**Dokumentacja Projektu: Aplikacja Webowa do Rozmowy z Chatem**

**1.Opis Projektu**

**Celem projektu jest stworzenie aplikacji webowej, która umożliwia użytkownikom rozmowę z chatbotem w wybranym przez nich języku.**

**2.Macierz kompetencji**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kompetencje** | **Paweł Piwowarczyk** | **Piotr Stasiak** | **Aleksy Dąda** | **Kamil Niechaj** |
| **Programowanie C/C++** | **Podstawowe** | **Podstawowe** | **Podstawowe** | **Podstawowe** |
| **Programowanie Python** | **Podstawowe** | **Podstawowe** | **Podstawowe** | **Posiada** |
| **React** | **Brak** | **Posiada** | **Zaawansowane** | **Posiada** |
| **Testowanie Oprogramowania** | **Brak** | **Brak** | **Brak** | **Brak** |
| **Node.js** | **Podstawowe** | **Podstawowe** | **Zaawansowane** | **Podstawowe** |

**3. Uszczegółowienie zadanego projektu**

|  |  |  |
| --- | --- | --- |
| Pytanie | Odpowiedz | Uwagi |
| Jaki typ aplikacji trzeba stworzyć? | Aplikacja webowa |  |
| Jakich języków można się na niej uczyć? | Języków taki jak Angielski, Hiszpański i Polski |  |
| Jak ma wyglądać działanie aplikacji( co może robić użytkownik)? | Polegać ma ona na rozmowie z botem w obcym języku | Chodzi o język angielski, hiszpański albo polski |
| Jaki może być input danych od użytkownika? | Aplikacja będzie przyjmować teks napisany na czacie albo mowę z mikrofonu |  |
| Z czego korzystamy | Korzystamy z COTS |  |
| Czy użytkownik może tylko rozmawiać z botem? | Użytkownik może również prosić o poprawę błędów w swoich wypowiedziach |  |
| Jakie ma być GUI? | GUI ma być proste, przejrzyste i intuicyjne |  |
| Jaki ma być output z aplikacji? | Ma być to tekst lub mowa |  |
| Jaki ma być maksymalny czas odpowiedzi bota | Ma być to 10s |  |
| Wymagania sprzętowe | Webbroswer nie starszy niż 6 miesięcy |  |

4. **Technologie**

**4.1 Frontend**

* **React**: Aplikacja korzysta z Reacta do stworzenia interaktywnego interfejsu użytkownika.
* **Kompatybilność przeglądarek**: Wspiera najnowsze wersje przeglądarek (wydane w ciągu ostatnich 6 miesięcy).

**4.2 Backend**

* **Node.js**: Backend jest zbudowany na bazie **Node.js’a**.
* **API**: Backend zapewnia API do obsługi wprowadzanych danych oraz generowania odpowiedzi.
* **Rozpoznawanie i synteza mowy**: Integracja z zewnętrznymi usługami rozpoznawania mowy oraz syntezy głosu w trzech językach.

5.**Architektura**

**5.1 Architektura frontendu**

* **Komponenty React**: Aplikacja jest podzielona na modułowe komponenty React, umożliwiające elastyczne zarządzanie interfejsem użytkownika.
* **Moduły wprowadzania danych**: Komponenty umożliwiają obsługę tekstowego i głosowego inputu.

**5.2 Architektura backendu**

* **REST API**: API odpowiada za odbieranie danych, integrację z serwisami rozpoznawania mowy oraz zwracanie odpowiedzi w formie tekstowej i głosowej.
* **Serwer w Node.js**

6. Diagram Przepływu Danych

Obraz zawierający tekst, zrzut ekranu, diagram, design

Opis wygenerowany automatycznie

7. Diagram Komponentów

Obraz zawierający tekst, zrzut ekranu, design

Opis wygenerowany automatycznie

8.**Podział pracy**

* **Frontend**: Aleksy Dąda
* **Backend**: Kamil Niechaj, Paweł Piwowarczyk, Piotr Stasiak

9. Uzasadnienie wybranych technologii

Wszyscy członkowie naszego zespołu maja doświadczenie w pracy w języku **Node.js** jak i niektórzy mają duże doświadczenie w pracy z Reactem.