***EduSkan***

***Dokumentacja Techniczna***

Spis treści:

1.[Ogólne informacje o aplikacji](#_Ogólne_informacje_1)

2. [Wymagania aplikacji](#_Wymagania_aplikacji)

3.[Działające funkcjonalności](#_Działające_funkcjonalności)

a) [Dla ucznia](#_Wyszukiwanie_szkół:)

b) [Dla szkół](#_Dodawanie_szkół:)

4.[Architektura](#_Architektura_aplikacji)

5.[Technologie i narzędzia](#_Technologie_i_narzędzia:)

6.[UI/UX](#_UI/UX)

7.[Funkcjonalności do zaimplementowania](#_Funkcjonalności_do_zaimplementowani)

8.[Instrukcja obsługi](#_Instrukcja_obsługi)

# **Ogólne informacje**

Projekt **EduSkan** to inicjatywa mająca na celu pomóc uczniom szkół podstawowych oraz ponadpodstawowych w wyborze odpowiedniej szkoły/uczelni i kierunku kształcenia. Pomysł na aplikację powstał ze względu na to, że dostępne w Internecie aplikacje na ogół wyszukują tylko podstawowe dane teleadresowe szkoły i adres strony www. Wyszukiwarki te nie uwzględniają potrzeb edukacyjnych uczniów, ich słabych i mocnych stron oraz zainteresowań wykraczających poza skostniały podział na szkolne przedmioty. W dodatku są niekompletne i często nieaktualne. Jedna z istniejących wyszukiwarek oferuje filtrowanie szkół za pomocą rozbudowanych formularzy i wpisywania bardzo szczegółowych informacji (np. ocen z poszczególnych przedmiotów). Tego typu aplikacja działa bardzo schematycznie i uwzględnia tylko elementy przewidziane „na sztywno” w formularzu.

Z wyszukiwarką **EduSkan** użytkownik porozumiewa się językiem naturalnym, wprowadzając odpowiedzi własnymi słowami, aniżeli wybierając filtry. Jest to możliwe dzięki zastosowaniu sztucznej inteligencji. Użytkownik nie określa z góry typu szkoły/uczelni oraz profilu/wydziału. Odpowiada na pytania aplikacji, a wyszukiwarka zwraca gotową rekomendację.

**Przykład**:

W klasycznej wyszukiwarce użytkownik wstępnie określa typ i profil szkoły, sugerując się często oczekiwaniami rodziców lub wyborem przyjaciół (np.: liceum ogólnokształcące, profil humanistyczny). Natomiast **EduSkan** nie bierze pod uwagę presji otoczenia, tylko skupia się na rzeczywistych potrzebach ucznia w oparciu o jego mocne i słabe strony, zainteresowania i oczekiwania względem przyszłej szkoły. W rezultacie, zamiast liceum o profilu humanistycznym, uczeń otrzyma sugestię, że lepszym dla niego wyborem byłaby np. szkoła branżowa w zawodzie fryzjer.

**EduSkan** jest elastycznym systemem pozwalającym każdej zainteresowanej szkole/uczelni dodać swoją ofertę do naszej wyszukiwarki. Gotowa baza uwzględnia różne typy placówek edukacyjnych:

- Szkoły branżowe

- Technika

- Licea ogólnokształcące

- Politechniki

- Uniwersytety

- Pozaszkolne placówki edukacyjne

Głównym odbiorcą są uczniowie klas ósmych szkoły podstawowej, którzy stoją przed wyborem szkoły średniej oraz maturzyści podejmujący decyzję o wyborze uczelni wyższej. Nasza aplikacja oferuje narzędzia do dopasowania szkoły na podstawie otwartego zestawu kryteriów, a nie ograniczonej, wstępnie zaimplementowanej ich listy. Również szkoły/uczelnie dodając się do bazy aplikacji, mogą uwzględniać w swojej charakterystyce wszelkie formy działania, aktywności, inicjatywy itp., które wymykają się utartym schematom. Rolą algorytmów AI jest wydobycie z profilu ucznia i profilu szkoły/uczelni kluczowych informacji i dopasowanie ich do siebie.

