Uniwersytet im. Adama Mickiewicza w Poznaniu   
Wydział Neofilologii  
Instytut Językoznawstwa  
  
Numer albumu: 456498

Kierunek studiów: lingwistyka  
Specjalność: lingwistyka komputerowa

Adam Michał Śliwiński  
  
**System dialogowy Aestus - pomocniczy program dla pracowników i użytkowników pływalni z elementami syntezy i rozpoznawania mowy**

Praca licencjacka  
napisana pod kierunkiem   
dr Jolanty Bachan

Poznań 2022

OŚWIADCZENIE

Ja, niżej podpisana Patrycja Świeczkowska, studentka Wydziału Neofilologii  
Uniwersytetu im. Adama Mickiewicza w Poznaniu oświadczam, że przedkładaną pracę dyplomową pt: „Asystent – system dialogowy jako pomocnik w realizacji planu dnia” 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 części innym osobom, ani nie odpisywałam tej rozprawy lub jej części od innych osób. Oświadczam również, że egzemplarz pracy dyplomowej w wersji drukowanej jest

całkowicie zgodny z egzemplarzem pracy dyplomowej w wersji elektronicznej. Jednocześnie przyjmuję do wiadomości, że przypisanie sobie, w pracy dyplomowej,

autorstwa istotnego fragmentu lub innych elementów cudzego utworu lub ustalenia naukowego stanowi podstawę stwierdzenia nieważności postępowania w sprawie nadania tytułu zawodowego.

]\* - wyrażam zgodę na udostępnianie mojej pracy w czytelni Archiwum UAM ]\* - wyrażam zgodę na udostępnianie mojej pracy w zakresie koniecznym do

ochrony mojego prawa do autorstwa lub praw osób trzecich

\*Należy wpisać TAK w przypadku wyrażenia zgody na udostępnianie pracy w czytelni Archiwum UAM, NIE w przypadku braku zgody. Niewypełnienie pola oznacza brak zgody na udostępnianie pracy.

..............................................  
(czytelny podpis studenta)

Streszczenie pracy w języku polskim

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

. <GDZIES TYLE>

.

.

.

.

.

.

.

.

.

Słowa kluczowe: system dialogowy, zarządzanie obiektem rekreacyjno-sportowym, komunikacja człowieka z komputerem, Python

Streszczenie pracy w języku angielskim (Abstract)

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

. <GDZIES TYLE>

.

.

.

.

.

.

.

.

.

Keywords: dialgue system, zarządzanie obiektem rekreacyjno-sportowym, komunikacja człowieka z komputerem, Python

**Spis treści**Streszczenie pracy w języku polskim

Streszczenie pracy w języku angielskim (Abstract)

Spis treści

Spis rysunków i tabel

Wstęp

Rozdział I: Systemy dialogowe

1.1 Czym są systemy dialogowe

1.1.1 Czym jest dialog, wypowiedź, intencja, tura

1.1.2 Typy Systemów dialogowych

1.1.3 Schematy systemów dialogowych

1.1.4 Jak zbudowane są systemy dialogowe

1.1.5 Zastosowania systemów dialogowych

1.2 Historia systemów dialogowych

1.2.1 Imitation game Alana Turinga

1.2.2 DARPA, Georgetown-IBM

1.2.2 ELIZA, PARRY

1.2.3 SHRDLU

1.2.4 Siri, Cortana, Google Assistant, Amazon Alexa

1.3 Oprogramowanie do ustalania harmonogramów

1.3.1 Oprogramowanie do rezerwacji terminów

1.3.2 Oprogramowanie do tworzenia harmonogramów dla pracowników

1.4 Podsumowanie

Rozdział II: Interakcja z Aestus

2.1 Rodzaje komunikacji

2.1.1 Człowiek z człowiekiem (międzyludzka)

2.1.2 Człowiek z maszyną

2.2.3 Komunikacja programu Aestus z człowiekiem

2.2 Program Aestus

2.2.1 Specyfikacje programu Aestus (z Agile: Requirements)

2.2.2 Aestus na tle klasycznych systemów dialogowych

2.3 Komunikacja programu Aestus z człowiekiem

2.4 Podsumowanie

Rozdział III: Projekt Aestus

3.1 Agile

3.1.1 Czym jest Agile, Sprint

3.1.2 Projekt

3.1.2.1 Cel programu

3.1.2.2 Scenariusz i schemat blokowy

3.1.2.3 Automat skończenie stanowy

3.1.2.4 Użyty język programowania

3.1.2.4.1 Python 3.10

3.1.2.5 Użyte API

3.1.2.5.1 (TBD)

