Politecnico di Milano

Computer Science and Engineering's master degree course
Department of Electronics and Information

eMall – e-Mobility for All

Requirements Analysis and Specification Document

Version 1.5

Release Date:

Customer: Eng. Elisabetta DI NITTO

Authors:

- Eng. Roberto CIALINI Id. 933385
- Eng. Umberto COLANGELO Id. 935073
- Eng. Vittorio LA FERLA Id. 224509

Contents

1 Introduction

1.1 Purpose

Nowadays sustainability is one of the most important and debated topics in our society. In fact, in the next few years we are going to deal with a huge green transition to limit our carbon footprint on the planet, such as in the transportation field, which finds itself as one of the main contributors of global warming. In this direction, in recent years the old motor vehicles running on gasoline are leaving space for electricity-powered vehicles, even though there are several central aspects to deal with in order to let the electric vehicles be competitive with the old vehicle generation. In this direction, the goal is to create a fully operative and diffused infrastructure for the fast charging of the batteries, which is one of the main limitations for the final customer. In fact, the batteries need to be charged often and nowadays the task of finding an available charging spot is not as easy as it seems.

With this issue in mind, e-MALL is an operating system, itself composed of two subsystems, whose goal is to offer a way to find available charging stations for electric vehicles, offering at the same time to the user the possibility to access to several features such as the reservation of a specific socket at a certain timeframe or the reception of personalised proactive suggestions by the system.

This document will further explain in detail goals and requirements put on the system to be with this purpose of guiding the development.

1.1.1 Goals

Goal	Description
G1	Allow Drivers to find available charging stations, the nearest ones, their energy cost per unit (MW/h), any special offer they have, their available sockets and a path to reach the selected one
G2	Allow Drivers to book the desired charging station for a certain timeframe and eventually cancel it

G3	Allow Drivers to start and monitor the charging process at a certain station and to be notified when the charging process is finished
G4	Allow Admin to track data regarding internal and external status of each charging station
G5	Allow the Admin to manually or automatically decide for each station where to get energy for charging (station battery, DSO, or a mix).
G6	Allow Admin to have access to DSOs information about the current price of energy and to decide from which to acquire energy
G7	Allow Drivers to receive recommendations based on the status of the battery of his active vehicle and the schedule of the user and his current position

1.2 Scope

While there are several stakeholders to consider, this document is only concerned about two actors: Drivers and Administrators. The former is the final user, or rather the one who interacts with the eMSP to have the possibility to book the battery recharge of his vehicles. Instead the role of the system Administrator mainly concerns to monitor the correct behaviour of the system and to take strategic decisions.

The main system is divided into two subsystems: the eMSP and CPMS. The eMPS is designed to be an interface and to communicate both with the Driver and the Administrator, driving their requests. The CPMS, instead, is modelled on the OCPI 2.2.1 protocol and is referred to as a specific CPO. The main task of the CPMS is to supply information about its CPO charging stations to the eMSPs it is linked to, both for the Driver and the Administrator usage.

Although CPO and DBO are mentioned in this document along with the other entities described before, we will not consider either their internal system or their decision making.

This application is supposed to work properly in every situation in which are well defined the previously mentioned rules, with no limitation to the metropolitan areas.

1.2.1 World Phenomena

Identifier	Description
WP1	Charging stations are owned by a CPO, can have a battery and provide energy through sockets of different speeds
WP2	DSOs provide energy to CPO stations
WP3	Drivers use the charging stations to charge their vehicles
WP4	CPO Administrators manage their CPMS

1.2.2 Shared Phenomena

Identifier	Description
SP1	User registers an account
SP2	User pays the cost for recharging
SP3	User chose the charging station he prefers from the available ones
SP4	Admin has access to the all the data regarding of CPMS
SP5	User books for specified amount of time a socket in a recharging station
SP6	System visualises data about recharging station based on the CPOS information
SP7	User decides to end the charge of the battery
SP8	System make recommendation on the best charging station available
SP9	System sends notification to user
SP10	Admin manually selects from which DSO to acquire energy from or chooses this selection to be automatic
SP11	Admin manually selects for each station where to get energy from or chooses this selection to be automatic
SP12	System know when the battery is fully recharged

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

Definitions	Descriptions

Driver Identifier	To identify a specific driver, this could be an identification number such as his/her SSN
Car Identifier	To identify a specific car, this could be the licence plate
Station Identifier	To identify a specific charging station

1.3.2 Abbreviations

RASD	Requirements Analysis and Specification Document
WP	World Phenomena
SP	Shared Phenomena
GX	Goal number X
DX	Domain assumption number X
RX	Requirements number X
eMALL	e-Mobility for All
EV	Electric Vehicle

1.4 Revision History

1.5 Reference Documents

The specification document “Assignment RDD AY 2022-2023_v3.pdf ”

1.6 Document Structure

This document is composed of six sections, detailed below.

In the first section the problem is introduced together with the goals of the project. Additionally, the scope of the project is specified along with the various phenomena occurring. Lastly, the necessary information to read the report is presented, such as definitions and abbreviations.

Section two contains an overall description of the system, including a detailing of its users and main functions. Moreover there is the class

diagram, descriptions of several scenarios, some statecharts and finally the domain assumptions made in this report.

In section three the requirements on the system are specified. This includes functional requirements, non-functional requirements and requirements on external interfaces. Furthermore use cases are described, with accompanying use cases and sequence diagrams. Section three also contains mappings of functional requirements to the goals of the system, and to the use cases.

Section four contains a formal analysis with the help of Alloy. Together with the Alloy code, the analysis objective is described.

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

Section six contains the references used.