Ponieważ aplikacja jest agile’owa, uwzględnia feedback użytkowników. Dzięki temu developerzy mogą modyfikować wyszukiwarkę, dodając lub zmieniając funkcjonalności zgodnie z oczekiwaniami użytkowników.

Jednym z naszych celów jest również to, aby aplikacja była przydatna studentom po licencjacie oraz poszukującym studiów podyplomowych.

Aby zwiększyć wiarygodność wyszukiwarki i zrównoważyć potencjalnie zbyt przekoloryzowane opisy szkół/uczelni o sobie, szybko zostaną wprowadzone ankiety ewaluacyjne dla uczniów/studentów/ /absolwentów.

Osobną grupę stanowią uczniowie zainteresowani zajęciami w placówkach pozaszkolnych. W przyszłości planujemy rozbudować aplikację o dodatkowe funkcje, takie jak branie pod uwagę opinii uczniów/absolwentów określonych placówek, w celu uniknięcia sfałszowania reputacji szkoły poprzez formularz – będzie to działać na zasadzie wag, gdzie sztuczna inteligencja sama uśredni statystyki szkoły, łącząc różne źródła.

# **Wymagania aplikacji**

## Serwer:

Dla prawidłowego działania aplikacji serwer powinien być skonfigurowany w następujący sposób:

-Python 3.11 lub wyższy

-Pip v23.2.1

Biblioteki:

-langchain

-openai

-chromadb

-tiktoken

-unstructured

-Node v.12.22.9

-Npm v.8.5.1

## Klient:

Aby wyszukiwarka zwracała jak najbardziej dopasowane wyniki, każda szkoła i uczelnia powinny co roku przed rozpoczęciem naboru aktualizować dane na swój temat (kierunki i profile kształcenia, zajęcia pozalekcyjne, koła naukowe, wolontariat, zajęcia sportowe, itp.).

Instytucja zamawiające - govtech.pl. we współpracy z Ministerstwem Edukacji i Nauki – powinna wprowadzić obowiązek aktualizacji danych przez palcówki edukacyjne.

# **Działające funkcjonalności**

## Wyszukiwanie szkół:

Aplikacja działa poprzez 2 warstwy algorytmów sztucznej inteligencji. Pierwsza warstwa określa jaki **typ szkoły** pasuje dla danego zapytania, a następnie dzięki temu dobiera odpowiednią bazę danych z której ma wybrać szkołę. Usprawnia to czas poświęcony na obliczenia, polepsza jakość odpowiedzi, gdyż bot nie ma potrzeby analizować zbyt dużej ilości placówek oraz zmniejsza koszty zapytań do API, gdyż ma mniej informacji do analizy.

Aplikacja uwzględnia wszystkie informacje wprowadzone przez użytkownika w odpowiednie pola tekstowe, które są podane na poniższym schemacie. Dla każdego rodzaju placówki pytania są różne, aby lepiej określić profil użytkownika.



Po naciśnięciu przycisku „Skanuj” prześlesz nam swoje dane które wprowadziłeś w pola tekstowe, a my postaramy się dobrać dla ciebie idealną szkołę oraz wyświetlimy ci przykłady szkół które do ciebie pasują

Użytkownik ma opisać siebie w dość dokładny sposób. Im więcej napiszesz w tym polu tym lepiej sztuczna inteligencja będzie mogła dopasować profil użytkownika. W tym polu opisz to co według ciebie jest ważne w twojej osobowości itp.

Użytkownik ma podać czym się interesuje np.:(ornitologia)

Użytkownik ma podać swoje oceny np.:()

Użytkownik ma podać czym planuje zajmować się w przyszłości np.: (programować, malować)

Warto podkreślić, że dane, które użytkownik przekazuje, są używane zaledwie jednokrotnie, aby dostosować profil szkoły lub kierunku do Twoich potrzeb. Nie są one w żaden sposób archiwizowane ani nie są nikomu udostępniane, co zapewnia prywatność każdemu użytkownikowi przestrzegając wszystkich regulacji związanych z RODO.

