POLITECNICO
MILANO 1863

Requirement Analysis and Specification Document

eMall – e-Mobility for All

Software Engineering 2, 2022/23

Giulia Huang, Zheng Maria Yu, Linda Zhu

V1.1 2023/01/08

Contents

1 Introduction

1.1 Purpose

In the last years, the high carbon footprint has been a global challenge for its negative environmental impact, since it plays a relevant role in climate change and air pollution. In particular, the transportation sector is one of the primary sources of greenhouse gas emissions due to the combustion of fossil fuels.

Electric mobility represents a solution to alleviate the problem: electric cars require less energy and produce less polluting gas as well. However, people should take into account the charging-related issues when using electric vehicles: they need to find available charging stations, and to consider the charging time of the cars too.

To facilitate the use of electric vehicles, the eMall (e-Mobility for All) system proposes to keep together and to coordinate all the activities that the charging process would require. In fact, The eMall system connects the various providers involved in the activity: e-Mobility Service Providers (eMSPs) aim to help drivers in planning and completing the charging process of their electric vehicles introducing minimal interference and constraints with respect to their daily schedule, while Charging Point Operators (CPOs) manage the changing stations, offer functionalities through their own Charge Point Management System (CPMS), and eventually acquire energy from Distribution System Operators (DSOs).

The present document represents the baseline for the project planning and implementation. Its objective is to analyze the eMall system requirements and to define its application domain, functionalities and use cases.

1.1.1 Goals

Goal	Description
G1	The system shall allow the users to know the location and the status of charging stations
G2	The system shall allow the drivers to book a charge
G3	The system shall allow the drivers to charge the vehicle according to the reservation
G4	The system shall allow the drivers to pay for the obtained service

G5	The system shall suggest the drivers to charge the vehicle when necessary
G6	The system shall allow the CPOs to decide the energy source for charging
G7	The system shall allow the CPOs to change the charging cost and the special offers
G8	The system shall allow the CPOs to acquire energy from the DSOs

1.2 Scope

The present document will focus on the analysis of the eMall case, which is composed of the two subsystems eMSP and CPMS. In our application domain, each CPMS is owned by a different CPO. An eMSP can interact with multiple CPOs through uniform APIs, while a CPO can interact with multiple eMSPs and DSOs as well.

1.2.1 World Phenomena

Identifier	Description
WP1	People use electric vehicles
WP2	Electric vehicles have a rechargeable/working battery
WP3	Drivers need to charge their vehicle
WP4	There are available charging stations
WP5	Charging stations have enough energy supply

1.2.2 Shared Phenomena

- World Controlled Phenomena

Identifier	Description
SP1	User registers an account
SP2	User logs into his account
SP3	User consults information about the charging stations
SP4	Driver books a charge in a charging station
SP5	Driver starts the charging process in a booked charging station
SP6	Driver pays for the obtained service

SP7	Operator changes the charging cost
SP8	Operator sets a special offer
SP9	Operator sets the energy source for charging
SP10	Operator acquires energy from the DSOs

- Machine Controlled Phenomena

Identifier	Description
SP11	System notifies the user when the charging process finishes
SP12	System suggests the user to charge the vehicle
SP13	System gives information on the status of a charging station

1.3 Definitions, acronyms, abbreviations

1.3.1 Definitions

Definition	Description
Electric vehicle	Vehicle with battery and electric motors for propulsion
Driver	User of electric vehicles
Energy	Power used by electric vehicles
Charging station	Location where the electric vehicles can be charged, it could have batteries to store energy
Charging column	Structure positioned in charging stations to charge electric vehicles. It has a display screen and a charging socket
Charging socket	Connector to plug in electric vehicles. There are different types of charging sockets according to the power output, such as slow/fast/rapid
Operator	Charging station administrator working for a CPO
User	Person using the system, can be a driver or an operator
Reservation	Booking made by a driver for charging at a specified time frame

1.3.2 Acronyms

Definition	Description
RASD	Requirements Analysis and Specification Document
eMall	e-Mobility for All
eMSP	e-Mobility Service Provider
CPO	Charging Point Operator
CPMS	Charge Point Management System
DSO	Distribution System Operator

1.3.3 Abbreviations

Abbreviation	Description
WP	World Phenomena
SP	Shared Phenomena
GX	Goal number X
DX	Domain assumption number X
RX	Requirement number X

1.4 Revision history

- V1.0: Initial version.
- V1.1: Review of system requirements (in particular about the charging process), update of use cases according to the modification to the start and end charging procedures, minor fixes.

1.5 Reference Documents

- Project specification: "Assignment R&DD A.Y. 2022-2023 v3"
- Software Engineering 2 Course slides A.Y. 2022-2023

1.6 Document Structure

This document is composed of six sections that describe our system in detail:

The first section consists in an introduction of our project. It starts with a description of the main problem that the system will deal with, the list of goals to

achieve and the specification of its scope with various phenomena occurring. In the last part, a list of several definitions and abbreviations is reported for a better understanding of the document.

Section two contains an overall description of our system. It includes several realistic scenarios on the interaction between Users and the System, clarified by statecharts which describe the behavior of system and by a class diagram which offers an overview of the main entities of the system and their relationships. Moreover, there is a list of main functionalities that the system will offer under some domain assumptions, assumed to hold in the world.

Section three aims to specify the requirements of the system, such as requirements on external interfaces, functional requirements with the defined use cases, and non-functional requirements. Use cases are described and clarified by use case diagrams and sequence diagrams. Section three also contains details on the relationships between functional requirements, the goals of the system and the use cases. Lastly, the design constraints are specified too.

Section four includes a formal analysis of the system with the use of Alloy. The Alloy code is reported with the description of analysis objectives.

In section five there is a presentation of the project members total effort spent on writing the RASD.

Section six contains the references used in writing this document.

2 Overall Description

2.1 Product perspective

2.1.1 Scenarios

A. Driver wants to start to use the system

Driver A owns an electric vehicle, but he always has troubles on choosing which charging stations to go to charge his car. He decides to start to use the eMall platform. First, he downloads the application and then registers himself with the requested data. After the completion, he logs into his account. He registers his vehicle in the profile and finally he can take advantage of all the services offered by the application.

B. Driver wants to get information about charging stations

Driver D arrives in a new city for business, and he wants to compare the price of charging stations with those in his city. For this reason he decides to use the eMall application. After he logs in with his account, the map in the main page shows the position of Driver D and the available charging stations nearby. When Driver D selects a charging station from the map, he can visualize its charging prices and the special offers if available.

C. Driver wants to book a charge

Driver E realises that his electric vehicle is running out of battery, and he decides to use the eMall platform to find the nearest charging station. He does the login and successfully enters into his account. Then he clicks on the "List" button to visualize a list of charging stations sorted by the distance. He chooses the nearest charging station and books the charge selecting the earliest time frame available.

D. Driver starts the charging process at the station

Driver Z has already booked a charge on the eMall platform. He arrives at the chosen charging station by time, and he plus the vehicle in the socket assigned by

his reservation. Then he confirms to start the charging process on his mobile application. At this point, the display of the charging column shows him the remaining charging time. Now the car is locked there, and Driver Z is free to go to deal with personal issues while leaving the car at the charging station.

E. Driver receives suggestions about going to charge the vehicle

The eMall system checks the status of the battery and the schedule of Driver Y when he begins his journey. The sensor detects the low battery state of the vehicle, and the system is notified about this condition. After checking the calendar and the navigation system of the driver, the system notices that he will arrive nearby a certain charging station with available charging sockets. At this point Driver Y receives a notification on his smartphone with the suggestion to go at that charging station and charge the vehicle. The charging station's location is reported too.

F. Operator wants to check the status of the charging station

Operator X wants to check the status of the charging station managed by him. He logs into his operator account, and then he visualizes the status page of the charging station: the information shown includes its location, its external status (number of charging sockets available, their type, their cost, the special offers, and the estimated time until finishing the charge), and its internal status (the amount of energy available in its batteries, number of vehicles being charged, the time left and the amount of power absorbed).

G. Operator wants define a special offer

Operator W wants to promote a special offer in the charging station managed by him. He logs into his operator account, and selects the option to manage the charging offers for that station. He decides a new cost for a certain charging type, with the starting date and the end date, and he confirms the modification. Now the system displays this offer on the charging station page.

H. Operator wants to acquire energy

Operator V manages a charging station, and he wants to acquire energy for it. He logs into his operator account, and he checks the list of the current prices of energy published by the Distribution System Operators. After thinking on the options, the operator V decides from which DSO he would like to acquire energy, and select it from the list. He is asked to input the quantity of energy he wants to acquire, and the system shows the summary of his operation. After confirming, the request of the operator will be sent to the chosen DSO and processed soon.

