OFFICIAL REQUIREMENTS DOCUMENT

Authors: Francesco Xia

version: 1.1

CHANGELOG:

ver. 1.1:

* added the glossary
* updated the stakeholder section
* minor changes to the functional requirements

ABSTRACT

Ezgas is an application that enables end-users to search for gas stations, along the pricing of various types of fuel, that meet a set of criteria (i.e. filters). The search results are presented in a list, or optionally displayed on a built-in map.

The prices are kept up to date by updates made by verified users and/or directly from gas station owners.

Ezgas is supported by a web application, accessible both via smartphone and

STAKEHOLDER

* END USER

person who ultimately use the web application

* verified user (profile 1)

can send price updates related to a gas station

* business user (profile 2)

owner of a gas station

* unverified user (profile 3)

are not allowed to send price updates.

* CUSTOMER SUPPORT

responsible of the daily to daily operations

* DEVELOPER

responsible in building and writing the web application

* GOOGLE MAPS

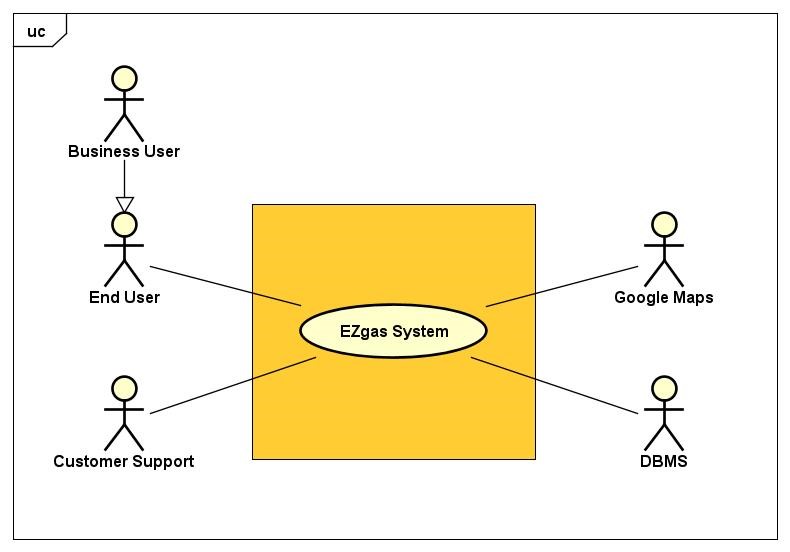
responsible of locating geographically the gas stations

* DBMS

responsible of managing information about accounts and gas stations

CONTEXT DIAGRAM

The following diagram defines and clarifies what we consider part of the system and what it is outside.



INTERFACES

|  |  |  |
| --- | --- | --- |
| Actor | Physical interface | Logical interface |
| end user | screen, keyboard | GUI |
| customer support | screen, keyboard | GUI |
| google map | internet connection | google maps API |
| dbms | internet connection | dbms API |

STORIES AND PERSONAS

Mark is a graduate student that just transferred in a new neighborhood. During the week, he goes back and forth to the university using the car, spending too much money, in his opinion, on fuel. He would love to find the cheapest gas station in his neighborhood, but he has little or no time to look around.

Evan is the owner of a gas station which it is situated far away from the main road. Although there is only an another gas station in the radius of 10km, he has not many customers since not many people know of its whereabouts. Thus, he would like to advertise his gas station on the web hoping to attract new customers.

Elizabeth needs to find the nearest gas station to refuel her car.

SCENARIOS

SCENARIO ID: SC1

description:

* user wants to update the price of fuel with respect to a gas station

precondition:

* user must be authenticated

post condition:

* the system logs the update made by the user

STEPS

NOMINAL

1. user fills in the minimum required information (see GLOSSARY)
2. receive a visual confirmation

EXCEPTIONS

2a. the price is invalid, e.g. price out of range

2b. the gas station id is invalid

SCENARIO id: SC2

description:

