REQUIREMENTS

PROJECT TITLE:

Microservices for Tourism. Romanian Northeast + Republic of Moldova

TEAM TASK:

Identifying local points of interest:

1. Museums
2. Theaters
3. Pedestrian Zones
4. Parks
5. Botanical Garden
6. Restaurants
7. Bars
8. Sports

TEAM:

1. Balan Gheorghe
2. Bute Iulian
3. Dăscălescu Cezar
4. Maganu Mihai
5. Pîrîu Adrian
6. Stoleru Georgiana Ingrid
7. Strîmtu Gheorghe
8. Vasilache Mihai

TABLE OF CONTENTS:

1. DESCRIPTION:

Firstly, our microservice will respond to requests for a list of POIs (Points of Interest) in a certain area identified by a center point and a radius. The center point is a constant, defined by latitude and longitude, both of which are provided by the geolocation of the user. The radius is an integer, also provided by the user. For example, the user might wish to visualize all the points of interest on an area, defined by a radius of 100m. Further, the application will be able to provide specific information regarding a specific POI, as well as searching for travelling routes between two or more POIs. The order of the operations is not important, as each functionality is independent. Routes/POIs will be delivered as a map frame/ JSON, while details about POIs will be delivered only as a JSON. The input will be JSON, with particular characteristics, adapted to each request. For example, for a route one needs two geolocation points, while for obtaining details about a POI, one needs the name, the type and the estimated geolocation. By using the Google API, we can provide the user with the possibility of finding the best route between two points, with further options of adding intermediate points/ points of interest.

The user will also be provided with the information about all the available transportation options for the route. For example, if the user chooses to travel by taxi, the application would provide to him with the telephone numbers of all the taxi services in the area, ordered by ranking. This would be computed as a function depending on the feedback given by other users. Also, if the user chooses to travel by a rent car, the application would suggest him the nearby stations.

1. SYSTEM ARCHITECTURE

The application is based on REST model. The user can therefore send a request to the server concerning a POI or a route. Afterwards, the information got as input from the user is parsed and several Google API requests will be made in order to get the necessary response as a JSON/ Frame Map. Then, the server creates the output and offers it as a response to the user.

2. SYSTEM MODELS

3. FUNCTIONAL REQUIREMENTS

The application should deliver the required information in the specified format.

When the user requires a route or a POI, the system must return the information regarding the object of the search, formatted as a JSON/ Frame Map.

When the user requests the available transportation options for a particular route, he should receive the telephone numbers of all the taxi services in the area and the stations/schedules specific to the public transport vehicles.

Any other request from the user will be ignored.

4. NON-FUNCTIONAL REQUIREMENTS

The input JSON will be parsed and based on its content, several requests to Google will be made in order to retrieve needed information.

In order to retrieve details regarding an area of a POI, Google Places API will be used. The responses are in JSON format, which will be further processed and only the wanted information will be used for creating the output.

To establish a route through multiple POIs and intermediate locations, the API from Google Maps will be used to retrieve details about possible routes and also details about transportation services for those routes. This data will be further processed to match the wanted output of the initial request.

2. ACTORS AND OBJECTIVES:

The main actors are the software applications, which retrieve information about local points of interests and available vehicles for travelling between them, with further options of passing through intermediate checkpoints.

Objectives:

- a software application should receive details about a specific POI, when needed;
- a software application should receive details about POIs in a specific area as a JSON or as a map, when needed;
- a software application should receive a route passing through multiple points, when needed;

3. USE CASES:

If the user wants to get information about a particular POI, the associated software application will make a POST request to our service, in which there will be integrated the necessary information formated as a JSON. There will be further identified the particular POI, by name, type and geolocation, which are contained in the input JSON.

There will be sent back a JSON that contains all the details about that POI.

If the user wants to get a list of POIs in a specific area, the associated software application will make a POST request to our service in which there will be integrated the necessary information formated as a JSON. This will contain the location and optionally the POI and the output type (IFRAME with a map or a JSON). There will be sent back a JSON that contains all the details about that POIs or an IFRAME with a map.

If the user wants to get a route between multiple points, the associated software application will make a POST request to our service in which there will be provided a JSON that identifies the particular points. This will contains the geolocation and the transportation type (by foot, by car (taxi, rented or personal), public transportation) and the output type (IFRAME / JSON) . There will be sent back a JSON with the desired information or an IFRAME with the required route.