2. Overall description

2.1 Product perspective

2.1.1 Scenarios

1. **Electric vehicle driver starts using the system**

The EV Driver Carlos wants to register to the service to have access at the several facilities it offers, such as finding a charging station and charging his vehicle outdoors. He launches the service and chooses to sign up, fulfilling the mandatory information required to access the service.

2. **Electric vehicle driver setting personalised data**

Jim, an electric vehicle driver, once he has registered to the system using his credentials, by selecting his profile and then from the section “Active Vehicle” he chose one of his vehicles and set it to be his Active Vehicle. In this way, the system automatically filters the stations according to the vehicle information, meaning that will show as clickable only the stations that have the socket compatible with the Active Vehicle.

3. **Electric vehicle driver wants to book a charging station**

John is an electric vehicle driver who needs to charge his car. After he logs in into the system using his user credentials, he has access to the homepage where all the stations available are displayed. The user can also decide to filter the station by selecting the feature he needs the most. Once John has selected the station, can click on “Book Now” to reserve the slot for a certain timeframe. Jack, another user that needs to charge his vehicle in the same period of time, won’t be able to book the same socket and will see it as already engaged.

4. **Electric vehicle driver wants to start the charging process**

Samantha, an electric vehicle driver, wants to charge her car. Once she reaches the parking spot and has plugged in the vehicle, she logs into the system with her user credentials and by clicking on his reservation on “My Reservations” and then clicking on “Start Now”, she confirms to begin the charging process. Samantha can also monitor the progress live.

5. Electric vehicle driver pays for the charge

Steve logged into the system and is using an available slot in a station to charge his vehicle. Selecting “Details” he can see in live the charging progress of the battery and can decide to stop it whenever the charging reaches the desired level. Otherwise, the system notifies Steve when the charging process is finished. After that, selecting “Pay Now”, he can pay for the given service using the credit card registered on the system or using the contactless card he prefers directly in the proper payment area present in each slot. Then Steve can continue his journey happily.

6. CPO station administrator monitoring station status

A CPO employee, with station administration tasks, logs into the system using his administrator credentials. In the main page he has access to the stations of his company connected to the system backend infrastructure. By selecting one of them, he can monitor the status, such as electricity and free slots available, current price and if the station is properly working. He can also analyse the performance of the last 60 days.

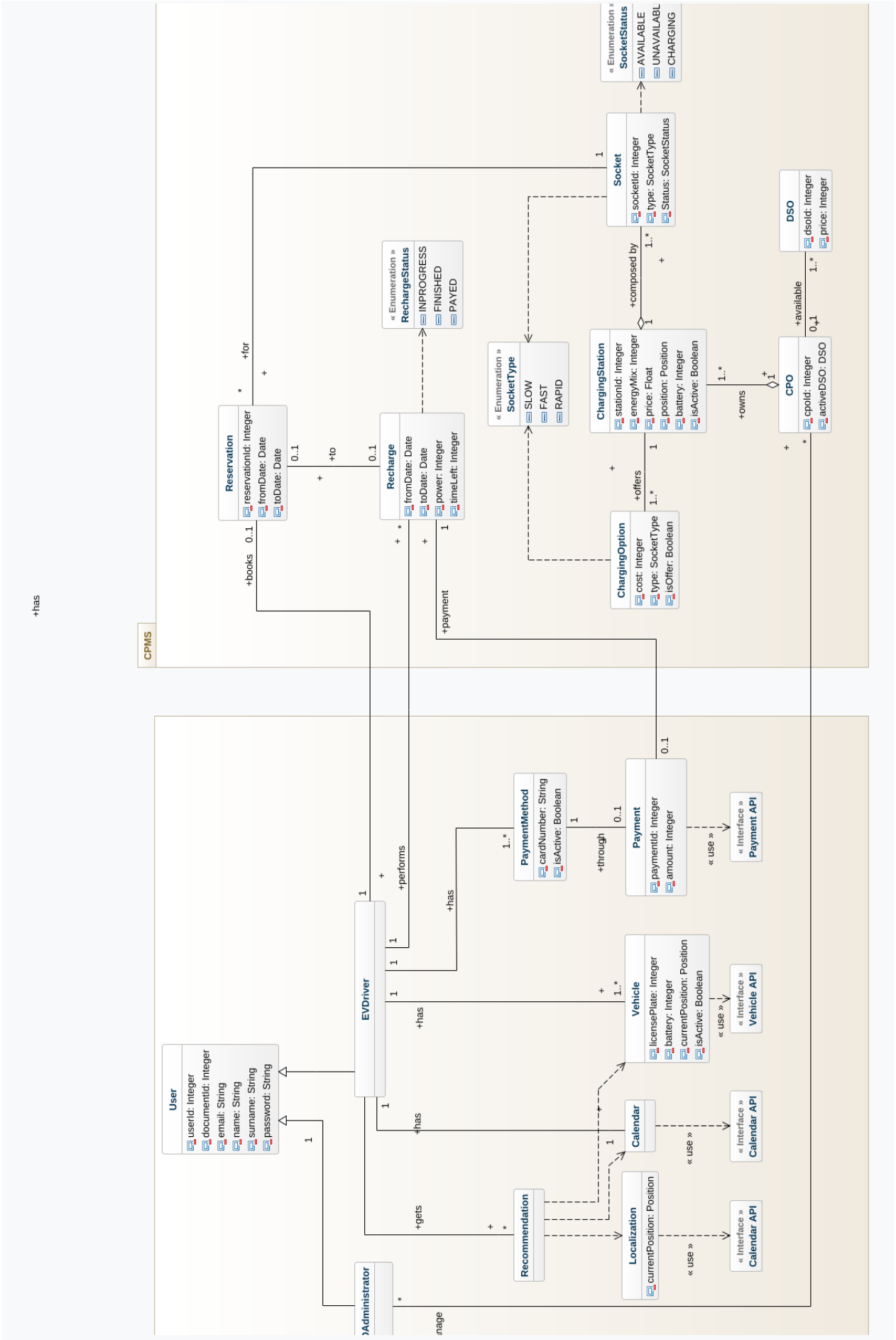
7. **CPO station administrator taking decisions**

The station administrator Mike logs into the system using his administrator credentials. After he has selected a charging station, Mike has at one's disposal its performance and can set a DSO to acquire the energy from, analysing the offered price, choose how the station must charge the cars, by using current ground energy or the one present in the stationary battery. The administrator can also choose to set these decisions to be automatized by the CPMS.