I. Operator wants to decide the energy source for charging

Operator U wants to manually decide where to get energy for the charging station. He logs into his operator account, and he visualizes the current schedule in the main page of the charging station: the energy source is determined dynamically unless any human operator decides to change it. Operator U selects one of the available alternatives from the list displayed by the system, and the change is applied after his confirmation.

2.1.2 Class Diagram

The class diagram of the system shows the relevant entities and their relations.

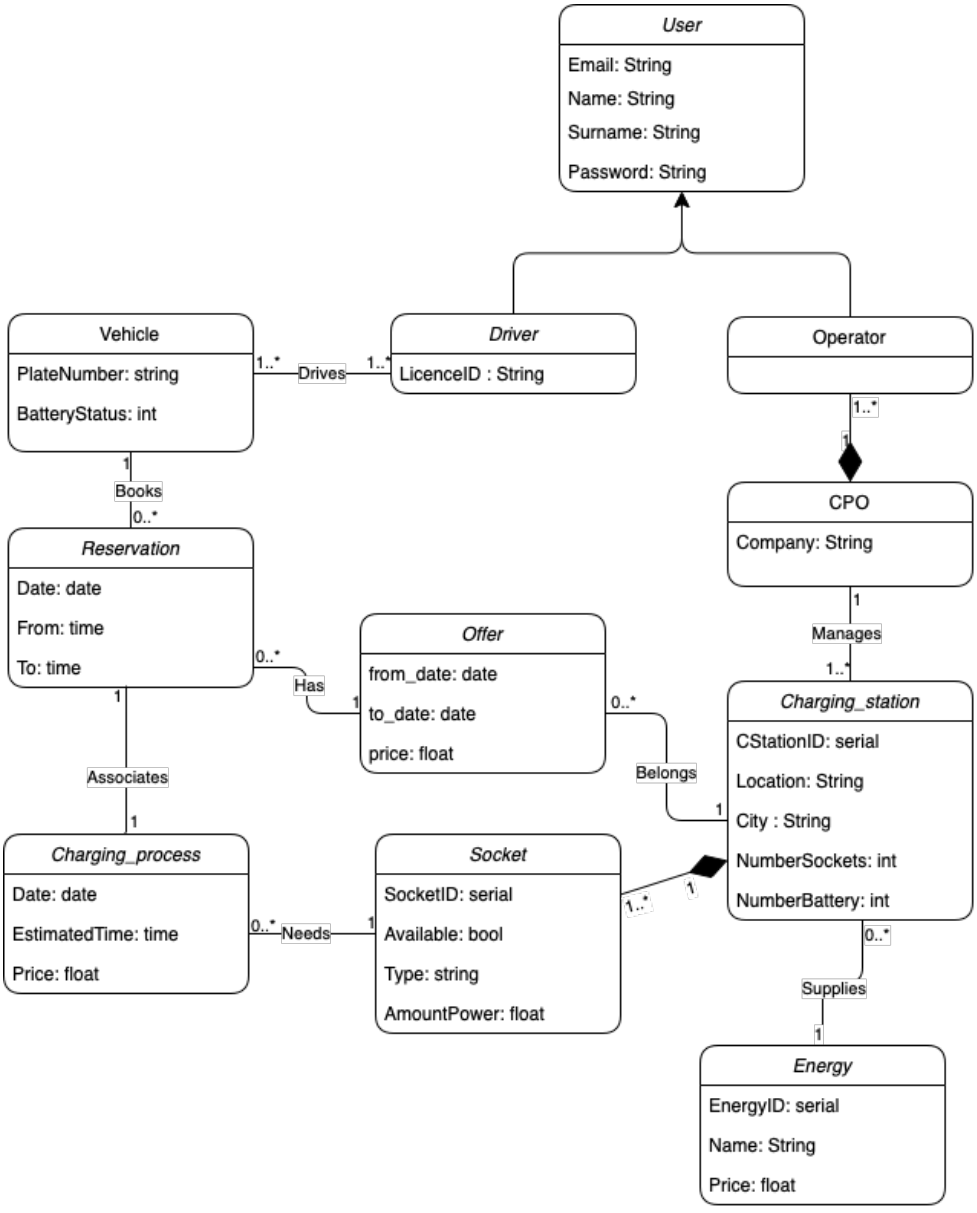

Figure 1: Class Diagram representing the system

2.1.3 Statecharts

The state diagrams presented in this part have the objective to better represent some events of the system by defining the state transitions.

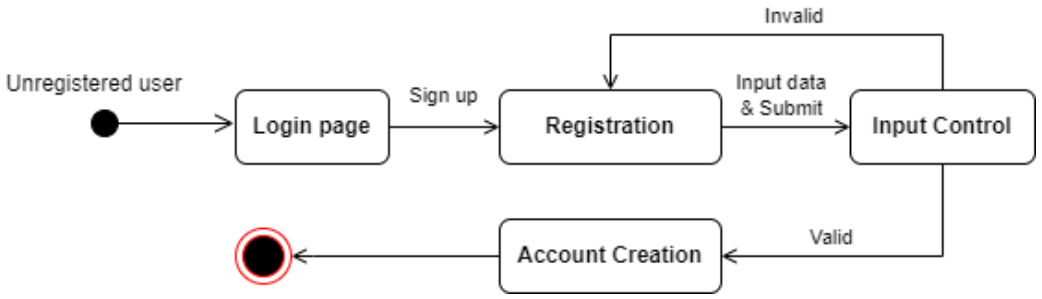

Figure 2: State diagram of the registration process. When the operation is completed, the user is redirected to the Login page.

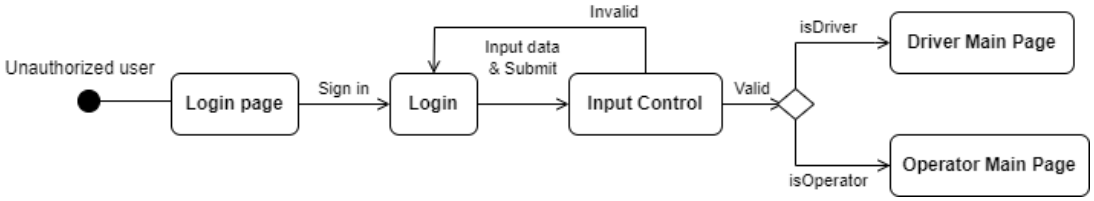

Figure 3: State diagram of the login process. If the authentication is successful, the user is redirected to the corresponding Main page with respect to his role (driver or operator).

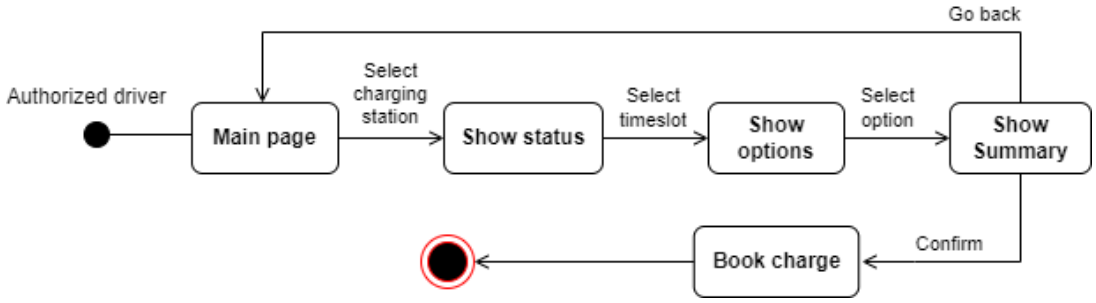

Figure 4: State diagram of the booking process. The authorized driver has already logged in, and he wants to book a charge. He is asked to choose the charging station, the time slot and the charging options respectively. After the completion of the process, the driver is redirected to the Main page and he can visualize the reservation in his profile.

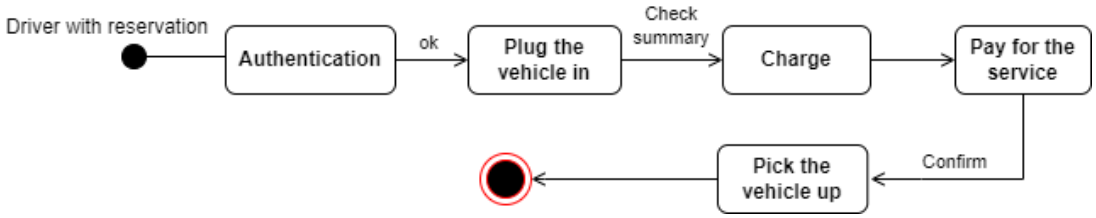

Figure 5: State diagram of the charging process. The driver has booked a time slot for charging, and he arrives at the charging station. He authenticates at the charging column, and plugs the vehicle in to start charging. When the operation is done, the driver is asked to pay for the charge before picking up the vehicle.