## Dodawanie szkół:

Proces dodawania szkół wymaga interwencji ludzkiej, ponieważ szkoła, która chce dołączyć do naszej bazy danych, musi najpierw przesłać zgłoszenie, które oczekuje na weryfikację. Dopiero po pomyślnej weryfikacji dane szkoły są przenoszone do głównej bazy szkół. W przyszłości planujemy zautomatyzować ten proces.

Warto podkreślić, że dane przesyłane przez placówki są gromadzone i przechowywane w naszej bazie, do której nikt z zewnątrz nie ma dostępu.

## Ocenianie szkół:

Każdy absolwent danej szkoły będzie miał możliwość oceny swojej placówki na podstawie własnej opinii. Będzie mógł wyrazić swoje preferencje, podkreślić aspekty, które uważa za wartościowe, a także zwrócić uwagę na obszary, które jego zdaniem mogłyby być udoskonalone. Ta funkcja pozwoli lepiej dostosować szkołę do kandydata.

# **Architektura aplikacji**

Aplikacja opiera się na architekturze SPA (Single-Page Application) - jest to nowoczesna koncepcja architektury, która została wybrana ze względu na jej prostotę obsługi od strony deweloperów oraz przyjemność użytkowania. W architekturze SPA cała interakcja odbywa się na jednej stronie, co skraca czasy oczekiwania i eliminuje potrzebę wczytywania kolejnych stron.

Warstwy aplikacji:

  
1. Interfejs użytkownika (UI) 🡪 Interfejs pełni kluczową rolę w aplikacji, zajmując się takimi zadaniami jak:

- Pobieranie danych od użytkownika;

- Wyświetlanie odpowiedzi;

- Gromadzenie informacji od szkół zainteresowanych dołączeniem do projektu;

- Informowanie użytkownika o podstawowych informacjach dotyczących aplikacji, takich jak informacje o autorach, celach oraz używanych technologiach.

2. Serwer (Backend) 🡪 Serwer to centralna część aplikacji, która odpowiada między innymi za:

- Przetwarzanie danych dostarczonych przez interfejs użytkownika;

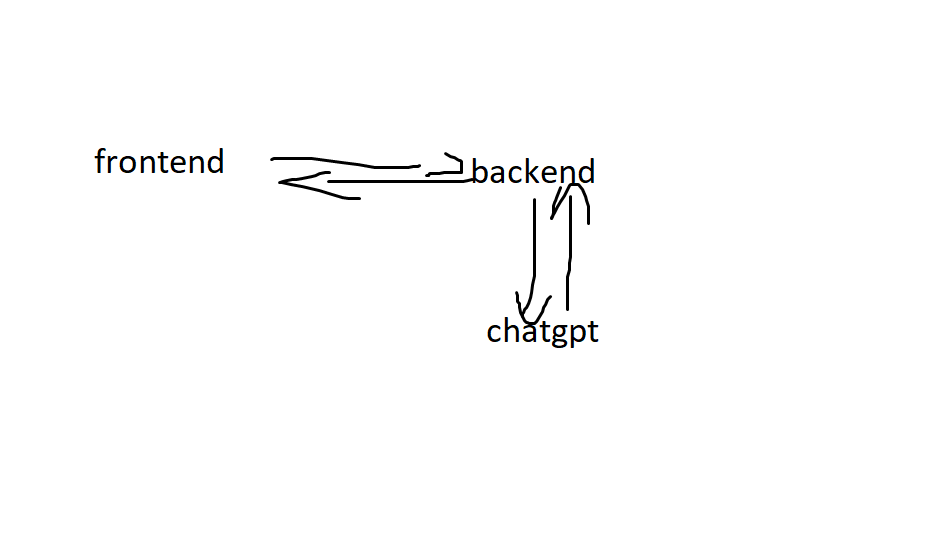
- Współpraca z API ChatGPT, przekazując mu informacje o użytkownikach i odbierając odpowiedzi;

- Przechowywanie danych szkół uczestniczących w projekcie;

- Przesyłanie odpowiedzi do wyświetlenia przez UI.