3.1.3 Opracowanie

3.1.3.1 Wykonanie i połączenie nagrań

3.1.3.1.1 Anotacja nagrań

3.1.3.1.2 Analiza statystyczna anotacji

3.1.4 Testowanie

3.1.4.1 Ankietowanie

3.1.4.1.1 Wersja testowa prgoramu Aestus

3.1.4.1.2 Interpretacja wyników

3.1.4.2 Debugowanie - spostrzeżenia

3.1.5 Wdrożenie

3.1.5.1 Wymagania Aestus

3.1.5.2 Możliwości Aestus

3.1.6 Przegląd

3.1.6.1 Mocne i słabe strony programu

3.2 Podsumowanie

Zakończenie

Załącznik 1: Kod programu

Załącznik 2: Instrukcja użytkowania

Załącznik 3: Przykładowy dialog

Załącznik 4: Ankieta dla testujących

**Spis rysunków i tabel**

Rysunek 1: Schemat architektury systemu dialogowego

Rysunek 2: Wykonywanie nagrań --------- usunac

Rysunek 3: Anotacja nagrań ------------ usunac

Rysunek 4: Analiza statystyczna anotacji ------- usunac

Rysunek 5: Automat skończenie stanowy

Rysunek 6-n: Przykładowe dialogi

Tabela 1: Wyniki ankiet

**Wstęp**

Od czasu, w którym w życiu człowieka pojawił się internet, ludzki postęp we wszelkich dziedzinach nauki niesamowicie wzrósł. Lata sześćdziesiąte przypieczętowały początek nowej ery, ery informacji (J. C. R. Licklider, 1960). Nie minęło trzydzieści lat, aby pojawiła się pierwsza wyszukiwarka, która zrewolucjonizowała dostęp do stron internetowych (Sergey Brin, Lawrence Page, The anatomy of a large-scale hypertextual Web search engine, 1998) i usług, bez których dzisiaj mało kto jest w stanie się obejść. Jeszcze dwadzieścia lat temu niewiele osób mogłoby przewidzieć, że w urządzeniu wielkości dłoni takim jak smartfon czy tablet będziemy mieli dostęp do największej skarbnicy wiedzy i informacji na świecie, jaką jest internet. W perspektywie czasu ta informacyjna rewolucja zdaje się tylko przyspieszać a popyt na wiedzę, którą niesie ze sobą, wcale nie maleje. Niewątpliwie prowadzi to do wzrostu potrzeb na innowacje, nie tylko dla wygody, ale również dla zysku. Już teraz można zauważyć, jak wiele korporacyjnych gigantów konkuruje ze sobą, tworząc nowsze i coraz bardziej inteligentne rozwiązania w wyścigu o zyski bądź uwagę potencjalnego nabywcy elektronicznych sprzętów i usług. Żyjąc w świecie, w którym coraz to bardziej ceni się wygodę, tradycyjne metody interakcji z urządzeniami są zastępowane w wielu interfejsach użytkownika przez urządzenia wykorzystujące bardziej intuicyjne techniki, takie jak mowa lub dotyk. Klawiaturę i myszkę, może zastąpić wtedy interfejs głosowy, wykorzystujący najbardziej naturalny i najefektywniejszy sposób komunikacji między ludźmi. (Wiśniewski, 2009, str. 97.)  
 Co więcej, interfejsy głosowe z dnia na dzień stają się coraz bardziej popularne. Znaczna większość urządzeń, takich jak komputery czy smartfony, które pojawiają się na rynku, posiadają swego rodzaju interfejs głosowy bądź asystenta, który nasłuchuje i czeka na wywołanie przez specjalną komendę. Użyteczność owych systemów przykładowo objawia się w momencie, w którym prowadzimy samochód i nasze skupienie na drodze (jak i przepisy ruchu drogowego) nie pozwala na skorzystanie z telefonu. Krótki dialog z asystentem może być wtedy niezwykle użyteczny.  
 Interakcje z użytkownikiem w systemach dialogowych mogą odnosić się do wyboru opcji z menu bądź prowadzenia konwersacji z użytkownikiem. Jako że interakcja międzyludzka jest procesem niezwykle złożonym, wdrażanie jej w jakikolwiek system wiąże się z wykorzystaniem sztucznej inteligencji. Zapewnienie efektywnej interakcji człowieka z wirtualną maszyną stanowi wieloaspektowe zagadnienie. Ludzka inteligencja, która pozwala na porozumiewanie, współdziała ze zdolnościami w sferze emocjonalnej, motywacyjnej i interpersonalnej, co stwarza proces komunikacji wielopłaszczyznowym. Zaprojektowanie więc schematu programu, który pozwoliłby na równym poziomie rozmawiać z człowiekiem jest wyzwaniem nawet dla największych firm i najlepszych programistów (Kułakowska, 2015, str 8.).  
 Celem niniejszej pracy jest zaprojektowanie systemu dialogowego, który wykorzystuje techniki rozpoznawania mowy, syntezy głosowej i komunikacji z bazą danych, aby stworzyć wygodny i prosty dla użytkownika interfejs. Zadaniem proponowanego systemu dialogowego o nazwie Aestus (łac. przypływ, fala) jest ułatwienie pracownikom i klientom obiektu rekreacyjnego, w tym przypadku basenu, między innymi: śledzenia godzin pracy czy kontroli rezerwacji torów do pływania. Program opiera się na wyborze głosowym z dostępnych opcji z menu a model programu jest zaprojektowany na podstawie automatów skończenie stanowych.  
 Pierwszy rozdział pracy jest ogólnym wprowadzeniem, składającym się z czterech części. Pierwsza część jest opisem, czym są systemy dialogowe. Przedstawiono tu między innymi definicję dialogu, typy, schematy, budowę i zastosowania systemów dialogowych. Kolejna część przestawia badania Alana Turinga, początki systemów dialogowych oraz opis oprogramowania do rezerwacji terminów i tworzenia harmonogramów dla pracowników.  
 Drugi rozdział zawiera informacje dotyczące rodzajów komunikacji i specyfikacje programu Aestus, który wykonany został z myślą o tej pracy, będącym narzędziem mającym na celu pomóc zarządzać publiczną pływalnią. Ostatnia część opisuje istotę komunikacji programu Aestus z człowiekiem.  
 Rozdział trzeci, szczegółowo przedstawia strukturę systemu Aestus na postawie metodyki systemu Agile. W pięciu punktach opisano projekt systemu, opracowanie, testowanie, wdrożenie i jego przegląd. Podrozdział „Projekt” zawiera cel programu, scenariusz, schemat blokowy, automat skończenie stanowy, użyty język programowania i API. Podrozdział „Opracowanie” prezentuje anotacje nagrań i ich analizę statystyczną, natomiast „Testowanie” skupia się na ankietowaniu, interpretacji wyników ankiety i debugowaniu. Ostatnie dwa podrozdziały, „Wdrożenie” i „Przegląd”, poświęcone są wymaganiom i możliwościom programu Aestus, oraz zaprezentowane są tam również jego mocne i słabe strony.   
 Zakończenie podsumowuje problemy, pomysły i zawiera rozważania, które zostały przeprowadzone w niniejszej pracy. Przedstawiono również wnioski dotyczące komunikacji człowieka z programem Aestus.   
 Praca zawiera również bibliografię i załączniki, w których znajdują się: kod programu, instrukcja użytkowania, przykładowe dialogi, ankietę dla testujących.