* user wants to find all the gas stations that meet a specified set of criteria

STEPS  
NOMINAL

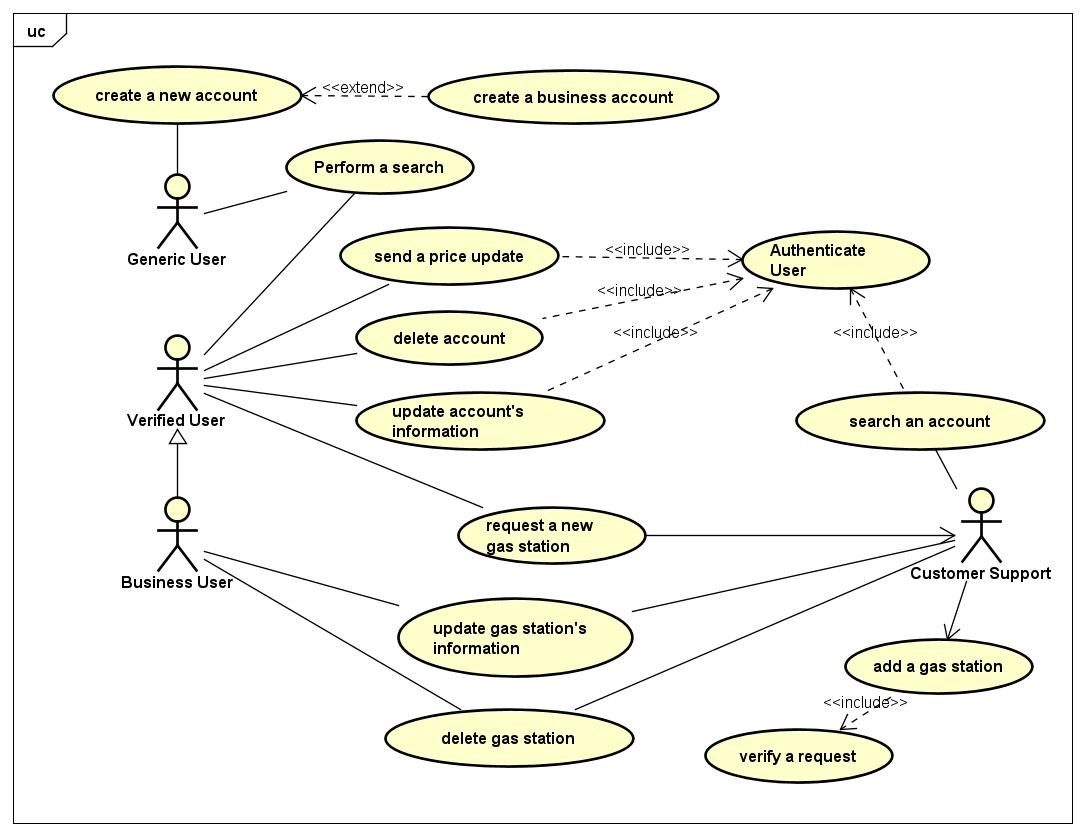
1. user fills in the address information and optionally apply a set of filters
2. show to the user the filtered results

EXTENSION

2a. show the result on a map

2b. the list is empty

USE CASES



USE CASE id: UC1 (related to FR3)

GOAL: send a price update

precondition:

* user must be authenticated before he/she can access this functionality

Main Success Scenario:

1. user fills in the required fields of the form, i.e. gas station id, fuel type and price (see GLOSSARY)
2. system confirms the request with an alert

Extension:

2a: the gas station doesn't exist in the system

.1: send an error message to the user

Optional:

* initially the user may perform a search to find the gas station id (see UC2)

USE CASE id: UC2 (related to FR4)

GOAL: perform a conditional search using a set of filters

Main Success Scenario:

1. user fills in the address information (i.e. street and city) and optionally can apply a set of filters (see GLOSSARY)
2. system presents a list of gas stations that meet the criteria

Extension:

2a: the list is empty

2b: show the list of gas stations on a map

Optional:

* the search functionality may be preceded by an authentication phase

USE CASE id: UC3 (related to F5.1a, F5.1b)

GOAL: create a new basic account

postcondition: the information about the newly created account must be stored in the DB

Main Success Scenario:

1. user fills in the minimum required information (see NOTE)
2. system confirms the successful creation of the account

Extension:

1a: username already taken and/or password too weak and/or email address is invalid or already taken

.1 user may retry, return to MSS at step 1

1b: user decide to create a business account

.1 user fills in the minimum required information for a basic account

precondition: the gas station is already in the system

.2a request to link this account to a gas station (using the gas station id) and return to MSS at step 2

extension:

.2aExt the gas station is already linked to an another account. Abort the procedure.