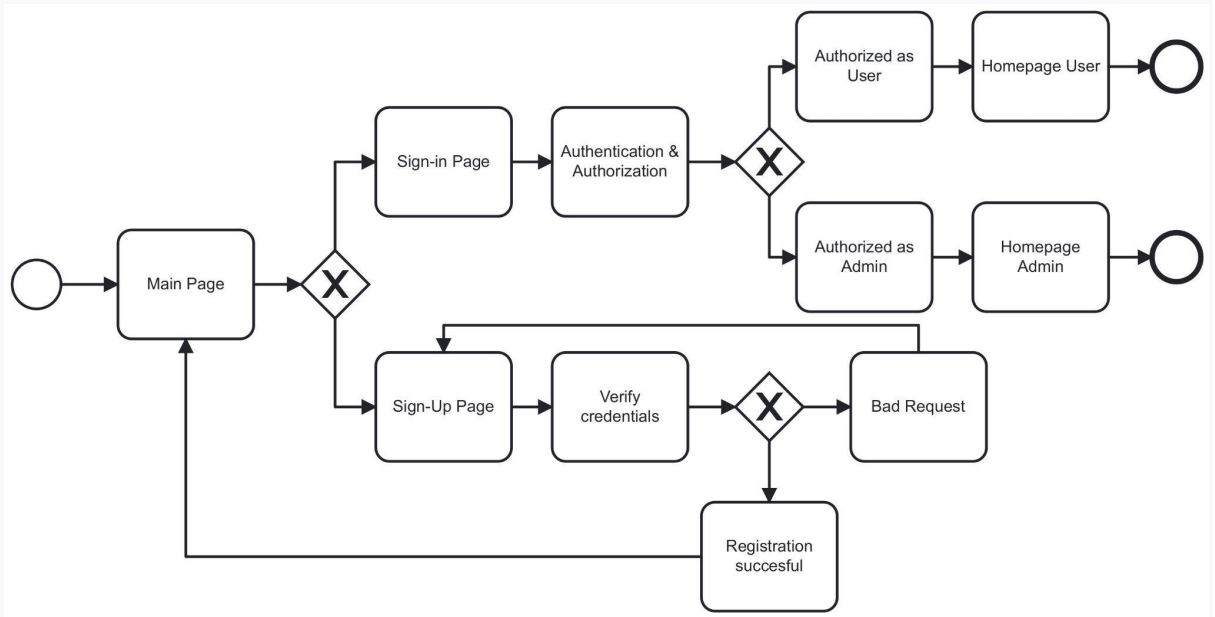
2.1.2 Class Diagram

This is the class diagram of the system. User is an abstract class, associated with its extensions: EV Driver and CPO Administrator, where each one has its own specific attributes. Following the diagram in the EV Driver direction there are Electric Vehicle (notice that a user can have more than one vehicle and a vehicle can be associated with more than one user), Reservation, where the timeframe and the related socket are specified. In the CPO Administrator direction there are CPO and DSO, with their identifiers. Finally, there are Charging Station class, linked to its CPO owner, and Socket classes composing the station.

Class Diagram

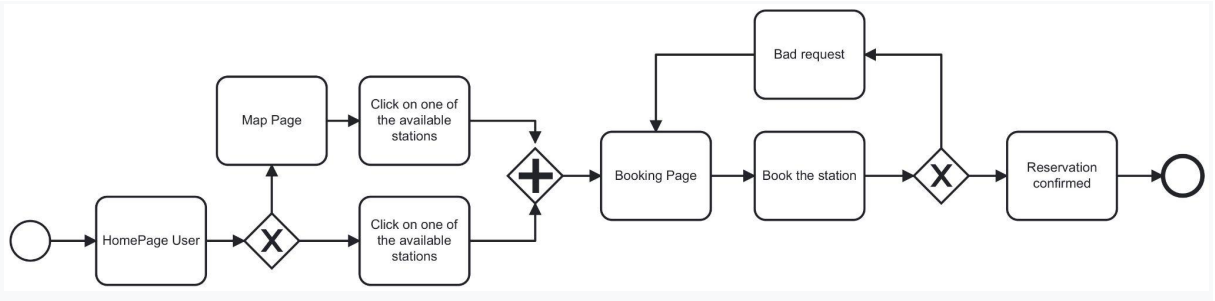


2.1.3 StateChart



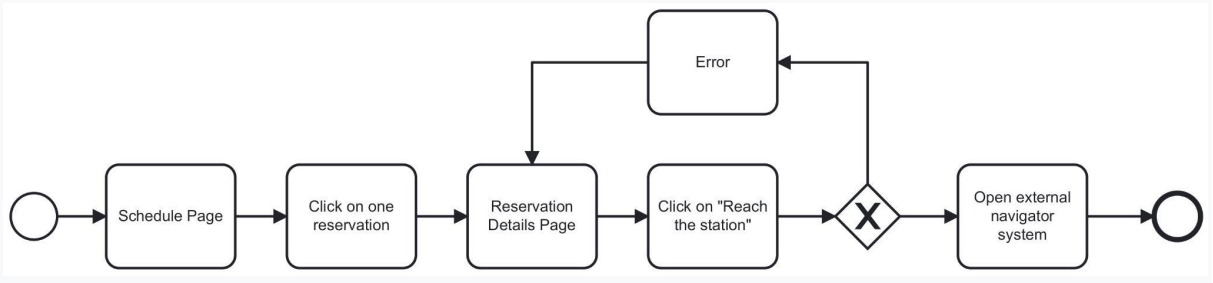
1-Sign-in and Sign-Up

- The state diagram summarises the sign-in and sign-up actions. Once the Driver enters the main Page, he can decide if he wants to register or to login, while the Administrator can only login. If the user chooses the login, depending on his authorizations, he is redirected to the Driver or Administrator main page.



