

1 Krátky popis

Student: Anastasiya Ihnatovich (ihnatovich1@uniba.sk)

Meno skolitela: Pavel Petrovič (pavel.petrovic@fmph.uniba.sk)

Cieľ projektu: Vyvinúť personalizovaný systém plánovania cestovných trás, ktorý odporúča optimálne trasy na základe východiskového a cieľového bodu používateľa, spolu s presnými dátumami odchodu a príchodu. Systém bude zohľadňovať rôzne faktory, vrátane záujmov používateľa, cestovných skúseností, preferovaných spôsobov dopravy a ďalších kritérií, ktoré ovplyvňujú zážitok z cestovania.

Cieľová skupina: Turisti

Repozitar: https://github.com/nignatovichhh/rocnikovy_projekt

Fázy zimného semestra:

1. Komunikácia s OpenAI API
2. Komunikácia s Tripadvisor API
3. Komunikácia s mapami
4. Počiatočný návrh systémovej architektúry a výber technológií
5. Vývoj 4o-modelového promptu na získanie odporúčaní

2 Počiatočný návrh systémovej architektúry a výber technológií

1. Frontend: React + TypeScript
2. Backend: Python (Flask, requests)
3. Integrácie: OpenAI API, TripAdvisorAPI, Mapbox GL JS

3 Komunikácia s OpenAI API

Na odosielanie promptov som vytvorila triedu OpenAIApiWrapper, ktorá používa knižnicu openai. Hlavné časti:

```
def __init__(self):
    # Client initialization
    self.client = OpenAI(
        api_key=OPENAI_API_KEY,
    )

    # Reading prompt from file
    with open('./apis/context_prompt.txt', 'r', encoding='utf-8') as file:
        self.context_prompt = file.read()

def send_request(self, query):
```

```

# ...

# Sending prompt to OpenAI API
chat_completion = self.client.chat.completions.create(
    messages=[
        {
            "role": "user",
            "content": query,
        }
    ],
    model="gpt-4o",
)

# Retrieving response
lines = chat_completion.choices[0].message.content.split('\n')
# ...

```

4 Komunikácia s Tripadvisor API

Na odosielanie požiadaviek do Tripadvisor API som vytvorila triedu `TripAdvisorApiWrapper`:

```

class TripAdvisorApiWrapper():
    def __init__(self):
        self.api_key = TA_API_KEY

    def send_request(self, url: str):
        headers = {"accept": "application/json",
                   "Referer": "https://rocnikovy-projekt.onrender.com"}
        response = requests.get(url, headers=headers)
        return json.loads(response.text)

    def find_search(self, search_query: str, lat_long: str, radius: int):
        """
        Parameters:
            search_query - location name;
            lat_long - approx. latitude, longitude;
            radius - radius for search with center in lat_long
        Used for:
            retrieving location id and address
        """
        url = f'https://api.content.tripadvisor.com/api/v1/location/search?language=en&'
        url += f"&searchQuery={search_query}"
        url += f"&latLong={lat_long}"
        url += f"&radius={radius}"
        url += f"&radiusUnit=m"
        return self.send_request(url)

    def location_photos(self, location_id: str):
        """
        Parameters:
            location_id - id from Tripadvisor retrieved using fin_search;

```

```
        Used for:
            retrieving photos
    '''
    url = f"https://api.content.tripadvisor.com/api/v1/location/{location_id}/photos"
    return self.send_request(url)

def location_details(self, location_id: str):
    '''
        Parameters:
            location_id - id from TripAdvisor retrieved using fin_search;
        Used for:
            retrieving latitude, longitude
    '''
    url = location_details_url = f"https://api.content.tripadvisor.com/api/v1/locati"
    return self.send_request(url)
```

5 Komunikácia s mapami

Na vizualizáciu údajov na mape som použila knižnicu MapBox GL JS. Hlavné časti:

```
// Map Initialization
mapboxgl.accessToken = <TOKEN>
mapRef.current = new mapboxgl.Map({
    container: mapContainerRef.current,
    center: [17.10, 48.14], // starting position [lng, lat]
    zoom: 5 // starting zoom
});

// Setting markers for attractions
const marker = new mapboxgl.Marker()
    .setLngLat([long, lat])
    .setPopup(new mapboxgl.Popup().setText(attraction.Name))
    .addTo(mapRef.current!);
```

6 Vývoj 4o-modelového promptu

Prompt môžete nájsť tu: https://github.com/nignatovichhh/rocnikovy_projekt/blob/main/apis/context_prompt.txt