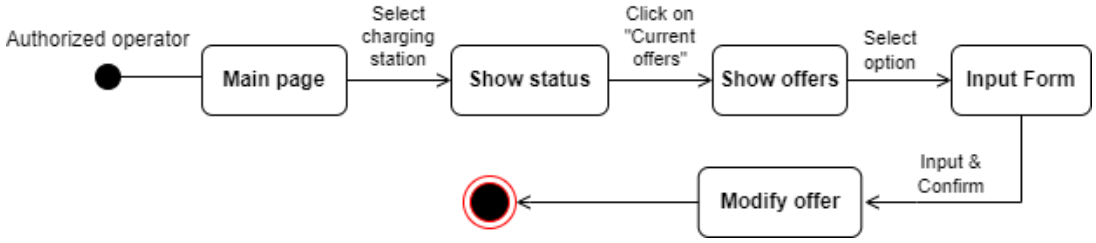

Figure 6: State diagram about promoting a new offer for the charging station. The authorized operator is asked to choose the charging station and to set the offer's price, starting and end date. At the end of the process, the operator is redirected to the Main page.

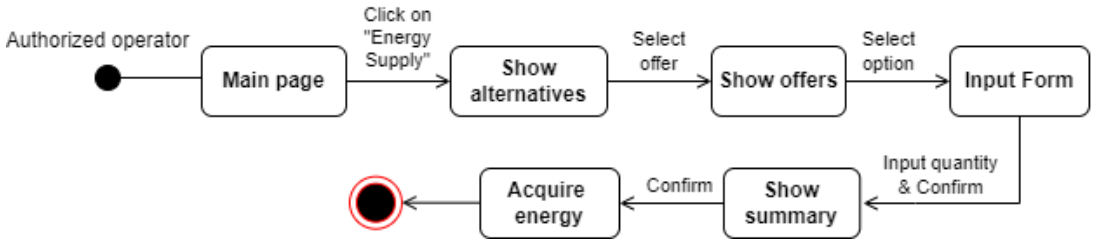

Figure 7: State diagram of the energy acquisition process. The authorized operator visualizes the offers published by the DSOs, and submits an energy acquisition request with the required data.

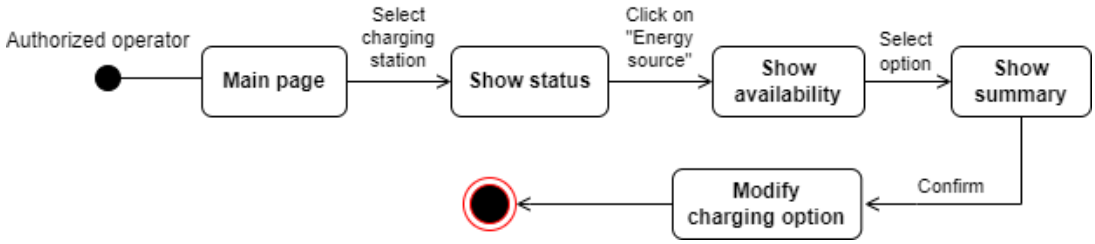

Figure 8: State diagram about changing the energy source for the charges. The operator has logged in and he checks the availability of energy sources of the charging station. He then chooses the option to modify the charging sources in use.

2.2 Product functions

In this section, a detailed description of the main functionalities of eMall is reported.

A. Give information about the charging stations to users

Giving information about the charging stations is at the basis of the system, and the given information is different according to the type of user.

The system will show to the drivers the location of nearby charging stations, their cost and eventually the available special offers. The drivers will know the number and type of available charging sockets in the chosen station, if all sockets of the chosen type are occupied, and the system will also show the estimated amount of time to have it be freed.

On the other hand, the system will show to the operators the status of their charging

station, such as the amount of energy available in its batteries, the number of vehicles being charged, the amount of power absorbed by these vehicles and the time left to finish the charge.

B. Book a charge

The system will offer to the drivers the possibility to book a charge at a chosen charging station. This functionality will help them to avoid long queue of people waiting for a charging socket, and to better organize their schedule.

The system will present to the drivers the charging stations and their sockets' available timeframe. If the driver confirms a book, the system will reserve that socket in that timeframe.

C. Charge a vehicle

Allowing to charge a vehicle is one of the most important functionalities of the system. Thanks to the interaction between eMSPs and CPOs, the system will start a charging process at a certain station according to the amount of power supplied by the socket. The process will also be monitored by the system to infer when the battery is full. When the process is finished, the system will notify the driver.

D. Suggest the users to charge the vehicle

The system will be able to suggest the driver to charge the vehicle actively, according to some monitored aspects. For example, checking the status of the vehicle's battery allows the system to notify the driver when the battery level is low. The system will also recommend some special offers of charging stations after checking the availability of the charging slots at these stations.

If the driver gives consent, the system will provide the possibility to make suggestions refer to the driver's schedule, by getting access to his/her calendar and navigation system.

This functionality can remind the driver of charging the vehicle without introducing much interference to his daily routine.

E. Manage the charging station

An important aspect of the system is to help the CPOs to manage the owned charging stations.

A charging station needs to acquire energy from external DSOs. So it is important to decide when, how much and from which DSO to acquire energy. The system will be able to make these decisions in an autonomous way by checking the availability of the DSOs and the current energy information acquired by them.

Moreover, the system will offer the functionality to dynamically decide the cost of a charge and set special offers based on the previous information. And when a vehicle starts charging, the system will monitor the process and decide dynamically where to get energy for the charge, such as from station battery, DSO or a mix thereof according to availability and cost.

All these kinds of decision could be made by the system automatically, but it will also provide the possibility to the human operators to handle each option.

2.3 User characteristics

There are three types of actors that we consider in the eMall system:

1. **Unregistered user**
   An electric vehicle's driver that cannot use eMall's functionalities without registering to the eMall platform.

2. **Driver**
   An electric vehicle's driver, registered to the platform, who can use the system to find charging stations and charge his/her electric vehicles, to book charges and to receive suggestions.

3. **Operator**
   A registered user who works for a specific CPO. The Operator can use the system to manage the charge stations of the belonged CPO and to receive current energy information from DSOs.

2.4 Assumption, dependencies and constraints

2.4.1 Domain assumptions

Identifier	Description
D1	Each electric vehicle has an unique number plate for identification
D2	Drivers will go to charge in the charging station indicated by their reservation
D3	Drivers will arrive at the chosen charging station in time for the reservation
D4	Drivers will complete the charging process within the booked time slot
D5	Drivers makes a significant use of their calendar
D6	Drivers are able to pay for the obtained service
D7	The information offered by the external DSOs is reliable
D8	The charging stations work correctly
D9	The geolocation obtained from external API is reliable
D10	Drivers own a valid driving licence

2.4.2 Constraints

Identifier	Description
C1	The permission of acquiring personal data (e.g. calendar and navigation system) must be given by the driver
C2	Users have a device with properly working internet connection

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The eMall program consists in a mobile application compatible with different operative systems such as iOS, Android and HarmonyOS. The application is designed to be used by both Drivers and Operators, and the user interface of the application is supposed to be simple to understand (e.g. functionalities can be accessed through clicking on common icons) and user-friendly, in order to maximize the usability and the user experience.

3.1.2 Hardware Interfaces

The hardware interface required to access to the eMall system is a working device (a mobile phone or a tablet) with internet connection that can successfully install the application. It must have fundamental sensors such as GPS and Bluetooth: the first is used for determining the location of the User and the second for the connection between the vehicle and the driver's device.

3.1.3 Software Interfaces

The application makes use of an external API for geo-localisation. It provides a map for Drivers to identify their actual position and the charging stations nearby, making the interaction more explicit. In addition to this, it uses the Bluetooth API to connect to the Driver's vehicle in order to retrieve useful data for the system.

For what concerns the server side of the application, the system requires the implementation of a DBMS to store the data.

3.1.4 Communication Interfaces

Many functionalities offered by the system relies on the communication with other services. Thanks to external APIs, e-Mall employs different interfaces:

- **Retrieval of data of a charging station**
  The interface is able to respond with an overview on the status a specific charging station, this is retrieved through the charging station's unique identification number.

- **Retrieval of data on DSOs**
  The interface is able to retrieve a list of available DSOs with the information they published. In this way the operators can acquire energy from any of them for the selected charging station.

- **Retrieval of map information**
  The eMall system provides a map for users to visualise their actual position and the available charging stations near them. In this case we need to interface with an external API which offers map and geolocation services.

- **Retrieval of vehicle data**
  The interface is able to retrieve relevant information about the user's actual guiding vehicle, such as its battery status. This is also made possible with an external API.

- **Retrieval of driver's schedule**
  The interface is able to communicate with the driver's calendar and the navigation system, to get information for making charging suggestions.

- **Payment support**
  The interface is able to handle the driver's payment with different methods.

3.2 Functional Requirements

3.2.1 Use cases

In this subsection, the main use cases of the system are reported.