**Rozdział I: Systemy dialogowe**

Rozdział ten jest wprowadzeniem do wiedzy o systemach dialogowych. Zawiera

on definicje podawane przez badaczy, których zadaniem jest przedstawienie podstaw,

na których są zbudowane systemy dialogowe.

* 1. Czym są systemy dialogowe

System dialogowy to system komputerowy zdolny do interaktywnej komunikacji "krok po kroku" z użytkownikiem. (Gibbon et al. 1997) Co więcej, jest to interfejs człowiek-maszyna, który rozpoznaje i analizuje wprowadzane dane, również głosowe, aby wygenerować odpowiedź słowną w kolejnych turach, aby osiągnąć cel, taki jak uzyskanie informacji lub wykonanie działania. W kwestii nomenklatury, najczęściej stosuje się zamiennie terminy takie jak: system dialogowy języka mówionego, interfejs konwersacyjny lub system konwersacyjny. (Wiśniewski, 2009, str. 99.)   
 Prezentacja architektury systemów dialogowych, wiąże się z koniecznością omówienia czterech kluczowych elementów, którymi są: dialog, wypowiedź, intencja i tura.   
  
1.1.1 Czym jest dialog, wypowiedź, intencja, tura

Ławrynowicz (2019)\*\*\* wyróżnia cztery komponenty umożliwiające sprawne działanie jakiegokolwiek systemu dialogowego:

* Dialog - w pojęciu systemu dialogowego to pisemna lub mówiona konwersacja między dwoma lub więcej osobami lub agentami konwersacyjnymi.
* Wypowiedź - (ang. utterance) jest to najmniejsza jednostka mowy lub ciągła mowa rozpoczynająca się i kończąca wyraźną pauzą. Dodatkowo zaznacza, że wypowiedzi nie istnieją w języku pisanym, tylko ich reprezentacje, które można odzwierciedlać i opisywać w języku pisanym na wiele różnych sposobów.
* Intencja - odnosi się do konketnego celu angenta w jakim wykonuje on czynnościu lub serie działań. Innymi słowy, jakie zadanie lub cel użytkownik chce wykonać lub osiągnąć.
* Tura – pojedyńcza kolejka, na którą mogą składać się zdania, choć może być tak krótka, jak pojedyńcze słowo. (str. 4)\*\*\*

Systemy dialogowe dzielą się na dwie klasy, chatboty i agenty dialogowe (ang.   
goal-based, mające zazwyczaj specyficzny cel) gdzie klasę agentów dialogowych reprezentują na przykład interfejsy głosowe w samochodach, telefonach lub robotach.

1.2 Typy Systemów dialogowych

Próbując podzielić systemy dialogowe na grupy, można napotkać się na problem. Niektóre kategorie mogą się ze sobą pokrywać, a ich różnice często nie są wystarczjąco wyraźne, aby stworzyć konkretne zbiory, które mogłyby podzielić tak obszerny temat. Niemniej jednak, analizując dotychczasowe osiągnięcia programistów, badaczy i naukowców można stworzyć zbiory, które przedstawiają spektrum typów systemów dialogowych.

