaking Act*, w: *The Economic History Society. Annual Conference*, Royal Holloway University of London 2004, s. 23-28.
- Ward Ken, *Social networks, the 2016 US presidential election, and Kantian ethics: applying the categorical imperative to Cambridge Analytica's behavioral microtargeting*, „Journal of Media Ethics” 2018, nr 33(3), s. 133-148.
- Wardrip-Fruin Noah, *Expressive Processing: Digital Fictions, Computer Games, and Software Studies*, The MIT Press, Cambridge, Mass. 2009.
- Watkins Ian, *The Utility of Level of Service Inventory – Revised (LSI-R) Assessments within NSW Correctional Environments*, „Research Bulletin (Corrective Services NSW)”, nr 28, s. 1-8.
- Webb Maureen, *Coding Democracy. How Hackers Are Disrupting Power, Surveillance, and Authoritarianism*, The MIT Press, Cambridge, Mass. 2021.
- Web2.0 Conference, 2004, online: <http://web.archive.org/web/20040602111547/http://web2con.com>
- Weizenbaum Joseph, *Computer Power and Human Reason: From Judgment to Calculation*, W. H. Freeman, San Francisco 1976.
- What is scrum?*, online: <https://www.scrum.org/resources/what-scrum-module>.
- Whorf Benjamin L., *Język, myśl i rzeczywistość*, Państwowy Instytut Wydawniczy, Warszawa 1982.
- Wilber Ken, *Trump and a Post-Truth World*, Shambala, Boulder 2017.
- Williams Robin, Cummings Steve *Jargon: An Informal Dictionary of Computer Terms*, Peachpit Press, Berkeley, California 1993.
- Wilson Kate Silverman, *The MIT Press relaunches the Software Studies series*, 13.01.2022, online: <https://news.mit.edu/2022/mit-press-relaunches-software-studies-series-0613>.
- Wstęp do kulturoznawstwa*, red. E. Baldwin, tłum. M. Kaczyński, J. Łoziński, T. Rosiński, Zys i S-ka, Poznań 2007.
- Wujec Paweł, *Szach-mat, enter – Kasparow kontra Deep Blue*, „Gazeta Wyborcza”, 1305.1997, online: <https://wyborcza.pl/7,75400,1290174.html?disableRedirects=true>.

- Wylie Christopher, *Mindf\*ck: Cambridge Analytica and the Plot to Break America*, Random House, Nowy Jork 2019.
- Zawojski Piotr, *Cyberkultura. Syntopia sztuki, nauki i technologii*, Wydawnictwo Poltext, Warszawa 2010.
- Zehner Brett, *Machines of Subjection: Notes on a Tactical Approach to Artificial Intelligence*, „APRJA” 2019, nr 8(1), s. 48-57.
- Zehner Brett, *The Situation in Micronesia: The Rise of Dispossession*, „Media-N” 2023, nr 19 (1), s. 27-41.
- Zhou Li, Gao Jianfeng, Li Di, Shum Heung-Yeung, *The Design and Implementation of XiaoIce, an Empathetic Social Chatbot*, „Computational Linguistics” 2020, tom 46, nr 1, s. 53-93.
- Zuboff Shosana, *Wiek kapitalizmu inwigilacji. Walka o przyszłość ludzkości na nowej granicy władzy*, tłum. A. Unterscheutz, Zysk i S-ka, Poznań 2020.
- Žižek Slavoj, *The Ticklish Subject: The Absent Centre of Political Ontology*, Verso, London, 1999.
- 10 Print Chr#36;(205. 5+rnd(1));:Goto 10*, red. N. Monfort, The MIT Press, Cambridge, Mass. 2012.
- 16th State of Agile Raport*, online:  
<https://digital.ai/resource-center/analyst-reports/state-of-agile-report>.

### **Filmy:**

*AlphaGo*, reż. Greg Kohs, 2017, USA.

*Climategate: Science of a Scandal*, 2019, reż. Steve O'Hagan.