NOTE: to find out if a gas station is already in the system, the user may perform a search

precondition: the gas station is not present in the system

.2b follow the steps described in UC7 ("add a new gas station")

.3 if the previous step has been successful link the gas station to the new account

.4 system confirms the creation of the account

USE CASE id: UC4 (related to FR5.2)

GOAL: update account information

precondition:

* user must be signed into the system, he/she can modify only the information related to his/her account

postcondition:

* the new information corresponding to the user's account must be stored in the DB

Main Success Scenario:

1. user updates his personal information (see GLOSSARY)
2. system confirms the update

Extension:

1a: user decides to update his/her password, but new one is not compliant with the minimum requirements

.1 user may reenter a new password

USE CASE id: UC5 (related to FR5.3)

GOAL: delete an account

precondition:

* user must be signed into the system, he/she can modify only the information related to his/her account

postcondition:

* the account has been deleted from the system

Main Success Scenario:

1. user sends a request to delete his account
2. system confirms the account was successfully deleted

USE CASE id: UC6 (related to FR5.4)

GOAL: search an account in the system

precondition: accessible only to customer support

Main Success Scenario:

1. search an account using an account id
2. system prints the result

Extension:

2.a the account doesn't exist

USE CASE id: UC7 (related to FR6.1a, FR6.1b)

GOAL: add a new gas station into the system

Main Success Scenario:

1. user fills in the minimum information required for a gas station (see GLOSSARY)
2. system accepts the request
3. customer support verifies and validates the information
4. customer support adds the gas station into the system
5. system sends a confirmation email to the user

Extension:

3b: the gas station already exist in the system

.1 sends an email to the user saying the gas station already exist in the system

1. abort the procedure

USE CASE id: UC8 (related to 6.2)

GOAL: updates the information related to a gas station

precondition:

* user must be signed into the system (either a customer support or the associated business account)

Main Success Scenario:

1. user updates the information related to the gas station (see GLOSSARY)
2. system confirms the update

USE CASE id: UC9 (related to 6.3)

GOAL: delete a gas station from the system

precondition:

* user must be signed into the system (either a customer support or the associated business account)

Main Success Scenario:

1. user sends a request to delete his account
2. system confirms the deletion

FUNCTIONAL REQUIREMENTS

|  |  |
| --- | --- |
| UNIQUE ID | DESCRIPTION |
| FR1 | Authenticate a user |
| FR1.1 | log in to the system with username and password |
| FR1.2 | log out from the system. |
| FR3 | send a price update corresponding to a gas station |
| FR4 | perform a conditional search using a set of filters |
| FR5 | manage new and existing accounts: create, show, modify or delete an existing account from the system. |
| FR5.1a | create a normal account |
| FR5.1b | create a business account, i.e. the account of the owner of a gas station |
| FR5.2 | modify the information related to an account |
| FR5.3 | delete an existing account |
| FR5.4 | search an account inside the system |
| FR6 | manage new and existing gas station |
| FR6.1a | send a request to add a new gas station |
| FR6.1b | add a new gas station to the system |
| FR6.2 | update the information related to a gas station |
| FR6.3 | delete a gas station from the system |

NON-FUNCTIONAL REQUIREMENTS

|  |  |  |  |
| --- | --- | --- | --- |
| UNIQUE ID | TYPE | DESCRIPTION | REFERS TO |
| NFR1 | Usability | application should be user-friendly with no training time required | ALL FR |
| NFR2 | Efficiency | the system should give an answer to any user request in < 1 sec | ALL FR |
| NFR3 | Portability | the web application should be accessible and compatible with the most popular browers (chrome, firefox, edge, opera), and mobile operating systems (OS, Android) | ALL FR |
| NFR4 | Reliability | the system should be available all the time with backup servers ready in case of failures of the main system |  |
| NFR5 | Localization | the prices are shown in euro |  |

GLOSSARY

* ACCOUNT

*minimun information* required for a *basic account* to be considered valid:

* username, password, email address

*minimum information* required for a *business account* to be considered valid:

* username, password, email address and gas station ID.

*OPTIONAL* properties

* full name, address, phone number

Every user account in the system is uniquely identified through an ID.

* GAS STATION

*minimum information* required for a *gas station* to be considered valid:

* gas station name, street name and number, city, postcode, country

*OPTIONAL* properties

* price catalog, i.e. a set of (fuel type, price), brand, charging station (boolean), opening hours, phone number.

If not supplied, they will be set to default values.

A gas station is only useful if the price catalogue property has been set.