U1. Driver Registration

Actor	Driver
Entry condition	The driver have not registered an account yet, and he is on the Login page of the application

Event flow	1. The driver clicks on the "Register" button in the Login page 2. The driver inputs his name and surname, the email address, the driving licence number and the password in the registration form 3. The driver clicks on the "Confirm" button 4. The system processes the submitted information
Exit condition	A new account is created for the driver and a success message is shown
Exceptions	1. The system shows an error message if the driver does not fill all the fields when submitting the form 2. The system shows an error message if the input data is invalid (e.g. the email address is already linked to another account)

U2. User Login

Actor	User (Driver or Operator)
Entry condition	The user has a registered account, and he is on the Login page of the application
Event flow	1. The user clicks on the "Login" button in the Login page 2. The driver inputs his email address and the password in the login form 3. The driver clicks on the "Login" button 4. The system processes the submitted information
Exit condition	The user is logged in, and he is redirected to the Main page

Exceptions	1. The system shows an error message if the driver does not fill all the fields when submitting the form 2. The system shows an error message if the input data is invalid (wrong email / password combination)

U3. Add a vehicle to the profile

Actor	Driver
Entry condition	The driver is logged in and he is on the Main page of the application
Event flow	1. The driver clicks on the "Profile" button in the Main page 2. The driver selects the option to add a vehicle to his personal profile 3. The system shows the form for adding a vehicle 4. The driver inputs the required data, such as the vehicle's plate number and the owner 5. The driver clicks on the "Confirm" button 6. The system processes the submitted information
Exit condition	The vehicle is successfully added to the driver's profile, and he can book a charge for it now
Exceptions	The system shows an error message if the input data is invalid or missing

U4. Visualize charging stations nearby

Actor	Driver
Entry condition	The driver has a registered account, and he is logged in
Event flow	1. The driver is redirected to the Main page of the application 2. The system shows the map with the charging stations located in the area 3. The driver selects a charging station 4. The system redirects the driver to the charging station's status page
Exit condition	The charging station's status is shown

U5. Book a charge

Actor	Driver
Entry condition	The driver has a registered account, he is logged in and he is at the Main page of the application
Event flow	1. The driver selects a charging station from the map or from the list displayed in the Main page 2. The system shows the status of the selected charging station and the available charging options 3. The driver selects the desired charging socket type 4. The driver selects a time slot from the list 5. The driver selects the vehicle he wants to charge 6. The driver clicks on the "Confirm" button 7. The system processes the submitted information 8. The system shows the summary of the operation
Exit condition	The driver successfully makes a reservation, which is added to his booking history
Exceptions	1. The system shows an error message if the data provided by the driver is invalid or missing 2. The system shows an error message if the chosen slot is not longer available at the moment of the form submission

U6. Start charging at the station

Actor	Driver
Entry condition	The driver is logged in, he has already booked a charge and he arrives at the charging station in time
Event flow	1. The driver checks the charging socket from his reservation summary 2. The driver plugs the vehicle in the correct socket 3. The driver confirms the starting of the charge on his mobile application 4. The system checks if the driver's reservation is valid 5. The system verifies the vehicle's status and locks the vehicle 6. The system shows the charging operation summary on the display
Exit condition	The vehicle starts charging
Exceptions	1. The system shows an error message if the driver's reservation is invalid 2. The system shows an error message if the vehicle is not correctly identified

U7. End charging

Actor	Driver
Entry condition	The driver's vehicle has finished to charge at the station
Event flow	1. The system notifies the driver about the status of the vehicle 2. The driver arrives at the charging station, and confirms the end of the charging operation on his mobile device 3. The system verifies the charging status 4. The system shows the operation summary and the amount to pay 5. The driver pays for the operation using credit/debit card or mobile payment methods 6. The system handles the payment
Exit condition	The system unlocks the vehicle, and the driver picks it up
Exceptions	1. The display shows an error message if the driver does not authenticate correctly 2. The display shows an error message if the payment fails

U8. Send charging suggestions

Actor	Driver
Entry condition	The driver is logged in, and the system detects the vehicle's low battery status
Event flow	1. The system checks the calendar of the driver to get information about his schedule 2. The system checks the navigation path of the driver 3. The system finds the charging stations near to the path of the driver 4. The system checks the selected stations' availability according to the driver's schedule 5. The system finds the suitable charging station and notifies the driver
Exit condition	The driver receives the suggestion of the system

U9. Define a special offer

Actor	Operator
Entry condition	The operator is logged in and he is at the Main page of the application
Event flow	1. The operator selects the charging station he wants to manage from the list reported in the Main page 2. The operator selects the option to make a new offer 3. The system displays a form for the creation of the offer 4. The operator defines the offer's type and price 5. The operator selects the starting date and the end date of the offer 6. The operator clicks on the "Submit" button 7. The system processes the submitted information and shows the operation summary
Exit condition	The new offer is published successfully
Exceptions	1. The system shows an error message if the chosen combination of starting date and end date is invalid 2. The system shows an error message if the value of price is invalid

U10. Modify the energy source used for charging

Actor	Operator
Entry condition	The operator is logged in and he is at the Main page of the application
Event flow	1. The operator selects the charging station he wants to manage from the list 2. The system displays the current status of the chosen charging station 3. The operator selects to change the energy source used for charging 4. The system shows a list of available options 5. The operator chooses an option and clicks on the "Confirm" button 6. The system processes the submitted information
Exit condition	The current energy source is modified
Exceptions	The system shows an error message if the option turns to be invalid at the moment of the submission

U11. Modify the charging price

Actor	Operator
Entry condition	The operator is logged in and he is at the Main page of the application
Event flow	1. The operator selects the charging station from the list 2. The system displays the current status of the chosen charging station 3. The operator selects the type of charge he wants to modify the price 4. The operator defines the new charging cost 5. The operator clicks on "Submit" button 6. The system processes the submitted information
Exit condition	The price is modified and the page is refreshed to show the updated information
Exceptions	The system shows an error message if the input data is invalid (wrong format)

U12. Acquire energy from DSOs

Actor	Operator
Entry condition	The operator is logged in and he is at the Main page of the application
Event flow	1. The operator clicks on the "Acquire energy" button 2. The system collects the list of offers published by the DSOs and displays it 3. The operator selects an available option 4. The operator decides the quantity of energy he wants to acquire and the related charging station 5. The system shows the operation summary 6. The operator clicks on the "Submit" button 7. The system processes the submitted information
Exit condition	The new offer is published successfully
Exceptions	The system shows an error message the input data is invalid

