Contents

[1. Abstract 2](#_Toc36394639)

[2. Stakeholders 2](#_Toc36394640)

[3. Context diagram and interfaces 2](#_Toc36394641)

[4. Stories and personas 3](#_Toc36394642)

[5. Functional requirements 4](#_Toc36394643)

[6. Nonfunctional requirements 4](#_Toc36394644)

[7. Use cases 4](#_Toc36394645)

[8. Use case diagram 5](#_Toc36394646)

[9. Relevant scenarios 5](#_Toc36394647)

# Abstract

Most individuals prefer to use their own cars instead of public transport. Even car prices are getting more and more affordable, the increasing gas price push people drive their cars more efficiently. Changing driving habits help to reduce the gas consumption. An additional solution could be to minimize gas expenses by filling the tank from the cheapest nearby gas station.

The EZGas application is a crowdsourcing application. It shows the gasoline prices at the nearby gas stations to the users. Instead of establishing a direct communication with gas station holders, the application allows users to insert new prices to the application. The other users are notified when an update request is sent to the application maintainers. In this way, the other users can check whether the update request is correct or not. The application is a simple, yet useful tool that helps the users to select the best option to fill their cars’ gas tank.

# Stakeholders

**Final user:** Car owner who will use the application to find the best nearby gas station. S/He also involve activities to update the prices in database. User can be registered or unregistered.

**Supervisor:** Since the prices are inserted to the database by final users, the other users, who buy gas from the same station, can check whether the registered prices in the application are correct or not. Thus, the final users take also the supervisor role.

**Inspector:** There might be some bugs in the application code. Final users can report these bugs while they are using the application. In this case, the users are also the inspectors.

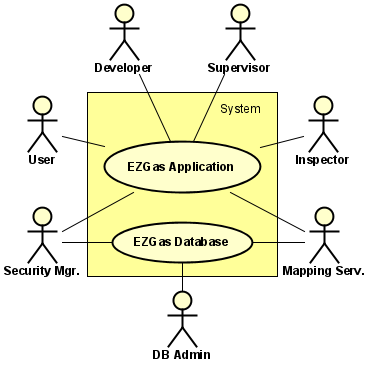
**Database administrator:** All the price information must be stored in a database. The database administrator manages it.

**Security manager:** Since the application uses the users’ location and this data is stored in the database, a security manager must care for the data privacy.

**Developers:** Persons who has expertise in code development

# Context diagram and interfaces

In the context diagram the whole system is EZGas application. The hole system consists of the application GUI for the users and the database to store all the information.

Context diagram

|  |  |  |
| --- | --- | --- |
| **Actor** | **Logical interface** | **Physical interface** |
| User | GUI | Touch screen |
| Supervisor | GUI | Touchscreen |
| Inspector | GUI | Touchscreen |
| Security manager | Database computer | Keyboard, mouse |
| Database administrator | Database computer | Ethernet wires and other devices |
| Developer | IDEs | Personal computer |
| Database | Internet connection | Wires |
| Mapping service | APIs | Smartphone |

Interfaces

# Stories and personas

Carl works in a company as apart time employee and he serves as a cab driver with his car in his spare time. With another application, he checks whether someone needs a taxi and he uses his car as a taxi to gain extra money in his spare time. He usually uses his car in urban traffic. Thus, he loses a lot of time while waiting in traffic and this causes a considerable increase in gasoline consumption. He wants to increase his benefit by paying as much less as possible for gasoline while serving as a cab driver. The best way is using a simple yet easy to use application to find the nearby station which offers the lowest price.

Carl also strictly relies on the power of communities. He believes he can make a positive impact on other peoples’ life by helping them to reduce their gasoline expense. For this reason, he also wants to share his experiences and gasoline prices at different stations. In this way, more people might reach relatively cheap gasoline and the other stations would decrease the prices. As a consequence, he could also make a positive impact also on income injustice.

He uses this application not only for their own benefit but also to help to the others for reducing their expenses, they would also be happy to involve in development of the application by sending bug reports.

Since anyone is free to install the application and become a member of this life changing community, some intruders can also be member of the community. The intruders may try to manipulate the prices in order to fool the members of this great community and they may canalize the community members prefer wrong gasoline stations. Hence, Carl would also want to inspect the price update requests.

# Functional requirements

|  |  |
| --- | --- |
| **FR ID** | **Functional requirement description** |
| FR1 | The user shall be able to manually adjust the location |
| FR2 | The user shall be able to send price update request |
| FR3 | The user shall be able to use the application with live location |
| FR4 | The user shall be able to filter gasoline type |
| FR5 | The user shall be able to filter gas station |
| FR6 | The user shall be able to see the other services at stations (w.c, market, etc.) |
| FR7 | The user shall be able to rate and post comments about gas stations |
| FR8 | The user shall be able to see real time station status (open, closed, crowded etc.) |
| FR9 | The user shall be notified about discounts |
| FR10 | The supervisor shall be able to confirm or reject the update request |

# Nonfunctional requirements

|  |  |
| --- | --- |
| **NFR ID** | **Nonfunctional requirement description** |
| NFR1 | The user shall obtain virtual ranks based on application usage frequency |
| NFR2 | The application should be able to be downloaded and installed in a minute |
| NFR3 | The user interface should be easy to use |
| NFR4 | The application should serve in different languages |
| NFR5 | The application should be used without any training |
| NFR6 | The application should run on Android and iOS OSes |
| NFR7 | The GUI should be simple in order to reduce CPU utilization |
| NFR8 | It must be able to serve during database update |