7 Vizualizácia výsledkov

Zdrojové súbory pre Frontend môžete nájsť v Git repozitári.
Príklad výstupu:

FILTERS

Starting city	Starting date	Choose your interests	Number of People	FIND ROUTE
<input type="text" value="Vienna"/>	<input type="text" value="21 / 01 / 2025 , 09 : 00"/>	<input type="button" value="Filter"/>	<input type="text" value="1"/>	
Destination city	Ending date	Choose your age groups		CLEAR FILTERS
<input type="text" value="Linz"/>	<input type="text" value="21 / 01 / 2025 , 23 : 00"/>	<input type="button" value="Filter"/>		

Transport: train

Short Description: A scenic train journey from Vienna to Linz, offering a quick and comfortable travel experience.

Schönbrunn Palace

Schonbrunner Schloßstrasse 47, Vienna 1130 Austria

A grand, historic palace with opulent gardens.

-50 USD

**Belvedere Palace**

Prinz-Eugen-Strasse 27, Vienna 1030 Austria

Historic site with museum showcasing artistic masterpieces.

-40 USD

**Albertina**

Albertinaplatz 1, Vienna 1010 Austria

Art museum boasting extensive graphic art collections.

-35 USD

**Danube Park**

Donauturmplatz 1, Vienna 1220 Austria

Lush park offering trails, outdoor activities and scenic views.

-10 USD

**Lentos Art Museum**

Ernst Koref-Promenade 1, Linz 4020 Austria

Modern art museum with a striking glass facade.

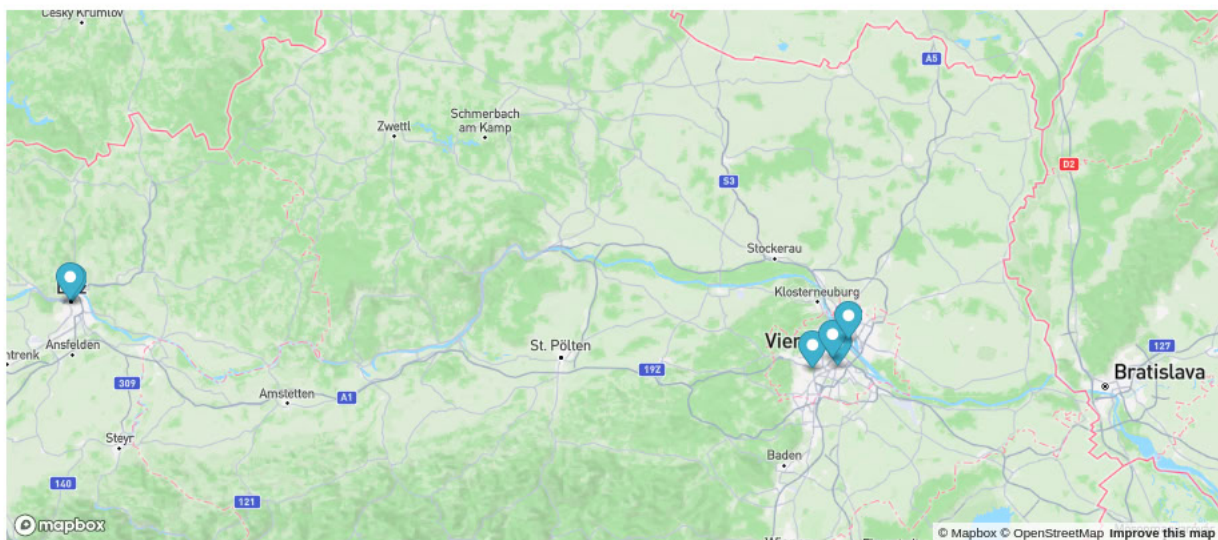
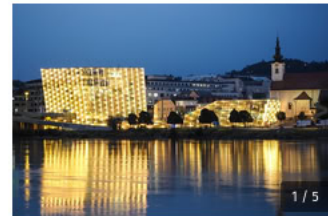
-25 USD

**Ars Electronica Center**

Ars-Electronica-Strasse 1, Linz 4040 Austria

Futuristic museum dedicated to technology and society.

-30 USD



8 Referencie

OpenAI API referencia: <https://platform.openai.com/docs/api-reference/introduction>

TripAdvisor API referencia: <https://tripadvisor-content-api.readme.io/reference/overview>

MapBox GL JS referencia: <https://docs.mapbox.com/mapbox-gl-js/api/>