3.2.2 Use cases Diagrams

1. Unregistered Driver

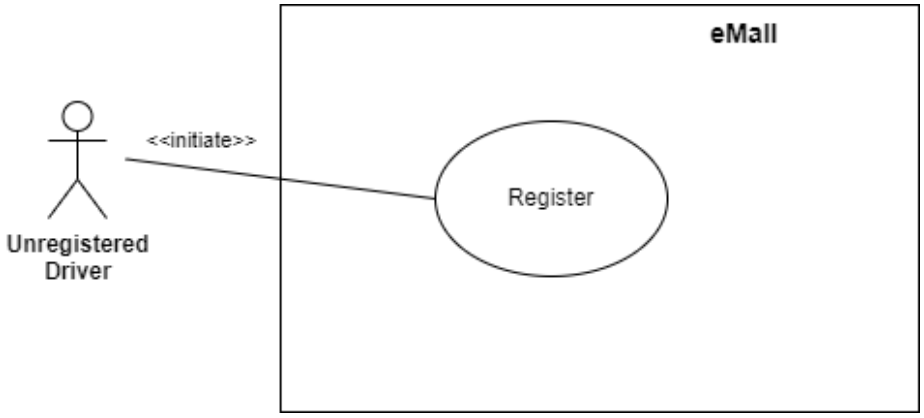

Figure 9: Use case diagram for an unregistered driver

2. Authenticated Driver

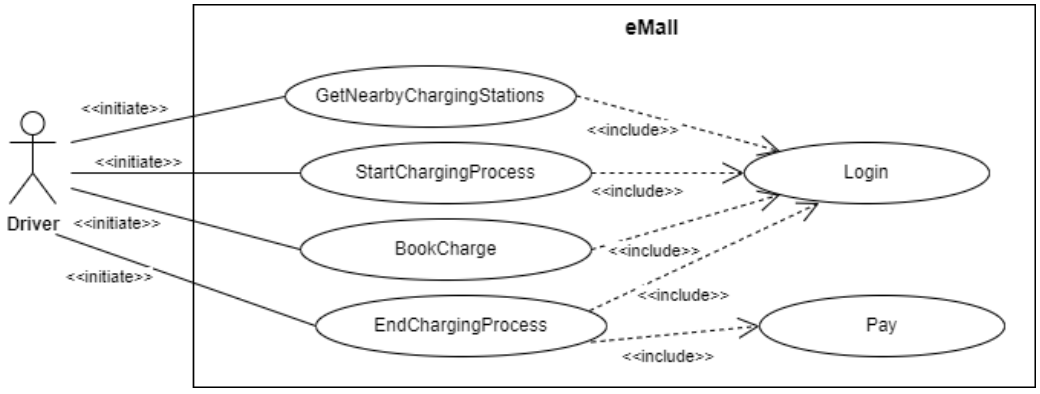

Figure 10: Use case diagram for an authenticated driver

3. Authenticated Operator

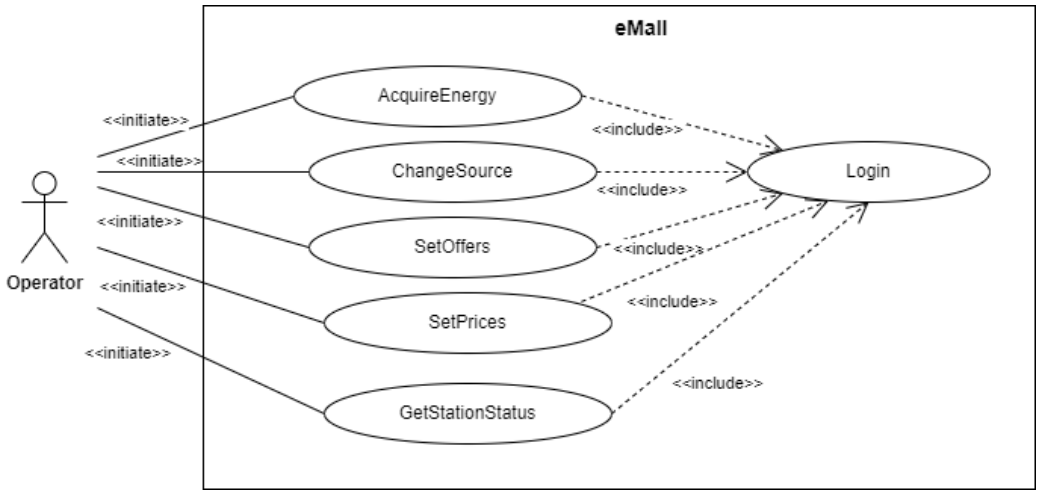

Figure 11: Use case diagram for an authenticated operator

3.2.3 Sequence Diagrams

In this section, the corresponding sequence diagrams for the use cases are presented. Overall, we consider that the actor has been logged in all cases except for "Registration" and "Login" cases.

1. Register

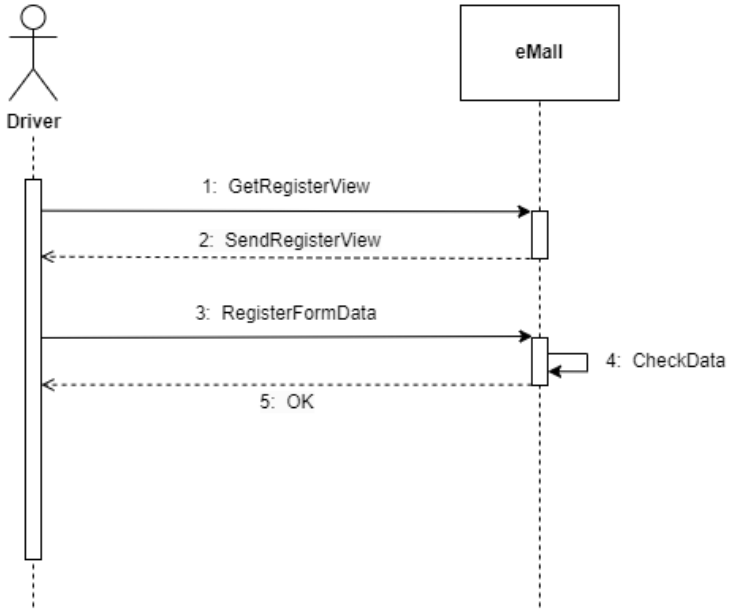

Figure 12: Sequence diagram for a Driver's Registration process

2. Login

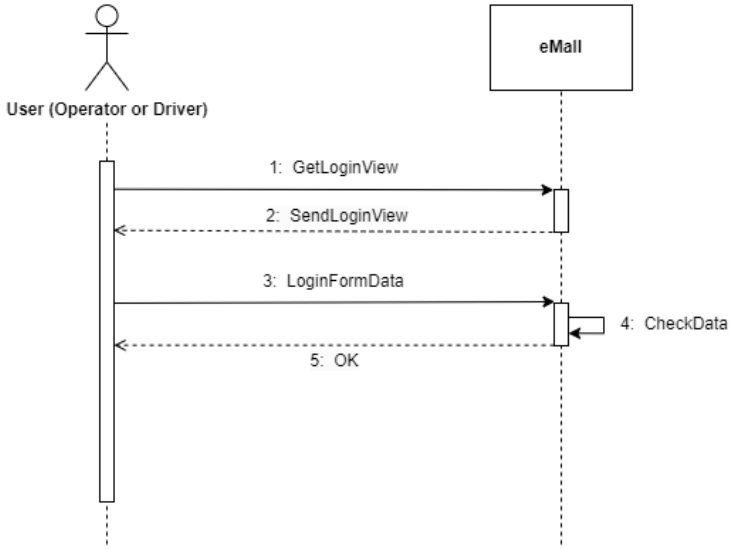

Figure 13: Sequence diagram for the Login process

3. Get charging stations

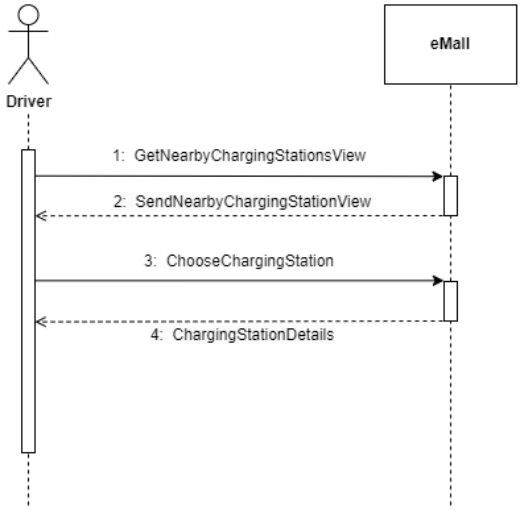

Figure 14: Sequence diagram for the process when a driver is searching for a nearby charging station

4. Book a charge

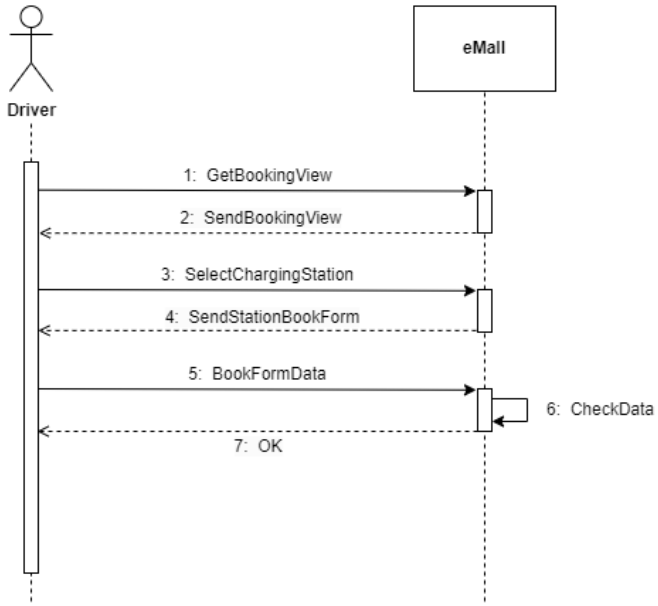

Figure 15: Sequence diagram for booking a charge by a driver

5. Start a charging process

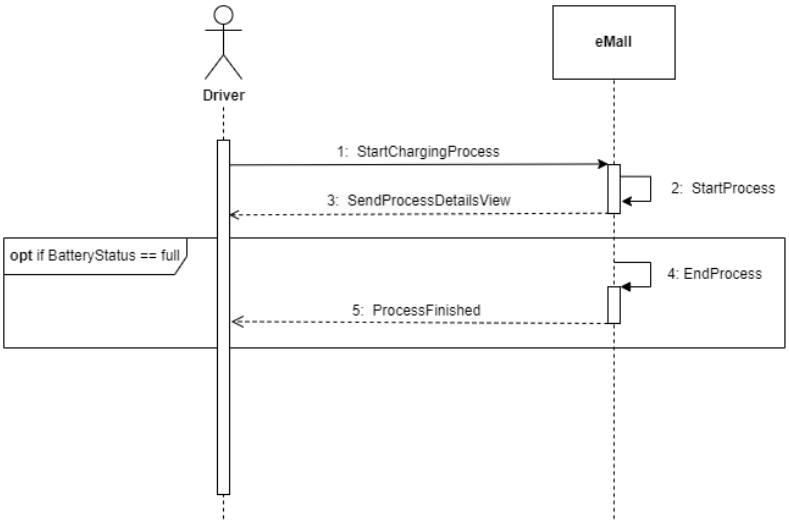

Figure 16: Sequence diagram for starting a charging process. The possibility of getting the battery full is also presented.