3. Interfejs Programistyczny Aplikacji (API) 🡪 Ta warstwa odpowiada za zapewnienie połączenia między wszystkimi komponentami aplikacji, umożliwiając im efektywną komunikację między sobą. Jest kluczowym elementem zapewniającym integralność i funkcjonalność aplikacji.

## Model i przepływ danych:



Aplikacja gromadzi informacje o szkołach w pliku .txt. Plik tekstowy służy jako baza danych. API zaczyna od wyboru typu szkoły, a dopiero później, na podstawie typu, wybiera bazę, z której będzie będzie wybierać szkołę. Odpowiedź jest przesyłana w postaci JSON z powrotem do wyświetlenia na stronie.

## Bezpieczeństwo:

Dbamy o właściwy format wprowadzanych treści formatując wszystko m.in. usuwając niepotrzebne znaki, ale nie uniemożliwia to użytkownikowi wpisać dane w jakiejkolwiek formie sobie życzy dla łatwiejszego korzystania. Nasza aplikacja jest także zabezpieczona przed spamem. W sytuacji, gdy wprowadzone treści są skrajnie nieodpowiednie, są dopuszczone do chatu, ale zostanie wyświetlona informacja o tym, że na wprowadzonych danych nie można wybrać placówki edukacyjnej.

## Skalowalność:

Skalowalność bazy danych jest obecnie standardem, który ten projekt spełnia. W każdym momencie dowolna placówka może dołączyć do naszej bazy szkół lub zmodyfikować wcześniejszy wpis, wypełniając odpowiedni formularz. Działa to również w drugą stronę.

Warto jednak podkreślić, że skalowalność to nie tylko obsługa obciążenia, ale także zdolność do obsługi małych zmian. Nasza aplikacja jest przystosowana do obsługi drobnych modyfikacji jak i całkowitej zmiany sposobu podejmowania decyzji przez ChatGPT – wystarczy „powiedzieć” mu co ma robić bez potrzeby programowania.

# **Technologie i narzędzia:**

Technologie, których użyliśmy do stworzenia naszej aplikacji, to:

a) UI:

- HTML5,

- CSS3,

- JavaScript.

b) Serwer:

- Python3,

- Node.js (z wykorzystaniem frameworka Express).

c) Analiza danych:

- ChatGPT API 3.5 Turbo.

Warto również wspomnieć o bibliotece, którą wykorzystujemy w naszej aplikacji, a mianowicie Langchain.

**Langchain** to potoczne określenie oznaczające zazwyczaj sieć blockchain, która jest wykorzystywana do zarządzania i śledzenia danych językowych. W edukacji EduSkan służy do tego, aby zmusić ChatGPT do wyciągania informacji z bazy danych, aniżeli jego ogólnej wiedzy.

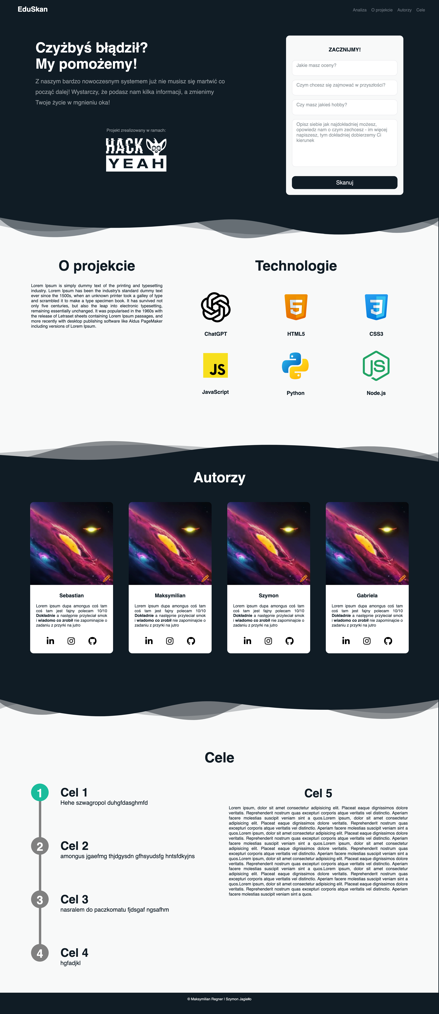
Dane są przesyłane do serwera w formularzu za pomocą protokołu HTTP POST

# **UI/UX**

## Interfejs ucznia:

Interfejs zaprojektowany z myślą o prostocie użytku. Dlatego kierowaliśmy się oraz dostarczaniem niezbędnych informacji w jak najbardziej czytelny sposób. Dzięki takiemu rozwiązaniu, proces wprowadzania danych jest intuicyjny dla każdego użytkownika.