Every gas station in the system is uniquely identified through an ID.

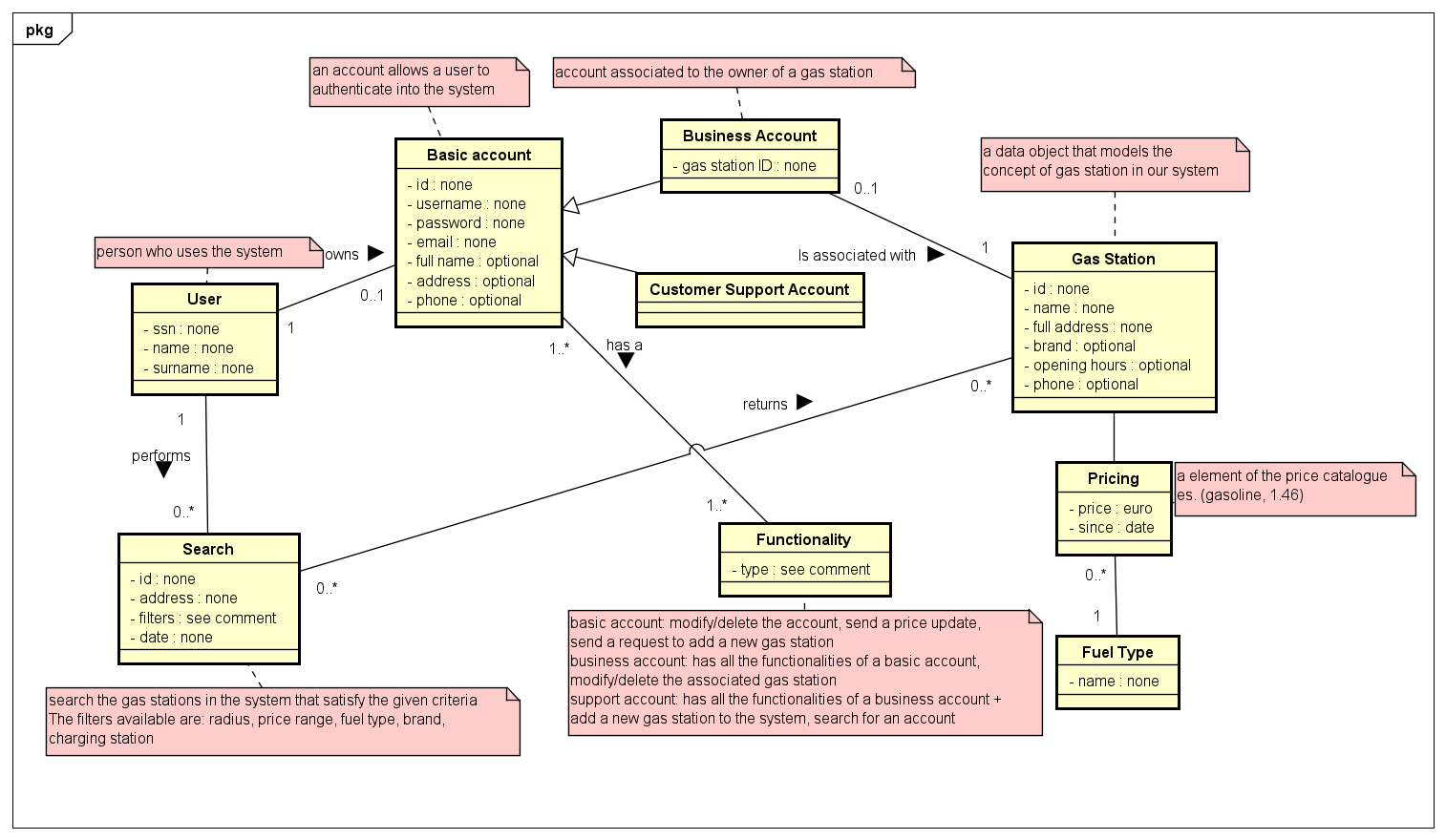
* FOR SEARCHING

For a basic search the user must fill this required fields: street name, postcode, country

The filters available are:

* radius (km), price range (min, max), fuel type (gasoline, diesel), brand (Total, Shell, BP, Esso…), and charging station (boolean).

If not specified, they will be set to default values.

* FOR SENDING PRICING UPDATES
  + minimum information required is the gas station ID, the fuel type and the price.