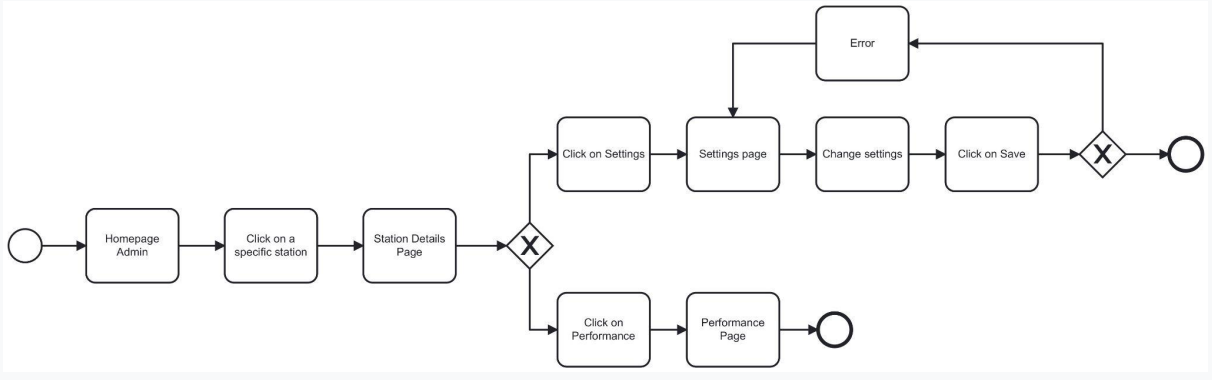
2- Book the station

- The user can decide which station he wants to book from two different pages. The Home Page shows a list of the available stations, while the Map Page displays all the available stations on the map.



3-Reach the station

The Driver can visualise the recommended



4- Admin access to the system

There are described all the possible actions that the Administrator can do. The Admin section has two main pages, the Performance page where, clicking on a specific station there are displayed the main statistics and information (retrieved from the CPMS and processed by the eMSP), and the Settings page, where the Admin can make decisions about each station(energy mix, DSO…).

2.2 Product functions

In this section the main functionalities of eMALL is described in more detail:

2.2.1 Let drivers know the charging stations nearby, their cost, special offers available and book a charge

One of the most important aspects of the system is the ability of the drivers to know the location of nearby charging stations: this will be achieved throughout the visualisation of an interactive map shown in the “Map Page”, that thanks to the drivers gps, will show the current user location as well as the real time data regarding the charging stations nearby.

In the “Main Page” a list of all available charging stations is shown. The user can sort the charging stations by applying filters to the list, such as cost, travel distance and presence of special offers available.

The user will be able to select a charging station to obtain more details regarding it and if it is available, he will be able to book a charge for a certain time frame, depending on the ones available.

After booking a charge, the user will have a certain period of time to show up at the charging station and start the charging process, otherwise the reservation previously booked will be cancelled.

In case a special offer is present, the user will be notified through the presence of a special icon on the charging stations promoting the offer. Furthermore, when selecting a charging station, the offer details will be shown.

2.2.2 Let drivers start the charging process, be notified when it ends and pay for the service

After a driver reaches the station and parks his EV in the designated area, he can connect it to the charging station’s socket he previously booked or to one available at the moment. To start the charging process the driver must scan a QR code present on the charging station or insert a code on the app.

After checking that the user has a valid payment method linked to his account and the charging station is available and not booked by other users, the charging process starts. The user will be able to monitor the current status of the battery and the estimated time left for the charging to complete. The user will be free to stop the charging process anytime through the app. After the charging process ends the payment will be automatically debited from the user’s account. In case the payment fails, the user will be unable to use the service until a valid payment method is selected and the transaction is completed.

When the vehicle battery is completely charged, the user will be notified through a notification on his device and will have a certain period of time to disconnect his EV and leave the charging spot free.

2.3 User characteristics

The following three actors are considered in the eMALL systems.

1. **Unregistered electric vehicle driver**

   A driver that needs to register to the eMALL platform before being able to use any of its functionalities.

2. **Electric vehicle driver**

   A registered user that uses the system to find the charging stations nearby, their cost and any special offer they have. He can also book a charge for a certain timeframe and monitor the live charging process.

3. **CPO station administrator**

   A registered user, working as station administrator for a specific CPO. He is able to monitor the status of his charging stations and to take strategic decisions in the supply chain of the stations.