6. End a charging process

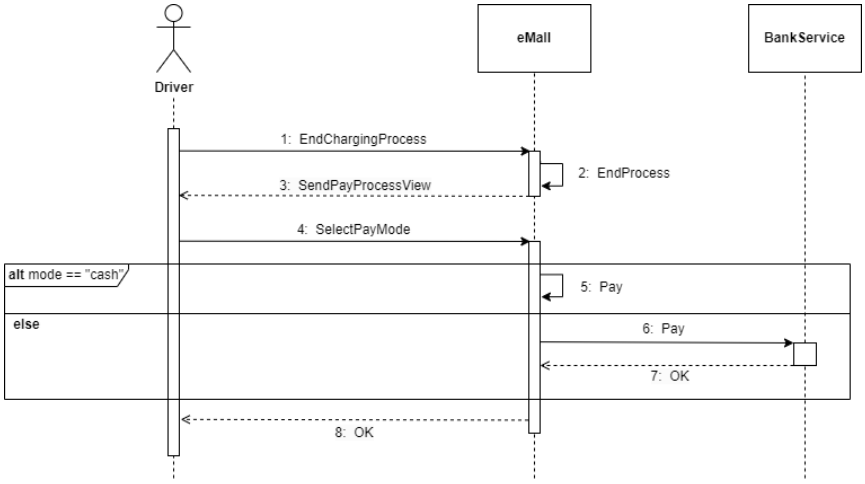

Figure 17: Sequence diagram for ending a charging process. The process of pay the charge is also included in the diagram and different methods of payment (Cash or Credit Card) are also presented.

7. Get a charging station's status

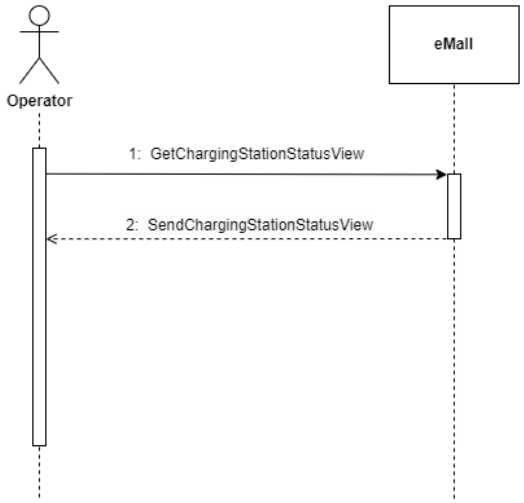

Figure 18: Sequence diagram for getting a charge station's status by an operator

8. Set an offer

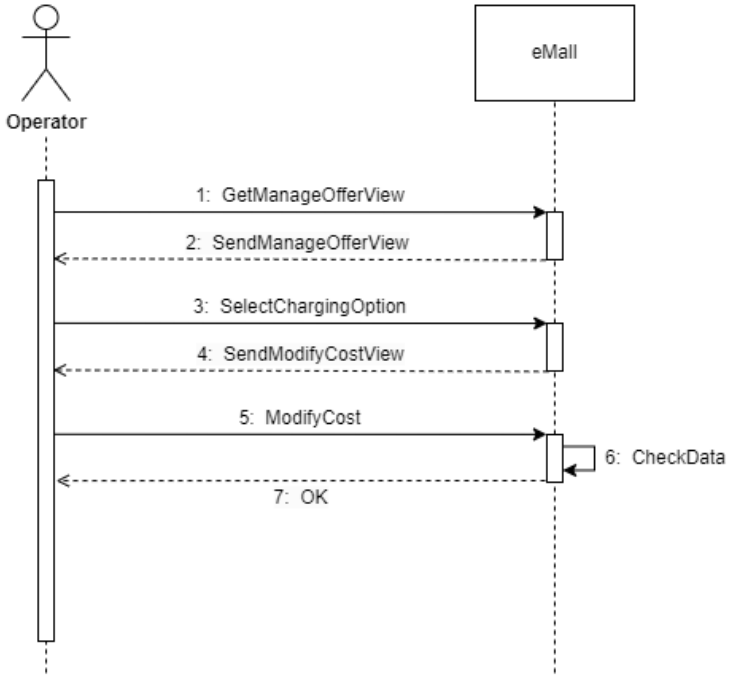

Figure 19: Sequence diagram for setting a special offer by an operator

9. Acquire energy

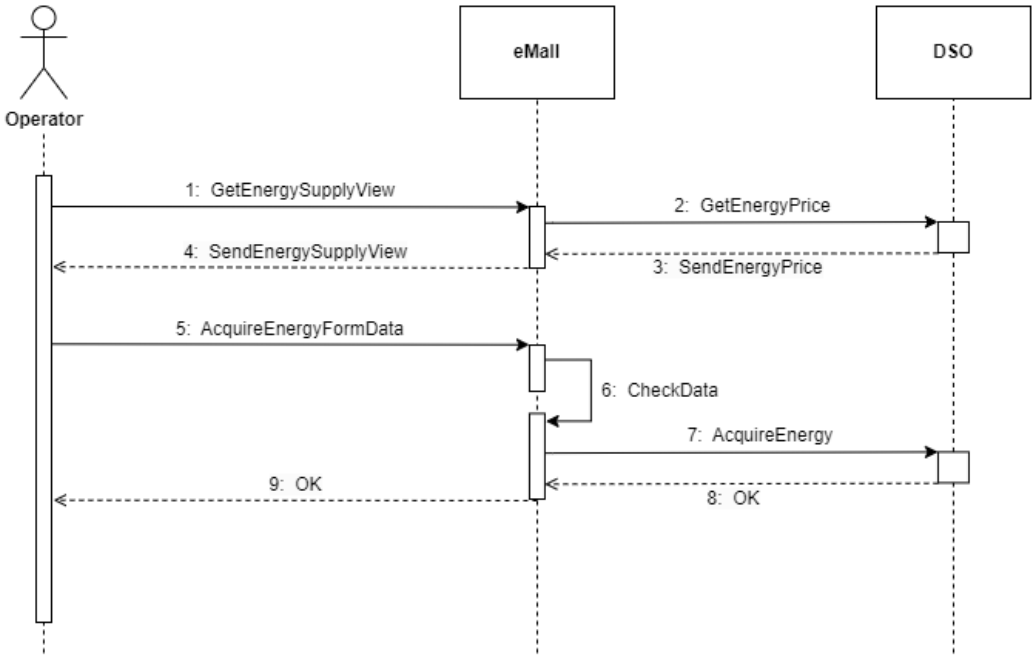

Figure 20: Sequence diagram for acquiring energy from external DSOs by an operator. The interactions between the system and a generic DSO are presented.

10. Change source

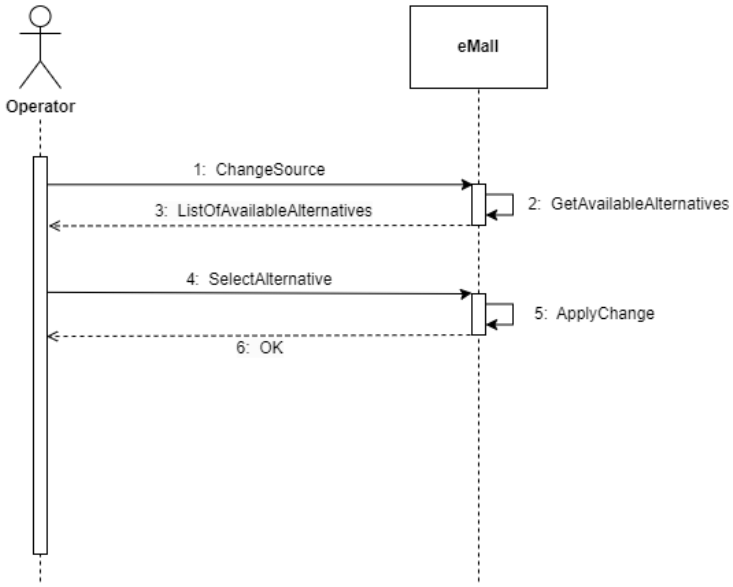

Figure 21: Sequence diagram for changing a station's source by an operator.