Bachan (2011, str. 34) dzieli systemy konwersacyjne na następujące kategorie:

* Systemy tekstowe – w których komunikacja opiera się na wprowadzaniu i odczytywaniu tekstu.
* Systemy dialogu mówionego – w których komunikacja opiera się na porozumiewaniu z użytkownikiem za pomocą mowy. Wzorowane są one na rozmowie człowieka z człowiekiem.
* Systemy graficznego interfejsu uzytkownika – które skupiają sie na stosowaniu obrazów i ikon do komunikacji człowiek-komputer zamiast poleceń tekstowych.
* Systemy wielomodalne – mają one coś z każdego dotychczas wymienionego rodzaju systemu konwersasyjnego, czyli łączą więcej niż jedną modalność w celu interakcji z użytkownikiem. (na podstawie Liao 2002)\*\*spr

Program przedstawiony w tej pracy będzie systemem wielomodalnym.

1.1.3 Schematy systemów dialogowych

Przy opisie architektury systemów dialowych najłatwiej jest zastosować schematy. Pokazują one krok po kroku jakie procesy zachodzą w standardowych programach, które oparte są na syntezie i rozpoznawaniu mowy jak i bazach danych.

1.1.4 Jak zbudowane są systemy dialogowe.

Wiśniewski (2009, str. 99.) przedstawia następujący schemat (Rys. 1.) typowego interfejsu konwersacyjnego.

Baza

danych

Sterownik dialogu

Parser  
(Natural Language Processing)

Generator tekstu

Urządzenie rozpoznawania mowy

Syntezator  
 (text-to-speech)

Rys. 1. Strzałki poprawić!!!!! Architektura systemu dialogowego języka mówionego

Gdzie:

* Baza danych – to zorganizowany zbiór danych, przechowywanych i dostępnych w formie elektronicznej, który ma reprezentować pewną całość. (Beynon-Davies, 2004, str. 6)
* Sterownik dialogu – szeroko rozumiany program, który łączy wszystkie elementy w spójną całość i nimi zarządza.
* Generator tekstu – to element zajmujący się wyborem wypowiedzi w turze.
* Syntezator tekstu lub TTS (z ang. text to speech, tekst na mowę) – jest odpowiedzialny za mechaniczną zamianę tekstu dostarczonego przez generator na wypowiedź w postaci dźwięku.
* Użytkownik – to osoba korzystająca z systemu.
* Urządzenie rozpoznawania mowy – to urządzenie, które pozwala na interpretacje ludzkiej mowy. Przekazuje ono rozpoznane dane do parsera.
* Parser – lub NLP (z ang. natural language processing, interpreter języka naturalnego) to element, który pozwala reagować sterownikowi dialogu na polecenia głosowe dostarczane przez urządzenie rozpoznawania mowy.

Podstawowym elementem jest tutaj sterownik dialogu, który jest największą i najbardziej złożoną częścią systemu. Zadaniem tego komponentu jest umożliwienie wymiany informacji między użytkownikiem a systemem i w razie potrzeby edytowanie bazy danych. Dialog z takim programem składa się sekwencji tur będących pytaniami (wychodzącymi od programu) i odpowiedziami (użytkownika), których celem jest przeprowadzenie takiej wymiany wypowiedzi aby zaplanowany zbiór potrzebnych informacji został dostatecznie wypełniony w celu wykonania zaplanowanych celów programu. Przy turze systemu generowany jest tekst, który w systemie dialogu mówionego jest następnie odczytywany przez syntezator mowy. W przypadku systemów wielomodalnch w tym samym czasie ten sam tekst jest wyświetlany na ekranie. Kiedy użytkownik poda odpowiedź przetwarzana jest ona przez urządzenie rozpoznawania mowy i interpretowana przez parser, który decyduje o tym jak ma zachować się sterownik dialogu. Następnie sterownik opierając się na informacji dostarczonych przez parser, uaktualnia swój stan i bazę danych a następnie przechodzi do kolejnej zaplanowanej akcji. Przerwa w interakcji z programem następuje kiedy potrzeby użytkownika zostają spełnione.

1.1.5 Zastosowania systemów dialogowych.

Interakcja głosowa człowieka z komputerem jest powszechnie wykorzystywana w sferach przedsiębiorstw biznesowych, edukacji, administracji, opieki zdrowotnej czy rozrywki. W tak licznej grupie zastosowań, przykładem mogą być (Wikipedia):