2.3 Assumptions, dependencies and constraints

2.3.1 Domain assumptions

Identifier	Description
D1	There exists an API where user credentials can be verified (licence plate,email..)
D2	There exists an API where the correct map and driver device gps data can be retrieved

D3	There exists an API where updated DSO prices can be retrieved and the DSO can be selected for energy acquirement
D4	When a driver disconnects his EV from the socket he utilised, he immediately leaves the parking area.
D5	Users insert their personally identifiable informations into the system
D6	CPO Admins have access to an already existing account on eMALL
D7	Users give the system the authorization to access personal data regarding their EV, current position and calendar
D8	Energy is supplied correctly from DSOs to CPOs charging stations
D9	Drivers correctly park and connect their vehicle to the charging station socket
D10	The drivers respect the reservations made by showing up at the charging station booked on time.
D11	All the vehicles considered have a Universal Socket accepted by all the sockets
D12	For each socket is associated a parking spot where the vehicle can be parked while it's charging.
D13	The Drivers own the vehicles they use
D14	The sockets charging a vehicle can retrieve in real time correct data regarding the current charging speed and vehicle battery status

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The user interface of eMALL is both a computer and a mobile application that will be used both by EV Drivers and CPO Administrators. It should be available as much as possible and easy to use, in particular the mobile app interface, where Drivers search and make reservations for charging slots, has to be fast and user-friendly. On the other side, Administrators are supposed to have office PCs where to work on the system, so the computer application has to be optimised and oriented towards data analysis through the use of specific accurate tools.

3.1.2 Hardware Interfaces

The system sets up from the necessity to handle multiple and simultaneous commands from different actors, which are driven in their decisions by the continuous upgrade of specific information from communication sources. Moreover, since the system has to be fully available both on the mobile and the computer application through the connection to , the only hardware interface requirements are a web browser (or even better a mobile application store) and the possibility to provide geolocalization information by the user for a better experience. The system also relies on the use of different sensors deployed to obtain data, such as vehicle battery percentage or the internal status of the charging stations. This data are supposed to be managed externally and that there are specific APIs to retrieve data for eMALL.

3.1.3 Communication Interfaces

The system is based on the management of many information sources to provide lots of its functionalities. Therefore, it is required to communicate with external information providers from where eMALL retrieves data or services. In particular, there are different interfaces that eMALL exploits, possibly through Web APIs, as previously explained in the assumptions.

- **Retrieval of data on EV battery status**
  The interface is able to respond with the current status of the battery, if there are malfunctions or not, the battery level both during the charging process and during its usage (in line with the goal of customization and the recommendations the system gives to the user).

- **Retrieval of data on charging station internal status**
  The interface is able to respond with the current internal status of the charging station, which means the mix of energy used (specifying the percentage from each source), the level of the storage of the stationary battery
  and the availability of each slot with its main information, such as the power provided (super fast, fast, slow).

- **Retrieval geolocalization data of EV Driver**
  The interface is able to provide in real time the current position (latitude and longitude) of the user device in the world map

- **Retrieval of current world map**
  The interface is able to provide data regarding the map through which the system shows to the user its current location, the location of charging stations and a possible path to reach them.

- **Retrieval of data on EV driver schedule**
  The interface provides the information about the user's daily schedule, contained in his own calendar. Using this information, the system is able to give proper suggestions, about the best time in which to charge the EV, to the user.

- **Payment authorization and correct completion**
  The interface deals with the payment process, in particular with the validation of the credit/debit card, the
  authorization to proceed with the payment (given by the user through his online bank o similar methods) and the
  effective correctness of the transaction.

Moreover, there are communication protocols needed for the correct forwarding of information among the different subsystems and the physical infrastructures (charging station and its components).

- **OCPI**
  This protocol is required in the communication between our subsystems: eMSP and CPMS. It is mainly used to provide charging station information (socket status and details, location and tariff), to book and to authorise the charging session.
- **OCPP**
  This protocol is required in the communication between CPMS and a specific charging station. The main information forwarded are the principal operative commands of the charging session (start/stop and current status) and diagnostic information/updates.

These protocols are supposed to work properly. Furthermore, the OCPP informations, differently from the OCPI ones, are no longer described and modelled in this document.

3.2 Functional Requirements

Requirement	Description
R1	The system shall allow an unregistered user to register an account
R2	The system shall allow a registered Driever to insert one or more EV and to set one of them as active
R3	The system shall allow a registered Driver to insert and modify (even if banned for an unsuccessful payment transaction) one or more valid payment methods
R4	The system shall allow a registered Driver to visualise the available charging stations in the list view (sorted by the filter he selected) or in the map view (together with his current position)

R5	The system shall allow a registered Driver to view the available charging stations (sorted by the filter he selected), visualising its CPO owner, the current energy price, the distance from the user location and the estimated time to reach it
R6	The system shall allow a registered Driver to get information (CPO owner, current energy price, distance from the user location and estimated time to reach it) of a station clicking on it in the map view
R7	The system shall allow a registered user to book a socket at a certain timeframe
R8	The system shall allow a registered user to choose the desired type of charging socket (slow, fast, rapid)
R9	The system shall allow a registered user to view his reservations
R10	The system shall allow a registered user to get the path to reach a station starting from his reservation view or from the station view (list and map)
R11	The system shall allow a registered Driver to start the charging process and to stop it
R12	The system shall allow a registered Driver to view the live status of the charging process and the remaining estimated time
R13	The system shall notify the registered Driver when the charging process is finished

R14	The system shall allow a registered Driver to view system's recommendations
R15	The system shall allow a registered Driver to accept one of the recommendations
R16	The system shall allow a registered Driver to cancel a reservation
R17	The system shall allow a registered Administrator to view the status of his CPO charging stations, including availability, DSO and energy price (for the CPO)
R18	The system shall allow a registered Administrator to view status, type and availability of each socket in a certain charging station
R19	The system shall allow a registered Administrator to view the status of the stationary battery (if present) of a certain charging station
R20	The system shall allow a registered Administrator to view the current energy mix (ground, stationary battery, green) used in the charging process in a certain station
R21	The system shall allow a registered Administrator to enable the CPMS to automatically select active DSO and energy mix, energy price (for the costumer) at a certain charging station
R22	The system shall allow a registered Administrator to manually select active DSO and energy mix, energy price (for the costumer) at a certain charging station