3.2.4 List of Functional Requirements

Identifier	Description
R1	The system shall allow an unregistered Driver to register and create an account
R2	The system shall allow a registered Driver to log into his account
R3	The system shall allow a registered Driver to delete his account
R4	The system shall allow a registered Driver to add one or more vehicles to his account
R5	The system shall allow a registered Driver to delete one or more vehicles from his account
R6	The system shall allow a registered Driver to book a charging slot of a station for a certain time frame
R7	The system shall allow a registered Driver to acquire information of a selected charging station
R8	The system shall allow a registered Driver to visualise the details of his charging process

R9	The system shall allow a registered Driver to visualize the current offers of charging stations
R10	The system shall allow a registered Driver to pay for the obtained service through an external API
R11	The system shall allow a registered Driver to cancel a reservation
R12	The system shall not allow a registered Driver to book for a charge when he has already a reservation
R13	The system shall allow a registered Driver to know about his actual position and nearby charging stations through an external API
R14	The system shall allow an Operator working for the CPO to log into his account
R15	The system shall allow a registered Operator to create an offer to the charging station managed by him
R16	The system shall allow a registered Operator to modify the charging source of the charging station managed by him
R17	The system shall allow a registered Operator to modify the price of the energy of a charging station managed by him
R18	The system shall allow a registered Operator to check the status of the charging station managed by him
R19	The system shall allow a registered Operator to visualise data provided by DSOs
R20	The system shall allow a registered Operator to choose from which DSO to acquire energy
R21	The system must be able to decide automatically where to get energy for charging
R22	The system must be able to notify the Driver when the charging process starts
R23	The system must be able to notify the Driver when the charging process ends
R24	The system must be able to suggest the Driver to charge the vehicle, the suggestion is based on information retrieved through external APIs
R25	The system must be able to notify user in case of exceptions
R26	The system must be able to notify user on successful actions
R27	The system must be able to retrieve a map through an external API

R28	The system shall allow a registered Driver to start his charge
R29	The system shall allow a registered Driver to finish his charge

3.3 Mapping on Goals

The following traceability matrix summarizes the mapping between the goals and the requirements which can guarantee the achievement of the goals under some domain assumptions.

Goals	**Assumptions**	**Requirements**	**Use Cases**
G1	D8, D9	R1, R2, R7, R9, R13, R25, R26, R27	U1, U2, U4
G2	D1, D2	R1, R2, R4, R6, R7, R9, R11, R12, R25, R26	U1, U2, U3, U4, U5
G3	D2, D3, D4, D8	R1, R2, R4, R6, R7, R8, R22, R25, R26, R27, R28, R29	U5, U6, U7
G4	D6, D8	R1, R2, R4, R5, R6, R8, R9, R10, R23, R25, R26, R28, R29	U7
G5	D1, D5, D9	R1, R2, R4, R5, R7, R9, R13, R24	U2, U3, U8
G6	D8	R14, R18, R20, R25, R26	U2, U10
G7	D8	R14, R17, R18, R21, R25, R26	U2, U9, U11
G8	D7, D8	R14, R15, R17, R18, R19, R25, R26	U2, U12

3.4 Design Constraints

3.4.1 Non-Functional Requirements

Identifier	Description
NR1	The system has to provide a feedback in 5 seconds
NR2	The system must tolerate slow internet connection
NR3	Failures in the system must be fixed within 2 hours
NR4	The system should be available 99% of the time
NR5	The system must be easy to understand and to use
NR6	The system must provide an user-friendly user interface
NR7	The system must support different operating systems including iOS, Android and HarmonyOS (Versions ?)
NR8	The system must support different devices in particular smartphone and tablet with different screen sizes
NR9	The hardware on which the system runs must have sensors in particular GPS and Bluetooth
NR10	The device while running the system must be connected to internet
NR11	The system should never allow unregistered user to use its functionalities
NR12	The system must protect sensitive information provided by users
NR13	personal data and sensitive personal data collected from the system shall be treated in compliance with GDPR
NR14	The system must not let unauthorised users to access sensitive information
NR15	When making use of external APIs, the system must follow their regulation and guidelines
NR16	The system should be easily modifiable for future improvement and adaptation to the evolving requirements or changes in the environment

3.4.2 Standards compliance

All users' personal data is treated in compliance with the GDPR (NR13). The employment of external APIs must be done following regulations and guidelines (NR15).

3.4.3 Hardware limitations

The device on which the application runs can be a smartphone or a tablet (NR8). It must be able to connect to Internet (NR10) and it must have the required sensors, in particular GPS and Bluetooth (NR9) for the correct working of the application.

3.5 Software System Attributes

3.5.1 Reliability

The system should be highly reliable, and running without failure most of the time. If failures happen, it should be able to recover quickly (NR3). To achieve this, regular maintenance should be done.
Since the system relies also on several external API which do not depend on our system, it must not completely fail when one of those fail.

3.5.2 Availability

The aimed availability for the system is 99%: in fact, the system must be available and running most of the time, especially during the business hours (NR2, NR3, NR4), although it is not a big issue for the system to have some short downtime. In the case that a Driver should book for a charge, if he is not in a rush, he can simply do it later. If the system is down while the driver is charging, he can complete the charging process and pay for the service later.

3.5.3 Security

The information that the system is handling can be either public or sensitive. The latest need to be protected (NR11, NR12, NR13, NR14, NR15). To do so, the system can make use of different approaches such as limiting the access by setting different authorisation level so that unauthorised users are denied to access. In the case of password, it is possible to scramble it through encryption algorithm, so that it is unreadable by unauthorised ones.

3.5.4 Maintainability

The system should be easily modifiable for future improvement and adaptation to the evolving requirements or changes in the environment (NR16). In order to achieve this, it is important to keep in line the document while realizing the application.

Moreover, it is fundamental to re-use design patterns which helps to prevent subtle issues that can cause major problems, and to improve code readability.

3.5.5 Portability

The eMall application should be installed on a device to be used. The User interface and functionalities of the system should work well on different screen sizes of different devices, mobile phone or tablet, with different operating systems such as iOS, Android and HarmonyOS (NR7, NR8).

4 Formal Analysis using Alloy

This section contains a formal analysis of the system performed with the declarative language Alloy. It is based on the description of the model of the system with its components and the interactions between them, with a particular focus on the constraints of the system. The validity of the