* Nawigacja po stronach internetowych, czyli prowadzenie klientów do odpowiednich części złożonych stron internetowych.
* Odpowiadanie na pytania klientów dotyczące usług za pośrednictwem firmowej strony internetowej.
* Spersonalizowana obsługa będąca agentami konwersacyjni korzystającymi z wewnętrznych i zewnętrznych baz danych w celu personalizacji interakcji, przykładowo odpowiadając na pytania dotyczące salda konta w banku, dostarczając informacji o portfelu.
* Baza wiedzy agentów obsługi klienta, która umożliwia agentom wpisywanie pytań klientów i udzielanie im odpowiedzi.
* Sprzedaż kierowana, czyli ułatwianie transakcji poprzez udzielanie odpowiedzi i wskazówek w procesie sprzedaży, zwłaszcza w przypadku złożonych produktów sprzedawanych nowym klientom.
* Wsparcie techniczne, będące reagowaniem na problemy techniczne, przykładowo diagnozowanie problemu z produktem lub urządzeniem.
* Szkolenia i edukacja, które mogą udzielać porad dotyczących rozwiązywania problemów, podczas gdy użytkownik się uczy.
* Proste systemy dialogowe są również szeroko stosowane w celu zmniejszenia obciążenia pracą ludzką w centrach telefonicznych. W telefonii przemysłowej funkcjonalność zapewniana przez systemy dialogowe jest znana jako interaktywna odpowiedź głosowa.
  1. Historia systemów dialogowych

Początki systemów dialogowych są mocno związane z ”chat-botami”. Ich rozwój rozpoczął się w 1950 roku, kiedy to Alan Turing zaczął przeprowadzać test inteligencji mający na celu pokazanie, że maszyny są zdolne do symulowania inteligencji w rozmowie z człowiekiem poprzez terminal tekstowy. Obecnie używane są znacznie bardziej złożone systemy takie jak Siri czy Alexa, ale między nimi a ELIZA i testem Turinga było wiele mniej lub bardziej udanch prób symulacji rozmowy człowieka z komputerem.

* + 1. Imitation game Alana Turinga

Test Turinga, lub Imitation game, to eksperyment zaprojektowany prez Alana Turinga mający za zadanie sprawdzić, czy badana maszyna potrafi skutecznie upozorować inteligentne zachowanie. Gdy ludzki obserwator nie jest w stanie odróżnić zachowania maszyny od zachowania człowieka wykonującego to samo zadanie, eksperyment uznaje się za udany. Alan Turing opisuję tą grę jako test, w którym biorą udział trzy osoby: mężczyzna, kobieta oraz przesłuchujący, którego płeć nie ma znaczenia. Przesłuchujący przebywa w pomieszczeniu będącym oddzielonym od pozostałych dwóch osób a celem gry dla przesłuchującego, opisanym jako C, jest ustalenie, która z dwóch pozostałych osób, opisanymi jako A i B, jest mężczyzną, a która kobietą. Zakładając, że kobieta to X i meżczyzna Y, C na koniec gry mówi: "A to X, a B to Y" lub "B to X, a A to Y". Przesłuchujący może zadawać pytania obu osobom. Załóżmy, że A jest w rzeczywistości X, wtedy A musi tak odpowiadać na pytania, żeby przekonać C, że jest Y. Celem A w grze jest doprowadzenie do tego, aby przesłuchujący dokonał błędnej identyfikacji. Więc co się stanie, gdy w tej grze maszyna wcieli się w rolę A? Czy w takiej grze przesłuchujący będzie równie często podejmował błędne decyzje, jak w przypadku gry między mężczyzną a kobietą? Turing zadaje więc następujące pytanie: "Czy maszyny mogą myśleć?". Warto zaznaczyć, że rozmowa byłaby ograniczona do kanału tekstowego, takiego jak klawiatura i ekran komputera, więc jej wynik nie zależałby od zdolności maszyny do przekształcania słów w mowę. Test pionierski ten zapoczątkował historię stał się jednocześnie sposobem, który określa w jakim stopniu maszyna jest w stanie posługiwać się językiem naturalnym. (A. Turing, 1950, str. 433)

* + 1. ELIZA, PARRY

Szesnaście lat później1966, [Joseph Weizenbaum](https://en.wikipedia.org/wiki/Joseph_Weizenbaum) created a program which appeared to pass the Turing test. The program, known as [ELIZA](https://en.wikipedia.org/wiki/ELIZA), worked by examining a user's typed comments for keywords. If a keyword is found, a rule that transforms the user's comments is applied, and the resulting sentence is returned. If a keyword is not found, ELIZA responds either with a generic riposte or by repeating one of the earlier comments.[[28]](https://en.wikipedia.org/wiki/Turing_test#cite_note-FOOTNOTEWeizenbaum196637-28)

In addition, Weizenbaum developed ELIZA to replicate the behaviour of a [Rogerian psychotherapist](https://en.wikipedia.org/wiki/Person-centered_psychotherapy), allowing ELIZA to be "free to assume the pose of knowing almost nothing of the real world."[[29]](https://en.wikipedia.org/wiki/Turing_test#cite_note-FOOTNOTEWeizenbaum196642-29) With these techniques, Weizenbaum's program was able to fool some people into believing that they were talking to a real person, with some subjects being "very hard to convince that ELIZA [...] is *not* human."[[29]](https://en.wikipedia.org/wiki/Turing_test#cite_note-FOOTNOTEWeizenbaum196642-29) Thus, ELIZA is claimed by some to be one of the programs (perhaps the first) able to pass the Turing test,[[29]](https://en.wikipedia.org/wiki/Turing_test#cite_note-FOOTNOTEWeizenbaum196642-29)[[30]](https://en.wikipedia.org/wiki/Turing_test#cite_note-FOOTNOTEThomas1995112-30) even though this view is highly contentious (see [below](https://en.wikipedia.org/wiki/Turing_test#Naïveté_of_interrogators)).