Po otrzymaniu wyniku, konieczne jest wyświetlenie go użytkownikowi. Do tego celu wykorzystywane jest okno popup, które pojawia się z odpowiedź.



Menu

Cele projektu wraz z oznacznikiem

Sekcja autorzy

Sekcja o projekcie

Formularz

Aplikacja jest w pełni responsywna, co oznacza, że każdy użytkownik może z niej korzystać niezależnie od rodzaju urządzenia, na którym pracuje.

W przypadku błędu, aplikacja informuje użytkownika, wyświetlając odpowiednie komunikaty, np. "Nieprawidłowo wprowadzone dane".

# 

# **Funkcjonalności do zaimplementowania**

**Tworzenie raportów i statystyk 🡪** na podstawnie wydawanych uczniom i studentom rekomendacji władze oświatowe, samorządowe organy prowadzące oraz szkoły i uczelnie, otrzymają informację o tym, jak często dana szkoła lub uczelnia (w rozbiciu na kierunki, profile itp.) pojawiła się w rekomendacjach wydawanych przez wyszukiwarkę. Z uwagi na to, że dane te pojawią się w systemie przed lub na początku procesu rekrutacji, możliwe będzie dostosowanie ilości miejsc na danym profilu, kierunku, wydziale.

**Ankiety ewaluacyjne dla uczniów, studentów i absolwentów 🡪** w których będzie można ocenić swoją szkołę. Tego typu opinie pozwolą wyszukiwarce generować rekomendacje dla uczniów/studentów w bardziej obiektywny sposób.

**Współpraca z rankingiem „Perspektywy” 🡪** pobieranie rankingu szkół, który usprawni wybieranie placówek dodając dodatkowe spojrzenie na daną placówkę przez istniejący renomowany system oceniania

**Baza rekomendacji wygenerowanych przez GPT 🡪**  na jej podstawie instytucja zamawiająca i MEiN oraz szkoły i uczelnie otrzymają informację jak często w wygenerowanych uczniom rekomendacjach wybrana została konkretna szkoła lub uczelnia z podziałem na profile/kierunki.

**Baza zawierająca odpowiedzi uczniów na pytania zadane im przez wyszukiwarkę 🡪** dane, które uczniowie podają podczas wyszukiwania szkoły/uczelni, zawierają istotne informację dotyczące wyborów edukacyjnych uczniów, ich zainteresowań oraz słabych i mocnych stron.Na tej podstawieMEiN i instytucje zajmujące się badaniem oświaty mogą stwierdzić jaki jest stopień dopasowania oferty edukacyjnej w Polsce do rzeczywistych potrzeb uczniów. Natomiast socjolodzy, psycholodzy mogą na podstawie analizy porównawczej określić tendencje w ewoluowaniu preferencji edukacyjnych uczniów i studentów, ich kompetencji i zainteresowań.

**Wprowadzenie trybu dla niepełnosprawnych 🡪** który wprowadza m.in.:

- Powiększanie i pomniejszanie tekstu

- Zmianę kontrastu

- Skalę szarości

Użytkownik będzie mógł włączyć tryb poprzez ikonkę na górze strony

# 

# **Instrukcja obsługi**

Jeśli po przejrzeniu całej dokumentacji masz trudności z zrozumieniem, jak korzystać z aplikacji, lub po prostu chciałbyś szybko poznać jej działanie, poniżej znajdziesz graficzne instrukcje, które krok po kroku przedstawiają, jak korzystać z aplikacji. Każdy ekran jest opatrzony objaśnieniem, które kieruje, co należy zrobić, aby wszystko działało tak, jak powinno.

Tu instrukcja.