# Use cases

|  |  |
| --- | --- |
| Use case 1, UC1 – FR1 Adjust location manually | |
| Actors involved | User |
| Pre condition | Location is defined basis on the live location |
| Post condition | Location is defined by the user |
| Nominal scenario | All the gas stations in the selected region are shown |
| Variants |  |

|  |  |
| --- | --- |
| Use case 2, UC2 – FR2 Send price update request | |
| Actors involved | Registered user |
| Pre condition | Internet connection must be established |
| Post condition | The request is received by the system administrator |
| Nominal scenario | User sees a message that the request is received |
| Variants | Request may not be received due to connection problems |

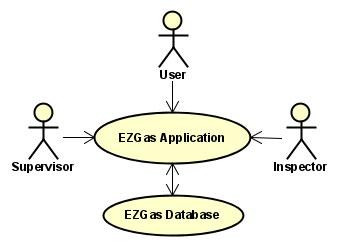
|  |  |
| --- | --- |
| Use case 3, UC3 – FR4 Filter gasoline type | |
| Actors involved | User |
| Pre condition | Any other type of gas is selected |
| Post condition | User selected the gas type suitable for his car |
| Nominal scenario | Only the selected gas type prices are shown |
| Variants | That type of gas may not available in the selected region |

|  |  |
| --- | --- |
| Use case 4, UC1 – FR5 Filter gas station | |
| Actors involved | User |
| Pre condition | Any gas station brand is selected |
| Post condition | The desired gas station brand is selected |
| Nominal scenario | Only the selected brands are shown |
| Variants | The selected brand may not exist in the selected region |

|  |  |
| --- | --- |
| Use case 5, UC1 – FR5 Additional services | |
| Actors involved | User |
| Pre condition |  |
| Post condition | User checks the nearby stations whether they provide the desired service |
| Nominal scenario | User finds the requested service in one of the nearby stations |
| Variants | None of the stations may not provide that service |

# Use case diagram

The application is simple. In the nominal case the user applies the filters as he wants and sends a request to the database to see the nearby gas stations. The request is sent through the application user interface. The database applies the request filters and returns results to the application.

Some of the registered users have supervisor authorities. Any user can send a price update request. Only the supervisors can confirm or reject an update request. All actions and information stored in the application database.

Use case diagram

# Relevant scenarios

|  |  |
| --- | --- |
| **Scenario ID: SC1** | **Corresponds to UC1** |
| Description | User wants to select the region manually |
| Precondition |  |
| Post condition | Stations and prices in the selected region are shown to the user |
| **Step#** | **Step description** |
| 1 | User pans, makes zoom in and out by using the touch screen |

|  |  |
| --- | --- |
| **Scenario ID: SC2** | **Corresponds to UC2** |
| Description | User wants to send a price update request |
| Precondition | Station and gas type must exist |
| Post condition | An update request is received by the database maintainer and the other registered users are notified |
| **Step#** | **Step description** |
| 1 | User selects a gas station to visualize gasoline types |
| 2 | User taps and hold on the target gas type |
| 3 | In the pop up menu, user taps on “send update request” |

|  |  |
| --- | --- |
| **Scenario ID: SC3** | **Corresponds to UC3** |
| Description | User wants to see the prices in his current location |
| Precondition | Live location must be switched on in phone settings |
| Post condition | All the nearby gas stations are shown on to the user |
| **Step#** | **Step description** |
| 1 | In the application menu screen user selects “show nearby stations” |

|  |  |
| --- | --- |
| **Scenario ID: SC4** | **Corresponds to UC4** |
| Description | User wants to see the prices of a specific gas type |
| Precondition |  |
| Post condition | Price of the selectedgas type in nearby gas stations are shown |

|  |  |
| --- | --- |
| **Scenario ID: SC5** | **Corresponds to UC5** |
| Description | User wants to filter gas stations |
| Precondition | There must be nearby gas stations |
| Post condition | The nearest selected gas station is shown in the application GUI |
| **Step#** | **Step description** |

|  |  |
| --- | --- |
| **Scenario ID: SC6** | **Corresponds to UC56** |
| Description | User wants to see extra services provided by the station |
| Precondition | A station must be selected |
| Post condition | The extra services at the selected station are shown |

|  |  |
| --- | --- |
| **Scenario ID: SC7** | **Corresponds to UC7** |
| Description | User wants to rate or wants to write comments about the station |
| Precondition | User must be registered |
| Post condition |  |
| **Step#** | **Step description** |
| 1 | User selects a gas station |
| 2 | User rates or send comment about the station |

|  |  |
| --- | --- |
| **Scenario ID: SC8** | **Corresponds to UC8** |
| Description | User wants to see current status of the selected gas station |
| Precondition | User must have internet connection |
| Post condition | The current status of the station is shown |
| **Step#** | **Step description** |
| 1 | User selects a gas station and taps on “show status button” |

|  |  |
| --- | --- |
| **Scenario ID: SC9** | **Corresponds to UC9** |
| Description | User wants to see any kid of discount or special offer at a station |
| Precondition | Station and gas type must exist |
| Post condition | Any offer or discount is shown to user |
| **Step#** | **Step description** |

|  |  |
| --- | --- |
| **Scenario ID: S10** | **Corresponds to UC10** |
| Description | Supervisor (user) wants to confirm or reject an update request |
| Precondition | User must be registered and must have supervisor authorization  Supervisor must select a station which is subject of an update request |
| Post condition | The update request is confirmed or rejected |
| **Step#** | **Step description** |
| 1 | Supervisor user is notified about update request |
| 2 | S/he confirms or rejects the request |