4.1 Alloy code

```
//email is an identification for all the users
sig Email{}
//password for the user
sig Password{}
//lincence number of a driver
sig LicenceId{}
//plate number of a vehicle
sig PlateNumber{}

//price of energy or of
sig Price{}
//date for reservation and charging process
sig Date{}
//time slot for reservation
sig TimeSlot{}
//location of charging station
sig Location{}
//energy supplied by charging station
sig Energy{}


//charging type of a socket in the charging station
abstract sig ChargeType{}
one sig SLOW extends ChargeType{}
one sig FAST extends ChargeType{}
one sig RAPID extends ChargeType{}

//boolean
abstract sig Bool {}
one sig TRUE extends Bool {}
one sig FALSE extends Bool {}

//all users of e-Mall
```

```
abstract sig User {
  email: one Email,
  password: one Password
}

//driver: user of electric vehicles
sig Driver extends User {
  licence_id: one LicenceId
}

//operator: charging station administrator working for a CPO
sig Operator extends User {
}

//CPO: company which owns and manages charging stations
sig CPO{
  operators: some Operator
}

//vehicle that belongs to a driver
sig Vehicle {
  plate_number: one PlateNumber,
  battery_status: one Int,
      drivers: some Driver
}{
  battery_status >= 0
}

//charging stations where drivers go to charge their vehicles
sig ChargingStation {
  location: one Location,
  number_socket: one Int,
  sockets: some Socket,
  energy_price: one Price,
  cpo: one CPO
}{
  number_socket = #sockets
}

//socket used for charging a vehicle
sig Socket {
  available: one Bool,
  type: one ChargeType,
  amount_power: one Int,
  energy: one Energy
}{
```

```
  amount_power >= 0
}

//offer set by a charging station with expire time
sig Offer {
  from_date: one Date,
  to_date: one Date,
  price: one Price,
  cstation: one ChargingStation
}

//charging process of charging a vehicle
sig ChargingProcess{
  vehicle: one Vehicle,
  date: one Date,
  estimated_time: one Int,
  price: one Price,
  reservation: one Reservation,
  socket: one Socket
} {
  estimated_time >= 0 and vehicle = reservation.vehicle and date =
   reservation.date and socket = reservation.socket
}

//a reservation by a driver to charge his vehicle
sig Reservation{
  vehicle: one Vehicle,
  date: one Date,
  time_slot: one TimeSlot,
  offer: lone Offer,
  socket: one Socket,
  charge_quantity: one Int
} {
  charge_quantity > 0 and socket in offer.cstation.sockets
}

/* fact */
//all the operators must belong to at least one CPO
fact allOperatorBelongsToCPO {
  all op : Operator |
  (one cpo: CPO | op in cpo.operators)
}

//all the sockets must belong to at least a charging station
fact allSocketBelongsToCS {
  all s: Socket |
```

```
  (one cs: ChargingStation | s in cs.sockets)
}

//one socket must not have different reservations at the same time
fact uniqueTimeSlotForSameSocket  {
  no disjoint r1, r2: Reservation | r1.time_slot = r2.time_slot and
   r1.socket = r2.socket and r1.date = r2.date
}

//a vehicle must have one and only one reservation on the same date
   and in the same time slot
fact uniqueReservationForSameVehicle {
  no disjoint r1, r2: Reservation | r1.time_slot = r2.time_slot and
   r1.vehicle = r2.vehicle and r1.date = r2.date
}

//there are no different charging stations in the same location
fact uniqueLocation {
    no disjoint cs1, cs2 : ChargingStation | cs1.location = cs2.
   location
}

//a reservation must have one and only one charging process
fact uniqueReservationForACProcess {
  no disjoint cp1, cp2 : ChargingProcess | cp1.reservation = cp2.
   reservation
}

//every identification of licence ID is associated to one and only
   one driver
fact allLicenceIdBelongsToDriver{
  all li:LicenceId | one d:Driver | li = d.licence_id
}

//every plate number belongs to one and only one vehicle
fact allPlateNumberBelongsToVehicle{
  all pn: PlateNumber | one v:Vehicle | pn = v.plate_number
}

//every email belongs to one and only one user
fact allEmailBelongsToUser{
  all e: Email | one u: User | e = u.email
}


/* pred */
```

```
//add a driver to an vehicle
pred addDriver [v1, v2: Vehicle, d: Driver] {
  v2.drivers =  v1.drivers + d
}

//del a driver to an vehicle
pred delDriver [v1, v2: Vehicle, d: Driver] {
  v2.drivers =  v1.drivers - d
}


/* assert */

//checking for operation correctness
assert delUndoesAddDriverToVehicle {
  all d: Driver, v1, v2, v3: Vehicle |
  (not (u in v1.drivers) implies
    (addDriver[v1, v2, u] and delDriver[v2, v3, u]
    implies
    v1.drivers =  v3.drivers
  ))
}
check delUndoesAddDriverToVehicle

//emails of users are unique
assert UniqueEmail {
   no disjoint u1, u2: User | u1.email = u2.email
}
check UniqueEmail

//every vehicle has at least one driver
assert everyVehicleHasADriver{
  all v: Vehicle |
  (some d: Driver |  d in v.drivers)
}
check everyVehicleHasADriver

//every charging station has at least one socket
assert everyChargingStationHasSocket{
  all cs: ChargingStation |
  (some s: Socket | s in cs.sockets)
}
check everyChargingStationHasSocket

//identifiers are all unique
```

```
assert UniqueIdentifier{
   no disjoint d1, d2: Driver| d1.licence_id = d2.licence_id and
   no disjoint v1, v2: Vehicle | v1.plate_number = v2.plate_number
}
check UniqueIdentifier

//every charging process must have one and only one reservation
assert everyChargingProcessHasReservation{
  all cp: ChargingProcess |
  (one r: Reservation | cp.reservation = r)
}
check everyChargingProcessHasReservation

//every charging station has at least one operator
assert everyCStaionHasOperator{
  all cs: ChargingStation|
  (some o: Operator | o in cs.cpo.operators)
}
check everyCStaionHasOperator

//all offer must be created by one and only one charging station
assert allOfferBelongsToCS{
  all of: Offer |
  (one cs: ChargingStation | of.cstation = cs)
}
check allOfferBelongsToCS


pred world {
    #Vehicle>0
  #Socket>0
  #Offer>0
  #Reservation = 1
  #ChargingProcess>0
  #ChargingStation>0
  #Operator = 2
}
run world for 3
```

4.2 Generated model

It is shown below a simple scenario, where a charging process is presented. In this instance there are 3 users: 1 driver and 2 operators. The vehicle of the driver is charged at a charging station after reservation. An offer is also applied on the

charging. The two operators work for the same CPO, which manage the charging station.

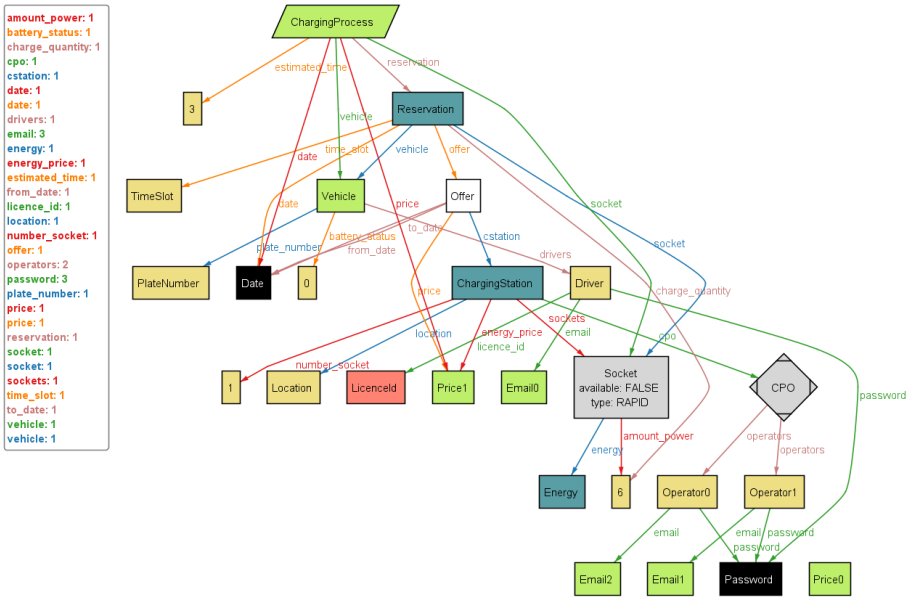

Figure 22: Generated Alloy model

Moreover, this instance satisfies all the following constraints:

- Every vehicle has a driver
- Every socket has a charging station
- All identifiers are unique
- Every charging process has a reservation
- Every charging station has at least an operator

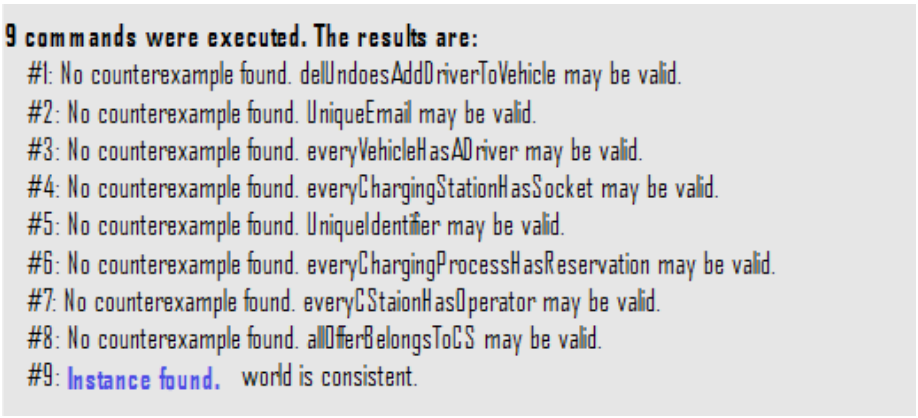

Figure 23: Validation of the Alloy model

5 Effort Spent

- Giulia Huang

Working Task	Time spent
Group discussion on the project plan	3h
Definition of project goals and scope	3h
Revision of project goals	1.5h
Group discussion on Section 2	3h
Creation of class diagram	4h
Group discussion on system requirements	2h
Definition of project requirements and assumptions	5h
Group discussion on mappings and constraints	3h
Group discussion on Alloy	1h
Coding for Alloy	15h
Review	2h
Group discussion for the final submission	2h
Total	44.5h

- Zheng Maria Yu

Working Task	Time spent
Group discussion on the project plan	3h
Definition of purpose and list of relevant terms	3h
Group discussion on Section 2	3h
Definition of scenarios	5h
Creation of statecharts	4h
Group discussion on system requirements	2h
Review of goals and requirements	2h
Definition of use cases	7h
Completion of diagrams	2h
Group discussion on mappings and constraints	3h
Group discussion on Alloy	1h
Restructuring the document	3h
Review	2h
Group discussion for the final submission	2h
Total	42h

- Linda Zhu

Working Task	Time spent
Group discussion on the project plan	3h
Definition of project scope	2h
Group discussion on Section 2	3h
Definition of product functions and users	4h
Group discussion on system requirements	2h
Group discussion on mappings and constraints	3h
Definition of sequence diagrams	6h
Completion of diagrams	2h
Group discussion on Alloy	1h
Coding for Alloy	12h
Group discussion for the final submission	2h
Review	2h
Total	42h

6 References

- Project specification: "Assignment R&DD A.Y. 2022-2023 v3"
- Software Engineering 2 Course slides A.Y. 2022-2023