[Kenneth Colby](https://en.wikipedia.org/wiki/Kenneth_Colby) created [PARRY](https://en.wikipedia.org/wiki/PARRY) in 1972, a program described as "ELIZA with attitude".[[31]](https://en.wikipedia.org/wiki/Turing_test#cite_note-FOOTNOTEBowden2006370-31) It attempted to model the behaviour of a [paranoid](https://en.wikipedia.org/wiki/Paranoia) [schizophrenic](https://en.wikipedia.org/wiki/Schizophrenic), using a similar (if more advanced) approach to that employed by Weizenbaum. To validate the work, PARRY was tested in the early 1970s using a variation of the Turing test. A group of experienced psychiatrists analysed a combination of real patients and computers running PARRY through [teleprinters](https://en.wikipedia.org/wiki/Teleprinter). Another group of 33 psychiatrists were shown transcripts of the conversations. The two groups were then asked to identify which of the "patients" were human and which were computer programs.[[32]](https://en.wikipedia.org/wiki/Turing_test#cite_note-FOOTNOTEColbyHilfWeberKraemer1972220-32) The psychiatrists were able to make the correct identification only 52 percent of the time – a figure consistent with random guessing.In the 21st century, versions of these programs (now known as "[chatbots](https://en.wikipedia.org/wiki/Chatbot)") continue to fool people. "CyberLover", a [malware](https://en.wikipedia.org/wiki/Malware) program, preys on Internet users by convincing them to "reveal information about their identities or to lead them to visit a web site that will deliver malicious content to their computers".[[33]](https://en.wikipedia.org/wiki/Turing_test#cite_note-33) The program has emerged as a "Valentine-risk" flirting with people "seeking relationships online in order to collect their personal data".[[34]](https://en.wikipedia.org/wiki/Turing_test#cite_note-34)

Common understanding has it that the purpose of the Turing test is not specifically to determine whether a computer is able to fool an interrogator into believing that it is a human, but rather whether a computer could *imitate* a human.

[Saygin 2000](https://en.wikipedia.org/wiki/Turing_test#CITEREFSaygin2000). Saygin, A. P.; Cicekli, I.; Akman, V. (2000), [*"Turing Test: 50 Years Later"*](http://crl.ucsd.edu/~saygin/papers/MMTT.pdf) *(PDF)*, Minds and Machines, **10** (4): 463–518,

1.2.3 SHRDLU

1.2.4 Siri, Cortana, Google Assistant, Amazon Alexa

1.3 Oprogramowanie do ustalania harmonogramów

1.3.1 Oprogramowanie do rezerwacji terminów

1.3.2 Oprogramowanie do tworzenia harmonogramów dla pracowników

1.4 Podsumowanie

2. Interakcja z Aestus

2.1 Rodzaje komunikacji

2.1.1 Człowiek z człowiekiem (międzyludzka)

2.1.2 Człowiek z maszyną

2.2.3 Komunikacja programu Aestus z człowiekiem

2.2 Program Aestus

2.2.1 Specyfikacje programu Aestus (z Agile: Requirements)

2.2.2 Aestus na tle klasycznych systemów dialogowych

2.3 Komunikacja programu Aestus z człowiekiem

2.4 Podsumowanie

3. Projekt Aestus

3.1 Agile

3.1.1 Czym jest Agile, Sprint

3.1.2 Projekt

3.1.2.1 Cel programu

3.1.2.2 Scenariusz i schemat blokowy

3.1.2.3 Automat skończenie stanowy

3.1.2.4 Użyty język programowania

3.1.2.4.1 Python 3.10

3.1.2.5 Użyte API

3.1.2.5.1 (TBD)

3.1.3 Opracowanie

3.1.3.1 Wykonanie i połączenie nagrań

3.1.3.1.1 Anotacja nagrań

3.1.3.1.2 Analiza statystyczna anotacji

3.1.4 Testowanie

3.1.4.1 Ankietowanie

3.1.4.1.1 Wersja testowa prgoramu Aestus

3.1.4.1.2 Interpretacja wyników

3.1.4.2 Debugowanie - spostrzeżenia

3.1.5 Wdrożenie

3.1.5.1 Wymagania Aestus (co ma zrobić)

3.1.5.2 Możliwości Aestus (co robi)

3.1.6 Przegląd

3.1.6.1 Mocne i słabe strony programu

3.2 Podsumowanie

4. Zakończenie

Załącznik 1: Kod programu

Załącznik 2: Instrukcja użytkowania

Załącznik 3: Przykładowy dialog

Załącznik 4: Ankieta dla testujących

Załącznik 5: Bibliografia

BIBLIOGRAFIA PROTOTYP:

ELIZA A Computer Program  
For the Study of Natural Language  
Communication Between Man  
And Machine  
Joseph Weizenbaum  
Massach.usclls [nshl-ute qf Tcchnu[ogg,\* Cambridge, Mass.

