# EZGas

## Abstract

Some people are fed up about the rising gas prices, so an app developer tries to capitalize on this need by creating an app in which is possible for each user to mark out gas prices at each gas station and to see prices uploaded by other users.

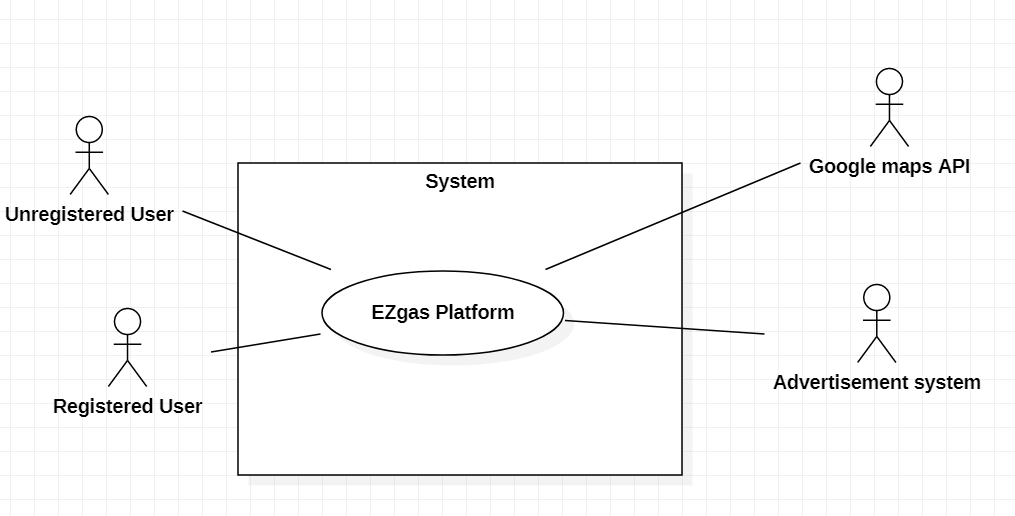
In this way everyone can choose the most convenient place to refuel his or her car. The interaction in done through an app which requires user registration. Each user is associated with a reliability score to prevent malicious app usage.

The source of income for the developer will be advertisement placed in the app in a non-invasive way

## Stakeholders

|  |  |
| --- | --- |
| **Stakeholder name** | **Description** |
| User | Uses the application to signal gas prices and to view reports for gas stations for that day. |
| Platform Administrator/App developer | Develops the App and the platform.  Manages and oversees backend platform operation, he makes sure the database server and application server stay up and running as well as |
| Google Maps API | Manages the maps API, the app must follow them |
| Advertisement system | Manages advertisement shown in the app |

## Context Diagram



## Interfaces

|  |  |  |
| --- | --- | --- |
| **Actor** | **Logical Interface** | **Physical Interface** |
| User | GUI | TouchScreen |
| Advertisement system | API | Automatic system |

## Stories and personas

Luca uses his car every day to go to work. He has to drive many kilometres so he has to refuel his car quite often. In his path there are many gas stations with different prices every day.

Obviously he would like to go where prices are lower, unfortunately these stations are distant from one another and some aren’t directly in his path despite being near, so Luca can’t visit them all to chose the most convenient.

To solve this problem Luca installs EZgas on his smartphone, in this way he can see gas prices beforehand and choose the gas station before leaving his home, saving him a lot of money and time during the year.

Luca leaves home very early in the morning so unfortunately sometimes prices on the app aren’t updated yet. He is a very selfless person and doesn’t want other people to see a wrong price on the app. So he opens EZgas and updates the gas prices for that station, from that moment everyone sees the new information and his reliability score goes up.