R23	The system shall allow a registered Administrator to view statistics for a certain charging station, including average customers usage and in which time slots, average energy price (both for CPO and user), average energy mix used
R24	The system must be able to notify user of exception
R25	The system must be able to notify user on successful action
R26	The system must store the history of charging stations performance
R27	The system must allow registered Driver to login
R28	The system must allow registered Administrator to login

3.2.1 Mapping on Goals

Goal	**Domain assumptions**	**Requirements**
G1	D1, D2, D3, D5, D13	R4, R5, R6, R8, R24 R25, R29
G2	D1, D2, D4, D10, D5, D13	R4, R5, R8, R9, R16, R24, R23, R24, R27
G3	D7, D8, D9, D11, D12, D13, D14, D15	R7, R9, R10, R22, R23, R24, R27
G4	D3, D6	R17, R18, R19, R20, R21, R22, R25, R26, R28
G5	D3, D6	R17, R18, R19, R20, R21, R22, R23, R24, R25, R26, R28
G6	D3, D6	R17, R21, R22, R23, R24, R25, R26, R28

G7	D2, D3, D5, D7, D13	R2, R14, R15, R16, R24, R25, R27

3.2.2 Use cases

1-Driver registration:

Actor	Driver
Entry condition	The Driver does not have an account and he is on the MainPage of the system
Events flow	1. The Driver clicks the “Sign-Up” button 2. The Driver enters name, surname, a valid payment method, email address and password. 3. The Driver enters the licence plate of the first vehicle he wants to add. 4. The Driver accepts the system to use the geolocalization service of his device and his daily schedule information. 5. e-Mall processes the information and shows a successful message
Exit condition	A new Driver account is created
Exceptions	Exceptions 1. The Driver does not enter all the mandatory data. Exceptions 2. The Driver does not enter a valid payment method. Exceptions 3. The Driver does not permit the system to access the localization service. Exceptions 4. The Driver enters a non-existing licence plate. • In all cases eMall notifies the Driver displaying an error message

2- User login to e-MALL

Actor	Driver or Administrator
Entry condition	The Driver or the Administrator is on the MainPage of the system

Events flow	1. The Driver clicks the “Sign-In” button. 2. The Driver enters his email and password 3. The Driver clicks on the submit button. 4. e-Mall processes the information and redirects to the User Homepage if it is a Driver or to the Admin Homepage if it is an Administrator
Exit condition	Driver or Administrator is logged in
Exceptions	Exceptions 1. The Driver does not enter a correct password for that email Exceptions 2. The email does not exist in the system. • In all cases eMall notifies the Driver showing an error message

3- Driver add new vehicle

Actor	Driver
Entry condition	The Driver is logged into the system and he is in the Stations Page
Event flow	1. The Driver clicks on UserProfile Page 2. The Driver clicks on “Enter new vehicle” 3. The Driver enters the licence plate of his vehicle. 4. e-Mall processes the information and shows a success message.
Exit condition	A new vehicle is inserted in the “My vehicles” section in the User Profile.
Exceptions	Exceptions 1. The Driver enters a licence plate which does not exist or already exists in the database. • In all cases eMall will notify the Driver through showing an error message

4- Driver Change Active Vehicle

Actor	Driver
Entry condition	The Driver is logged in to the system and he is in the

	Stations Page.
Event flow	1.The Driver clicks on UserProfile Page 2. The Driver clicks on “Change Active Vehicle” 3. The Driver chooses one vehicle from a list showing all his vehicles. 4. eMall processes the information and shows a success message.
Exit condition	The vehicle chosen is set as the Active Vehicle
Exceptions	No Exception

5- Driver add new payment method

Actor	Driver
Entry condition	The Driver is in the station he booked and he is logged in on the Stations Page.
Event flow	1.The Driver clicks on UserProfile Page 2. The Driver clicks on “Add new payment method” 3. The Driver fills the forms with all the mandatory data. 4. eMall processes the information and shows a success message
Exit condition	The Driver has a new payment method among his personal payment methods.
Exceptions	Exceptions 1. The Driver enters wrong data in the form. • In this case eMall will notify the driver via an error message

6- Driver Change Active Payment Method

Actor	Driver
Entry condition	The Driver is in the station he booked and he is logged in on the Stations Page.
Event flow	1.The Driver clicks on UserProfile Page 2. The Driver clicks on “Change Active Payment Method”

	3. The driver selects from a list with all his previously entered payment methods the payment method he wants to set as active. 4. eMall processes the information and shows a success message
Exit condition	The Driver's Active Payment method is modified.
Exceptions	• No Exceptions.

7- Driver Make a reservation

Actor	Driver
Entry condition	The Driver is logged in to the system and he is in the Stations Page.
Event flow	1. The Driver can decide if to click on Map Page and select the station from the map shown or select directly from the User HomePage where the is shown a list of the stations. 2. The Driver clicks on the station where he wants to book a socket for his vehicle. 3. The Driver clicks on "CreateReservation". 3. The Driver selects which socket he wants (ex: superfast, fast..) if more than one is available. 4.The Driver selects which TimeFrame he wants. 5. eMall processes the information and shows a success message
Exit condition	A new event in the User Calendar and in the Driver's Page"MyReservations" is created displaying the vehicle, time chosen for the reservation and the socket ID assigned.
Exceptions	Exceptions 1.The Driver clicks a station where there are no sockets accepted by the Driver's Active Vehicle. • In this case eMall will notify the driver via an error message

