ParkZone: Web application for finding parking spaces

Written and Produced by: Matej Tanaskovski, Vladimir Ivanovski, Hana Hasanicaj, Vesela Trajkoska, Tereza Grujevska

# Description of the project

“ParkZone” is an easy-to-use web-based parking space location application, designed for both computers and mobile phones. When the user begins the application, they are able to select the city or region they are currently in, which then immediately displaying a map with all available parking places in that area. Our application allows the user to pick a parking place from an interactive map. The application then finds and routes the shortest path available to the chosen parking place, while also providing them with directions shown on the interactive map to get there, all with the help of data taken from Open Street View. The application also provides the user with an estimated travel time for the route chosen to be the closest to the parking place. Our databases will be regularly updated, giving the user the most up to date information in order to make it convenient and easy for them to use. Once the user has parked in the parking space, with one click, they are able to fill a space in the parking place which will then be updated in a separate database which stores the status of the parking places. After the user is done using the parking space and has paid for it, the spot becomes free again to be used by another user entirely. Currently, the scope of our project covers Skopje, but in the future more cities and municipalities will be included.

# Functional Requirements

|  |  |
| --- | --- |
| ***Priority Level*** | ***Description*** |
| Priority 1 | Essential Functionality |
| Priority 2 | Desirable Functionality |
| Priority 3 | Optional Functionality |

1. The system shall allow the user to create only one account. (Priority 1)
2. The system shall be accessible on personal computers, which also includes Linux and Mac systems. (Priority 1)
3. The system shall be accessible on mobile devices, which includes IOS and Android. (Priority 1)
4. The system shall make use of data from Open Street View to provide the user with accurate locations and routes. (Priority 1)
5. The system shall store the parking places of a city/region in a database (Priority 1)
6. The system shall provide the user with an interactive map which allows them to pick the parking place they need. (Priority 1)
   1. If there are multiple routes to the parking place, the system shall be able to calculate the closest route possible. (Priority 1)
   2. The system shall also display the travel time to the user based on which route the system calculated (Priority 1)
7. The system shall provide the user with an option to press a button, which will signal that the parking spot the user has chosen is taken. (Priority 1)
   1. The number of free parking spots in a parking place is stored in a separate database, which shall be updated every 5 seconds. (Priority 1)
8. The system shall allow the user to view which parking places have free spots available, but only registered user shall be able to save the parking places they have used before through a separate menu. (Priority 2)
9. The system shall allow the user to view the status of the parking spot they are currently occupying. This shall include: length of stay. (Priority 2)
10. The system shall allow registered user to keep the data of their saved preferences without loss when updating to a newer version. (Priority 2)
11. The system shall be able to provide the user with a tutorial via a separate “Help” menu. (Priority 2)
12. The system shall be available in: Macedonian, Albanian and English languages. (Priority 3)

# Non-Functional Requirements

1. The user shall be able to fully understand the application after 5 minutes of use.
2. The system shall be interactive, and there mustn’t be any more than 20ms delay.
3. The system shall be secure 99% of the time.
4. If the internet connection fails while the user is receiving information, that same information should be resent when back online.
5. The system must show information about the availability of parking spots to the user.
6. The system shall be able to perform its operations successfully 99% of the time.
7. In case of a failure, the system shall be restored in the next 5 seconds.
8. The system’s loading times shall not be more than one second for the users.
9. The system shall be able to store up to 1,000,000 parking spaces.
10. The system shall be refreshed every 5 seconds.
11. The system shall have a response time of less than 4 seconds.
12. The system shall support up to 500 requests to fill a parking spot simultaneously.
13. The system shall support up to 50000 concurrent users.