

Dokumentacja inżynierii wymagań

Członkowie zespołu:

1. Macierz kompetencji zespołu.

Kompetencje	Paweł Kusz	Tymoteusz Mróz	Michał Pędziwiatr	Mykyta Boyko
Backend	Nie posiada	Nie posiada	Posiada	Posiada
Frontend	Posiada	Posiada	Nie posiada	Nie posiada
Testing	Posiada	Posiada	Posiada	Posiada
Communication	Nie posiada	Nie posiada	Nie posiada	Posiada

2. Szczegóły projektu.

Pytanie	Odpowiedź	Uwagi
Jaki jest limit czasu na przedstawienie propozycji aplikacji?	Trzy tygodnie.	Czas jest ograniczony do czasu zajęć.
Na jakiej platformie aplikacja ma zostać zrealizowana?	Na platformie webowej (jako aplikacja internetowa).	Aplikacja ma być dostępna zarówno na urządzeniach stacjonarnych, jak i mobilnych.
Jaka technologia ma zostać użyta do generowania odpowiedzi?	Sztuczna Inteligencja – modele open-source lub małopłatne.	Prawdopodobnie zostanie użyty model GPT-4o od OpenAI.
W jakim języku aplikacja ma generować odpowiedzi?	W języku polskim.	–
W jaki sposób użytkownik końcowy będzie komunikował się z aplikacją?	Za pomocą wiadomości tekstowych (czat) oraz mowy.	Tekst + Mowa – aplikacja będzie wymagała dostępu do mikrofonu.
Jaki stos technologiczny ma zostać zastosowany?	Dowolny stos technologiczny.	–
Jak ma wyglądać GUI (interfejs graficzny użytkownika)?	Ma być kolorowy, przyjazny, pozwalać na obsługę czatu i mowy.	Aplikacja jest przeznaczona dla dzieci – interfejs prosty, nieskomplikowany.
Jaki powinien być maksymalny czas odpowiedzi aplikacji?	Maksymalnie pięć sekund.	Będzie wymagany odpowiedni model SL.

Wszystkie wymagania są możliwe do spełnienia? **TAK**

3. Ustalony format danych.

Nazwa	Typ	Właściwości	Opis
text_prompt	String	typ wbudowany tekstowy, obsługa znaków specjalnych	Wysyłany tekst wejściowy.
audio_prompt	File	typ plikowy, nagranie dwukanałowe, wysyłany w całości	Wysyłany plik wejściowy audio.
text_response	String	typ wbudowany tekstowy, obsługa znaków specjalnych	Odbierany tekst wyjściowy.
audio_response	Stream	typ strumieniowy, odbieranie fragmentami	Odbierany strumień wyjściowy audio.

4. Przedstawienie modelowanego systemu.

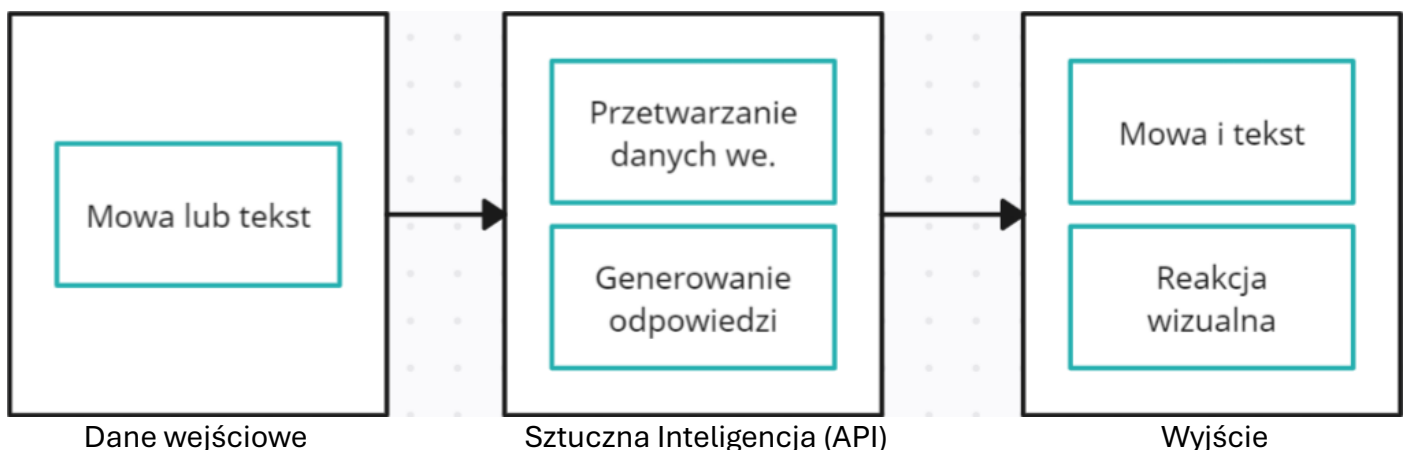
1. W formie tabeli:

Aktorzy	Sztuczna Inteligencja, Użytkownik końcowy
Opis	Celem działania aplikacji jest udzielanie prostych odpowiedzi na pytania otrzymane od użytkownika końcowego poprzez Sztuczną Inteligencję (z zachowaniem kontekstu w uzasadnionych przypadkach).
Dane	Wejściowe – mowa lub tekst. Wyjściowe – mowa i tekst.
Wyzwalacz	Ukończenie przekazywania pliku mowy lub całego tekstu do API.
Odpowiedź	Pochodzi z API – obiekt zawierający plik mowy oraz tekst z odpowiedzią na zadane pytanie w danych wejściowych.
Uwagi	Przekazywany plik mowy lub tekst nie mogą przekraczać rozmiaru maksymalnego podanego w specyfikacji API wybranego modelu SI. Wymagany dostęp aplikacji do Internetu oraz mikrofonu.

2. W formie diagramu UML:



5. Projekt architektury aplikacji.



Każdy z trzech dużych prostokątów w diagramie przedstawia kluczowe komponenty systemu. System działa w sposób, w którym dane wejściowe w formie mowy lub tekstu są przetwarzane przez sztuczną Inteligencję (dostęp do modelu następuje poprzez API) w celu wygenerowania odpowiedzi. Pierwszy element systemu to dane wejściowe, które są kluczowe do zainicjowania procesu. Drugim komponentem jest sztuczna inteligencja, która zajmuje się analizą danych, przetwarzaniem oraz generowaniem odpowiedzi. Na końcu, system dostarcza wyjście w formie mowy, tekstu lub reakcji wizualnej, stanowiąc finalny etap procesu interakcji z systemem.

6. Sugerowany język implementacji: Flask (Python) + React (JavaScript).

Uzasadnienie:

Użycie Flask i React w projekcie zapewnia elastyczność oraz prostotę zarówno w warstwie backendu, jak i frontendowej. Flask, będący mikroframeworkiem, pozwala na szybkie tworzenie lekkiego API, jest łatwy w użyciu i dobrze integruje się z popularnymi bibliotekami do obsługi baz danych czy autoryzacji. Z kolei React umożliwia tworzenie dynamicznych interfejsów użytkownika z wykorzystaniem komponentów, co ułatwia utrzymanie i rozbudowę aplikacji. Oba narzędzia są dobrze skalowalne, mają aktywne społeczności oraz wsparcie, co sprawia, że są efektywne w realizacji projektów webowych.