*Data Art: The Sheep Market*, 07.03.2008, online:  
<https://www.youtube.com/watch?v=3Mmb5aScck>.

*EMET*, 12.11.2018, online: <https://vimeo.com/300315429>.

*Facebook CEO Mark Zuckerberg testifies before Congress on data scandal*, 10.04.2018,  
online: <https://www.youtube.com/watch?v=u-FlWZ1BOcA>

Nix Alexander, *Cambridge Analytica – The Power of Big Data and Psychographics*,  
27.09.2016, online: <https://www.youtube.com/watch?v=n8Dd5aVXLCc>.

*The Great Hack*, reż. Karim Amer, Jehane Noujaim, USA 2019.

*Dylemat społeczny*, reż. Jeff Orlowski, USA 2020.

*The Thinking Machine (Artificial Intelligence in the 1960s)*, 31.05.2010, online:  
<https://www.youtube.com/watch?v=aygSMgK3BEM>.

*Zakodowane Upředzenie*, reż. Shalini Kantayya, Chiny, USA, Wielka Brytania 2020.

### **Platformy internetowe:**

<https://www.nask.pl>

<https://medialabkatowice.eu/en>

<https://www.mturk.com>

<http://www.thesheepmarket.com>

<http://www.thimbl.net>

## Spis Ilustracji

- Il. 1. Komputer ENIAC, 1940. Źródło: Liz Huang, *Refrigerator Ladies: The First Computer Programmers*, online: <https://tanzu.vmware.com/content/blog/refrigerator-ladies-the-first-computer-programmers>, dostęp: 20.11.2023.
- Il. 2. „Kaczka” Vaucasona, przekrój. Źródło: *Canard Digérateur de Vaucanson*, online: <https://www.atlasobscura.com/places/canard-digérateur-de-vaucanson-vaucansons-digesting-duck>, dostęp: 20.11.2023.
- Il. 3. „Turek” Kempellena z otwartymi drzwiczkami, Źródło: Tom Standage, *The Turk. The Life and Times of The Famous Eighteenth-Century Chess-Playing Machine*, Walker & Company, Nowy Jork 2002, s. 85.
- Il. 4. „Turek” Kempellena z widocznym człowiekiem w środku. Źródło: Tom Standage, dz. cyt., s. 87.
- Il. 5. Kadr z filmu AlphaGo (2017). Scena przedstawia pojedynek Lee Sedola z programem AlphaGo. Źródło: *AlphaGo*, reż. Greg Kohs, 2017.
- Il. 6. *Drawing Hands* Mauritsa Cornelisa Eschera, 1948. Źródło: *M.C. Escher's „Drawing Hands”*, online: <https://moa.byu.edu/m-c-eschers-drawing-hands>.
- Il. 7. Moje pierwsze eksperymenty z programami typu Quine w języku Processing. Źródło: opracowanie własne.
- Il. 8. *EMET* (2018). Kadr z dokumentacji video. Źródło: opracowanie własne.
- Il. 9. Kadr z filmu *Dylemat społeczny*. Metaforyczne przedstawienie oprogramowania jako trzech mężczyzn. Źródło: *Dylemat społeczny*, reż. Jeff Orlowski, USA 2020.
- Il. 10. Kadr z filmu *Dylemat społeczny*. Programistka usuwająca fragment kodu. Źródło: *Dylemat społeczny*, dz. cyt.
- Il. 11. Kadr z filmu *Dylemat społeczny*. Metaforyczne przedstawienie oprogramowania po przemianie. Źródło: *Dylemat społeczny*, dz. cyt.
- Il. 12. Kadr z filmu *Zakodowane uprzedzenie*. Abstrakcyjna reprezentacja oprogramowania. Źródło: *Zakodowane Uprzedzenie*, reż. Shalini Kantayya, Chiny, USA, Wielka Brytania 2020.
- Il. 13. *Facebook Algorithmic Factory*. Źródło: <https://wro2017.wrocenter.pl/facebook-algorithmic-factory>, dostęp: 06.11.2023.