8- Driver Cancel a reservation

Actor	Driver
Entry condition	The Driver is logged in to the system and he is in the Stations Page with at least an active reservation.
Event flow	1. The Driver goes to MyReservations Page and a list of all of his current bookings is displayed. 2. The driver clicks on the reservation he wants to cancel. 3. The Driver clicks on “Cancel” 4. eMall processes the information and shows a success message
Exit condition	The reservation is cancelled.
Exceptions	Exceptions 1. The time remaining for the reservation is less than 15 minutes • In all cases eMall will notify the Driver through showing an error message

9- Driver Start the charging process

Actor	Driver
Entry condition	The Driver is in the station he booked, he is logged in the Station Page with at least an active reservation.
Event flow	1. The Driver goes to MyReservations Page 2. The Driver clicks on the reservation. 3.The Driver clicks on “Start recharging” 4. eMall processes the information and shows a success message.
Exit condition	The recharging process starts.
Exceptions	Exceptions 1. The socket is not plugged in correctly. Exceptions . There is not enough energy in the station. • In all cases eMall will notify the driver via an error message

10 -Driver View charging status

Actor	Driver
Entry condition	The Driver is logged in and he is currently in the Station Page.
Event flow	1. The Driver goes to My Reservations Page. 2. The Driver clicks on one of his reservations. 3. The Driver clicks on “ChargingStatus” 4. The Driver has also the option to stop the charging process. 5. eMall processes the information and shows a success message.
Exit condition	The Driver visualises the charging status.
Exceptions	Exceptions 1. The reservation has not started yet • In this case eMall will notify the driver via an error message

11 - Driver reach a Station

Actor	Driver
Entry condition	The Driver is logged in and he is currently in the Station Page.
Event flow	1.The Driver has 3 alternatives: either he clicks on the My Reservations Page and clicks on the reservation, either he clicks on Map Page and clicks on the station, or he clicks on the Stations Page and clicks on the station. 2. The Driver clicks on “Go To Station” 3. e-MALL processes the information and shows the directions.
Exit condition	Driving directions are displayed on the Maps App on the Driver's device
Exceptions	No Exceptions

12- Administrator Modify the energy mix

Actor	Administrator
Entry condition	The Administrator is logged in and he is currently in the Stations Page.
Event flow	1. The Administrator clicks on the desired station 2. The Administrator can click either on the EnergyMix desired or set the automatic change mode. 3.eMall processes the information and shows a success message.
Exit condition	The CPO mix of supply sources is changed.
Exceptions	Exceptions 1. There is not enough power to change the mix of supply sources , this could be due to lack of energy provided by DSO or the energy stored in the batteries is not enough. • eMall will notify the Administrator through showing the reason why it failed

13 - Administrator modify DSO

Actor	Administrator
Entry condition	The Administrator is logged in and he is currently in the Stations Page
Event flow	1. The Administrator clicks on the desired station 2. The Administrator can click either on the DSO desired or set the automatic change mode. 3.eMall processes the information and shows a success message..
Exit condition	The DSO is changed.
Exceptions	No Exception

14 - Administrator set an offer

Actor	Administrator

Entry condition	The Administrator is logged in and he is currently in Stations Page
Event flow	1. The Administrator clicks on the station where he wants to set an offer. 2. The Administrator clicks on “Set Offer”. 2. eMall processes the information and shows a success message
Exit condition	The station has a new offer on its sockets.
Exceptions	No Exception

15 - Administrator view station performance

Actor	Administrator
Entry condition	The Administrator is logged in and he is currently in Stations Page
Event flow	1. The Administrator clicks on the station 2. eMall processes the information and shows the data about that station, such as energy stored in the battery, number of reservations…
Exit condition	The station performance and statistics are shown.
Exceptions	No Exception

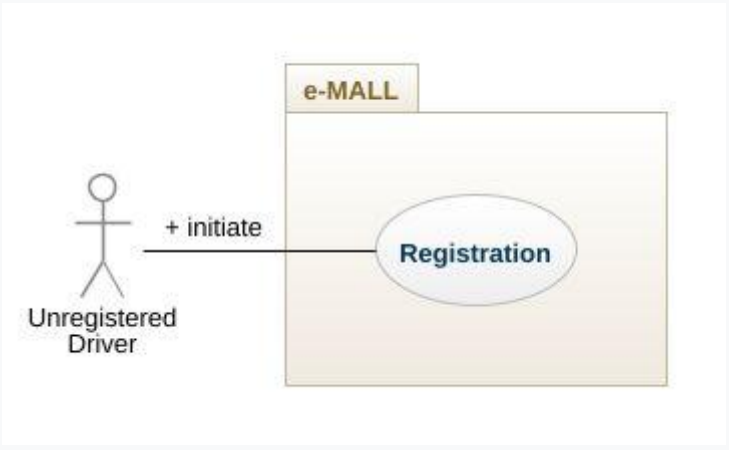
16 -Driver Accept a recommendation

Actor	Driver
Entry condition	The Driver is logged in and he is currently in the HomePage Driver
Event flow	1. The Driver clicks on “My Recommendations”. 2. The Driver clicks on one of the two recommendations shown. 3. The Driver clicks on “Add to My Reservations” 5. E-Mall processes the information and shows a success message.
Exit	A new Driver’s reservation is created and added to