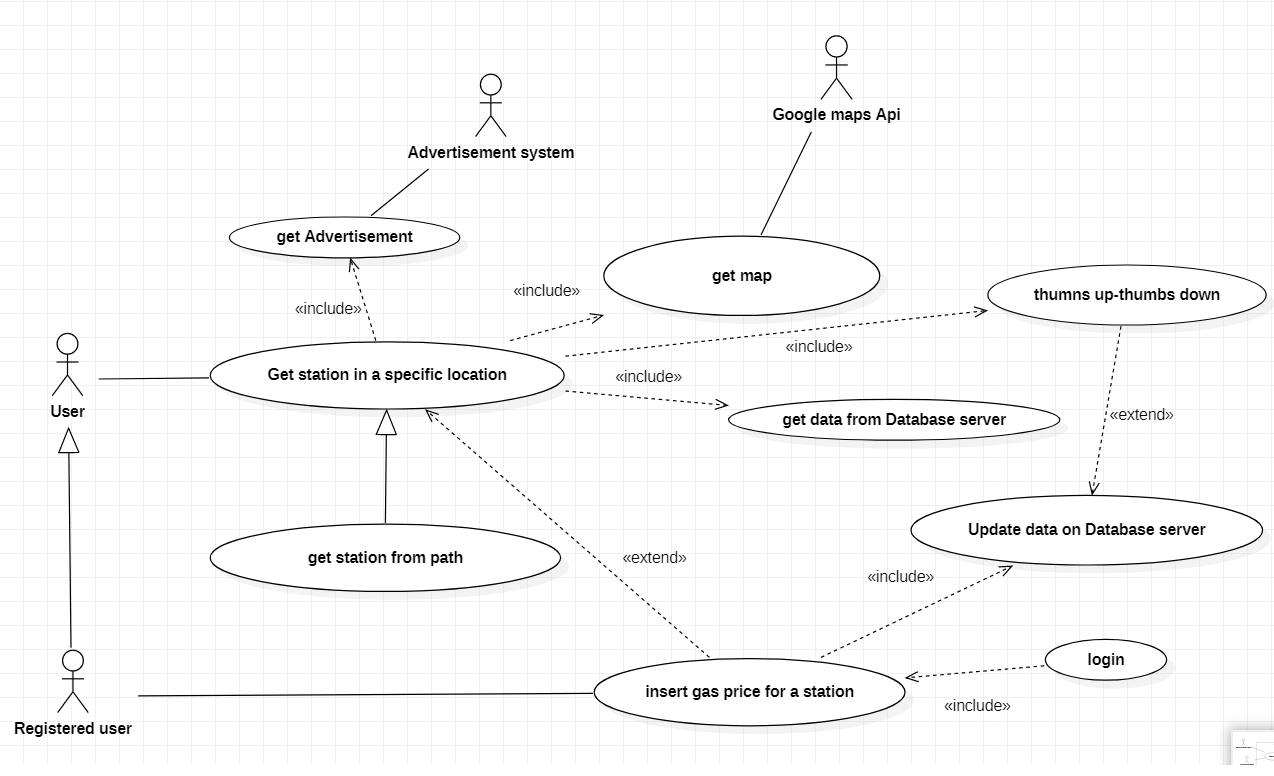
## Functional Requirements

|  |  |
| --- | --- |
| **ID** | **Description** |
| FR1 | Log in |
| FR2 | Log out |
| FR3 | Insert location with radius In which to show gas stations |
| FR4 | Insert path along which show gas stations |
| FR5 | Insert new gas station |
| FR6 | Update price for gas station |
| FR7 | Show pointers on map for gas stations |
| FR8 | Show prices for the day of a gas station |
| FR9 | Give another user a thumbs up (to increase his reliability score) |
| FR10 | Give another user a thumbs down (to decrease his reliability score) |
| FR11 | Temporarily block unreliable user from updating prices |
| FR12 | Show advertisement at the bottom of the screen |
| FR13 | Sign in |

## Non Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Type** | **Description** | **Refers to** |
| NFR1 | Portability | The application runs on android and iOS | ALL FR |
| NFR2 | Portability | Data must be compatible between android and iOS | ALL FR |
| NFR3 | Usability | Sign in and log in should be fast and non-tedious (possibly using google account) | FR1, FR2, FR13 |
| NFR4 | Usability | Adding a new station should be easy and intuitive (measure: mean number of click, number of forms to fill) | FR5 |
| NFR5 | Usability | Updating a price should be easy and intuitive (measure: mean number of click, number of forms to fill) | FR6 |
| NFR6 | Performance | All functions should complete in < 1 sec (depends on internet connection) | ALL FR |
| NFR7 | Localization | Currency is in Euro | ALL FR |
| NFR8 | Reliability | Block a user whose reliability scores is <0  For one week more each time it is blocked | FR9, FR10, FR11 |