J a n u a r y . , 1966

Jan. 1966 pp 36–45

https://dl.acm.org/doi/10.1145/365153.365168

October,

M I N D  
A QUARTERLY REVIEW OF PSYCHOLOGY AND PHILOSOPHY  
I.—COMPUTING MACHINERY AND INTELLIGENCE BY A. M. TURING

<https://academic.oup.com/mind/article/LIX/236/433/986238>

(A. M. TURING 1950, str. 433)

Beynon-Davies, Paul (2003). Database Systems (3rd ed.). Palgrave Macmillan.

<https://books.google.pl/books?id=OSJIEAAAQBAJ&pg=PR10&dq=Beynon-Davies,+Paul+(2003).+Database+Systems+(3rd+ed.).+Palgrave+Macmillan.&hl=en&sa=X&ved=2ahUKEwj2yvOUqrn3AhWMv4sKHXkTAf4Q6AF6BAgHEAI#v=onepage&q&f=false>

(Beynon-Davies, 2004, str. 6)

Suket Arora1, Kamaljeet Batra2, Sarabjit Singh

<https://arxiv.org/ftp/arxiv/papers/1306/1306.4134.pdf>

(Lawrynowicz, 2019, str. 4)

<http://www.cs.put.poznan.pl/alawrynowicz/PJN_9_chatbot.pdf>

(J. C. R. Licklider, 1960)

<http://worrydream.com/refs/Licklider%20-%20Man-Computer%20Symbiosis.pdf>

(Sergey Brin, Lawrence Page, The anatomy of a large-scale hypertextual Web search engine, 1998)

<http://infolab.stanford.edu/pub/papers/google.pdf>

(Kułakowska, 2015 ,str 8.)

<https://home.agh.edu.pl/~horzyk/pracedyplom/2015%20IzabelaKu%c5%82akowskaPracaMagisterska.pdf>

(Wikipiedia)

<https://en.wikipedia.org/wiki/Dialogue_system>

Bachan (2011, str. 34)

<http://repozytorium.amu.edu.pl:8080/bitstream/10593/1089/1/Bachan_Jolanta_Rozprawa-doktorska.pdf>

(Gibbon et al. 1997)

Gibbon, D., Moore, R. & Winski, R. 1997. Handbook of Standards and Resources for Spoken Language Systems. Berlin: Mouton de Gruyter

Daniel Jurafsky Stanford University James H. Martin  
University of Colorado at Boulder

<https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf>

Sprawdzić ZA – to do źródła źródła

Spis rysunków i tabel

Rysunek 1: Schemat architektury systemu dialogowego

Rysunek 2: Wykonywanie nagrań

Rysunek 3: Anotacja nagrań

Rysunek 4: Analiza statystyczna anotacji

Rysunek 5: Automat skończenie stanowy

Rysunek 6-n: Przykładowe dialogi

Tabela 1: Wyniki ankiet

https://sentione.com/pl/zasoby/slowniki/chatboty

Ciekawe:

***Proof of concept***

**Czym jest proof of concept?**  
Proof of concept to minimalne wdrożenie produktu, zawierające tylko niezbędną funkcjonalność. Tworzy się je w celach demonstracyjnych – aby prezentować produkt potencjalnym inwestorom lub klientom.

***Test Turinga***

**Czym jest test Turinga?**  
Test Turinga to eksperyment mający na celu sprawdzenie, czy dana maszyna posiada umiejętność wykazywania inteligentnego, ludzkiego zachowania. Eksperyment uznaje się za udany, jeśli ludzki obserwator nie jest w stanie odróżnić zachowania maszyny od zachowania człowieka wykonującego to samo zadanie. Test nazwany jest nazwiskiem jego wynalazcy, Alana Turinga.

***Trigger***

**Czym jest trigger?**  
Trigger to dowolna fraza która wywołuje konkretny proces lub reakcję ze strony chatbota. Często spotykanym przykładem triggera jest prośba o przełączenie obsługi, która sprawia, że bot przekazuje sprawę agentowi obsługi klienta.

***User experience***

**Czym jest user experience?**  
User experience to termin którym określamy wypadkową wszystkich interakcji użytkownika z systemem – i reakcji emocjonalnych użytkownika na te interakcje. Projektanci user experience dokonują wszelkich starań, aby systemy były intuicyjne i łatwe do zrozumienia.

***Wdrożenie***

**Czym jest wdrożenie?**  
Wdrożenie to uruchomienie produktu lub usługi. Wdrożona usługa często nazywana jest “produkcją” (w odróżnieniu od usługi “testowej” lub “w opracowaniu”).