- Il. 14. *The Sheep Market*. Zbliżenie na fragment kolażu. Źródło: online:  
<http://www.aaronkoblin.com/work/thesheepmarket>, dostęp: 06.11.2023.
- Il. 15. *The Sheep Market*. Kolaż składający się z 10 000 prac stworzonych przez użytkowników MTurk. Źródło: online: <http://www.aaronkoblin.com/work/thesheepmarket>, dostęp: 06.11.2023.

## **SUMMARY**

The thesis is focused on the issue of the cultural significance of software. Based on the metaphor of software as a cultural operating system, it is presented how software registers, processes and updates existing cultural values. The considerations contained in the dissertation are based on the contributions of the software studies discipline, which has been developing since the beginning of the 21st century. The first chapter of the dissertation is a theoretical introduction providing an overview of the most important methodologies, research directions and profiles of researchers associated with this discipline. In the next part of the dissertation, considerations are undertaken on the social conditioning of the processes of production and interaction with software. The starting point for these considerations is the category of model, which was used to show how psychological subconscious content first influences the design of digital technologies and then how users interact with them. Another cultural problem analyzed is the issue of using software as a tool of social and political control, which is shown through the example of social media. The final part of the dissertation includes economic considerations, dealing with the issue of capitalization of user activity, as well as conscious and unconscious non-material labor online. The culmination of this part of the dissertation is the presentation of alternative forms of software production in the capitalist system associated with the free software movement, such as, for example, the idea of venture communism or copyleft and copyleft licenses. Each chapter is preceded by an introduction that provides a historical context for the issues discussed in it. The purpose of these references is to present the fact that the cultural problems analyzed are not solely due to the operation of software, but in some aspects developed even before the digital age began.

## STRESZCZENIE

Praca poświęcona została zagadnieniu kulturowego znaczenia oprogramowania. W oparciu o metaforę oprogramowania jako systemu operacyjnego kultury przedstawione zostało to, jak software rejestruje, przetwarza i aktualizuje zastane wartości kulturowe. Zawarte w dysertacji rozważania opierają się na dorobku rozwijającej się od początku XXI wieku dyscypliny software studies. Pierwszy rozdział pracy stanowi teoretyczne wprowadzenie przybliżające najważniejsze metodologie, kierunki badań i sylwetki badaczy i badaczek związanych z tą dyscypliną. W dalszej części pracy podjęte zostały rozważania na temat społecznego uwarunkowania procesów wytwarzania i interakcji z oprogramowaniem. Punktem wyjścia dla tych rozważań jest kategoria modelu, która wykorzystana została do tego, aby pokazać, jak psychologiczne treści podświadome wpływają najpierw na projektowanie technologii cyfrowych, a następnie na to, w jaki sposób użytkownicy wchodzi z nimi w interakcję. Kolejnym analizowanym problemem kulturowym jest kwestia wykorzystywania oprogramowania jako narzędzia kontroli społecznej i politycznej, co zostaje przedstawione na przykładzie mediów społecznościowych. W ostatniej części pracy zawarte zostały rozważania ekonomiczne, dotyczące kwestii kapitalizacji aktywności użytkowników, a także świadomej i nieświadomej pracy niematerialnej w sieci. Zwieńczenie tej części dysertacji stanowi przedstawienie alternatywnych form wytwarzania oprogramowania w systemie kapitalistycznym związane z ruchem wolnego oprogramowania, takich jak chociażby idea venture communism czy licencje copyleft i copyleft. Każdy rozdział poprzedzony został wprowadzeniem przybliżającym kontekst historyczny dla zagadnień w nim omawianych. Celem tych odwołań jest przedstawienie tego, że analizowane problemy kulturowe nie wynikają wyłącznie z działania oprogramowania, ale w pewnych aspektach rozwijały się jeszcze przed rozpoczęciem się epoki cyfrowej.