## Use case diagram



## Use cases

Use case 1 UC1, get station in a specific location

|  |  |
| --- | --- |
| **Actors Involved** | User |
| **Precondition** | The user has specified a location |
| **Post Condition** | The map with all the gas station is shown |
| **Nominal Scenario** | A user opens the app, types an address A and a radius R , the app communicates with the map system and a map with the pointers of all the gas stations in a radius R around A is shown, by clicking on a pointer price information about that station are shown |
| **Variants** | The users wants to see all the gas station in a path, so he types a departure address A and an arrival address B, the app communicates with the map system and a map with the path from A to B and all the gas stations in it is shown, by clicking on a pointer price information about that station are shown |

Use case 2 UC2, thumbs up- tumbs down

|  |  |
| --- | --- |
| **Actors Involved** | User |
| **Precondition** | Gas station information have been shown |
| **Post Condition** | The user who has updated the price receives a thumbs up or a thumbs down->its reliability score is updated (max +1, -1 every day, to get +1 at least 5 thumbs up and to get -1 at least 5 thumbs down) and if gets below 0 the user is blocked. |
| **Nominal Scenario** | A user U sees that the price is correct/incorrect and clicks on thumbs up/down button |
| **Variants** |  |

Use case 3 UC3, insert gas price for a station

|  |  |
| --- | --- |
| **Actors Involved** | Registered user |
| **Precondition** | The user has done log in and selected a gas station |
| **Post Condition** | The gas station information is updated an available to everyone |
| **Nominal Scenario** | A Registered user U selects a gas station (see UC1) and by pressing on a specific button is prompted to insert the updated information, after U confirms, data are sent to the server which updates the database and makes new information available to everyone |
| **variants** |  |

## Relevant Scenario

Scenario 1

|  |  |
| --- | --- |
| **Scenario ID: SC1** | **Corresponds to UC1** |
| **Description** | A User wants to see information about gas prices on his way to work |
| **Precondition** | EZGas is installed on User smartphone |
| **PostCondition** | The information is shown on User’s smartphone screen |
| **Step#** | **Step description** |
| 1 | User types departure address D |
| 2 | User types arrival address A |
| 3 | D and A are sent to google maps server |
| 4 | A map with the path between A and B is received |
| 5 | Based on the path (which is made of a series of coordinates) location data about gas stations are retrieved from database server |
| 6 | The map with the path is shown on user smartphone, for each gas station in the path a clickable pointer is printed in the corresponding location |
| 7 | The user clicks on a pointer |
| 8 | The id of the corresponding station is sent to the database server which retrieves the most recent information about the gas price in that station |
| 9 | The information is printed on a window above the pointer |

Scenario 2

|  |  |
| --- | --- |
| **Scenario ID: SC2** | **Corresponds to UC3** |
| **Description** | A User wants to update information about gas prices on his way to work |
| **Precondition** | EZGas is installed on User smartphone and the user is registered to the platform |
| **PostCondition** | The information about a gas station is updated |
| **Step#** | **Step description** |
| 1 | User fills login information |
| 2 | Login information are sent to the database server |
| 3 | User is granted Edit access |
| 4 | User types an address A and a radius R |
| 5 | A is sent to Google Maps server |
| 6 | A and R are sent to Database server |
| 7 | The map is received from google maps server |
| 8 | A list of all gas station location within a radius R from address is received from database server |
| 9 | The map is shown on user smartphone, for each gas station in the path a clickable pointer is printed in the corresponding location |
| 10 | The user clicks on a pointer |
| 11 | The id of the corresponding station is sent to the database server which retrieves the most recent information about the gas price in that station |
| 12 | The information is printed on a window above the pointer |
| 13 | The user clicks on edit button |
| 14 | The user is prompted for the new gas price |
| 15 | The new information is sent to the database server |