***Wirtualny agent***

**Czym jest wirtualny agent?**  
Wirtualny agent to chatbot zaprojektowany do obsługi klienta. Zazwyczaj zajmują się one najbardziej nużącymi i najczęściej powtarzającymi się zadaniami, których konsumenci oczekują od obsługi klienta: od sprawdzania stanu konta do śledzenia paczek.

***Wypowiedź***

**Czym jest wypowiedź?**  
Wypowiedź to pojedyncze zdanie wypowiedziane przez użytkownika. Wypowiedzi analizowane są przez chatboty aby określić ich zamiar, temat oraz nacechowanie emocjonalne. Dzięki temu mogą łatwiej zrozumieć kontekst rozmowy i udzielać właściwych odpowiedzi.

Diagram

Description automatically generated

Ditched:

It is played with three people, a man, a woman, and an interrogator who may be of either sex. The interrogator stays in a room apart from the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either ‘X is A and Y is B’ or ‘X is B and Y is A’. The  
interrogator is allowed to put questions to A and B. Now suppose X is actually A, then A must answer. It is A’s object in the game to try and cause C to make the wrong identification. We now ask the question, ‘What will happen when a machine takes the part of A in this game?’ Will the interrogator decide wrongly as often  
when the game is played like this as he does when the game is played  
between a man and a woman? Therefore, a question arose, ‘Can machines think?’. The conversation would be limited to a text-only channel such as a computer keyboard and screen so the result would not depend on the machine's ability to render words as speech. If the evaluator cannot reliably tell the machine from the human, the machine is said to have passed the test.

Kolejnym przykładem może być umożliwienie dostępu do internetu dla osób niedowidzących bądź niewidomych za pomocą rozpoznawania głosu, możliwośc używania swojego smartfona za pomocą komend głosowych otwiera przed osobami niepełnosprawnymi zupełnie nowy świat i samodzielny dostęp do informacji, do których reszta świata ma dostęp na wyciągnięcie ręki.

1. Systemy dialogowe
   1. Czym są systemy dialogowe.

1.1.2 Czym jest dialog, wypowiedź, intencja, tura

1.1.2 Typy Systemów dialogowych

1.1.3 Schematy systemów dialogowych

1.1.4 Jak zbudowane są systemy dialogowe

Text

Description automatically generated with medium confidence

1.1.5 Zastosowania systemów dialogowych

1.2 Historia systemów dialogowych

1.2.1 Imitation game Alana Turinga

1.2.2 DARPA, Georgetown-IBM

1.2.2 ELIZA, PARRY

1.2.3 SHRDLU

1.2.4 Siri, Cortana, Google Assistant, Amazon Alexa

1.3 Oprogramowanie do ustalania harmonogramów

1.3.1 Oprogramowanie do rezerwacji terminów

1.3.2 Oprogramowanie do tworzenia harmonogramów dla pracowników

1.4 Podsumowanie

2. Interakcja z Aestus

2.1 Rodzaje komunikacji

2.1.1 Człowiek z człowiekiem (międzyludzka)

2.1.2 Człowiek z maszyną

2.2.3 Komunikacja programu Aestus z człowiekiem

2.2 Program Aestus

2.2.1 Specyfikacje programu Aestus (z Agile: Requirements)

2.2.2 Aestus na tle klasycznych systemów dialogowych

2.3 Komunikacja programu Aestus z człowiekiem

2.4 Podsumowanie

3. Projekt Aestus

3.1 Agile

3.1.1 Czym jest Agile, Sprint

3.1.2 Projekt

3.1.2.1 Cel programu

3.1.2.2 Scenariusz i schemat blokowy

3.1.2.3 Automat skończenie stanowy

3.1.2.4 Użyty język programowania

3.1.2.4.1 Python 3.10

3.1.2.5 Użyte API

3.1.2.5.1 (TBD)

3.1.3 Opracowanie

3.1.3.1 Wykonanie i połączenie nagrań

3.1.3.1.1 Anotacja nagrań

3.1.3.1.2 Analiza statystyczna anotacji

3.1.4 Testowanie

3.1.4.1 Ankietowanie

3.1.4.1.1 Wersja testowa prgoramu Aestus

3.1.4.1.2 Interpretacja wyników

3.1.4.2 Debugowanie - spostrzeżenia

3.1.5 Wdrożenie

3.1.5.1 Wymagania Aestus (co ma zrobić)

3.1.5.2 Możliwości Aestus (co robi)

3.1.6 Przegląd

3.1.6.1 Mocne i słabe strony programu

3.2 Podsumowanie

4. Zakończenie

Załącznik 1: Kod programu

Załącznik 2: Instrukcja użytkowania

Załącznik 3: Przykładowy dialog

Załącznik 4: Ankieta dla testujących

Załącznik 5: Bibliografia

Spis rysunków i tabel

Rysunek 1: Schemat architektury systemu dialogowego

Rysunek 2: Wykonywanie nagrań

Rysunek 3: Anotacja nagrań

Rysunek 4: Analiza statystyczna anotacji

Rysunek 5: Automat skończenie stanowy

Rysunek 6-n: Przykładowe dialogi

Tabela 1: Wyniki ankiet