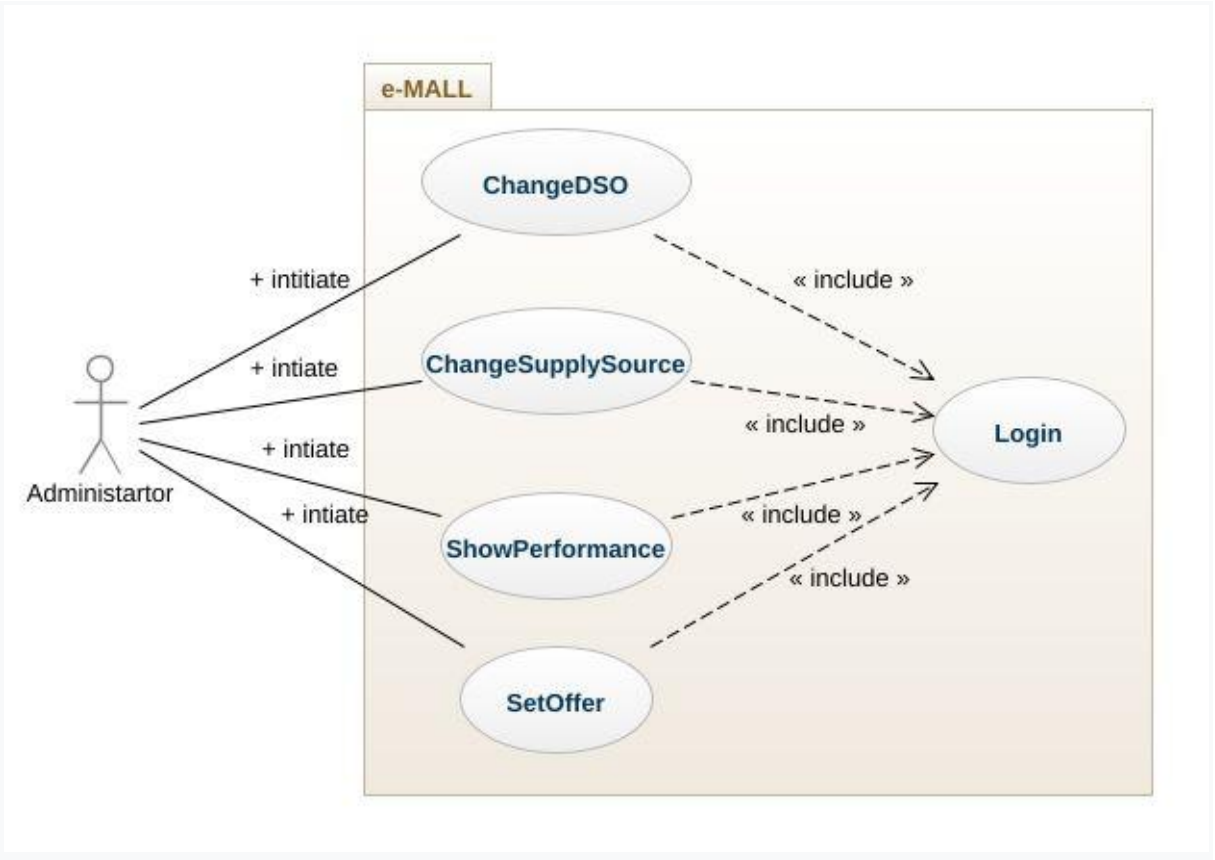
condition	the Driver’s calendar.
Exceptions	No Exception

3.2.3 Use case diagrams

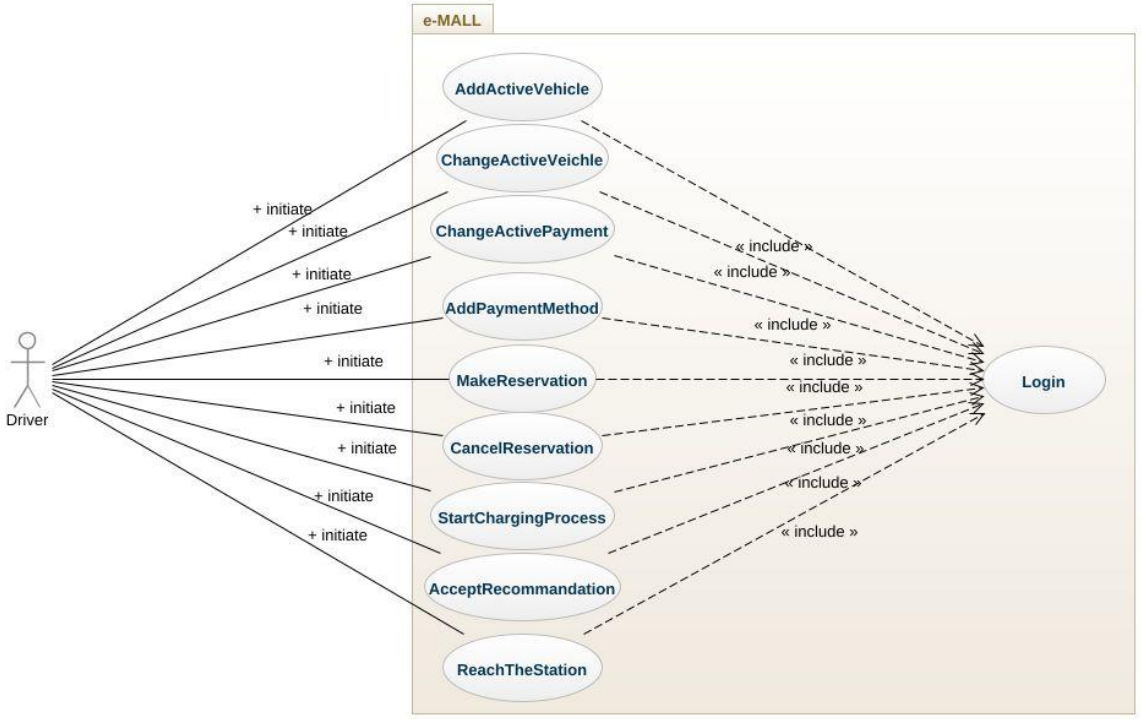
1. Unregistered Driver



2. Administrator

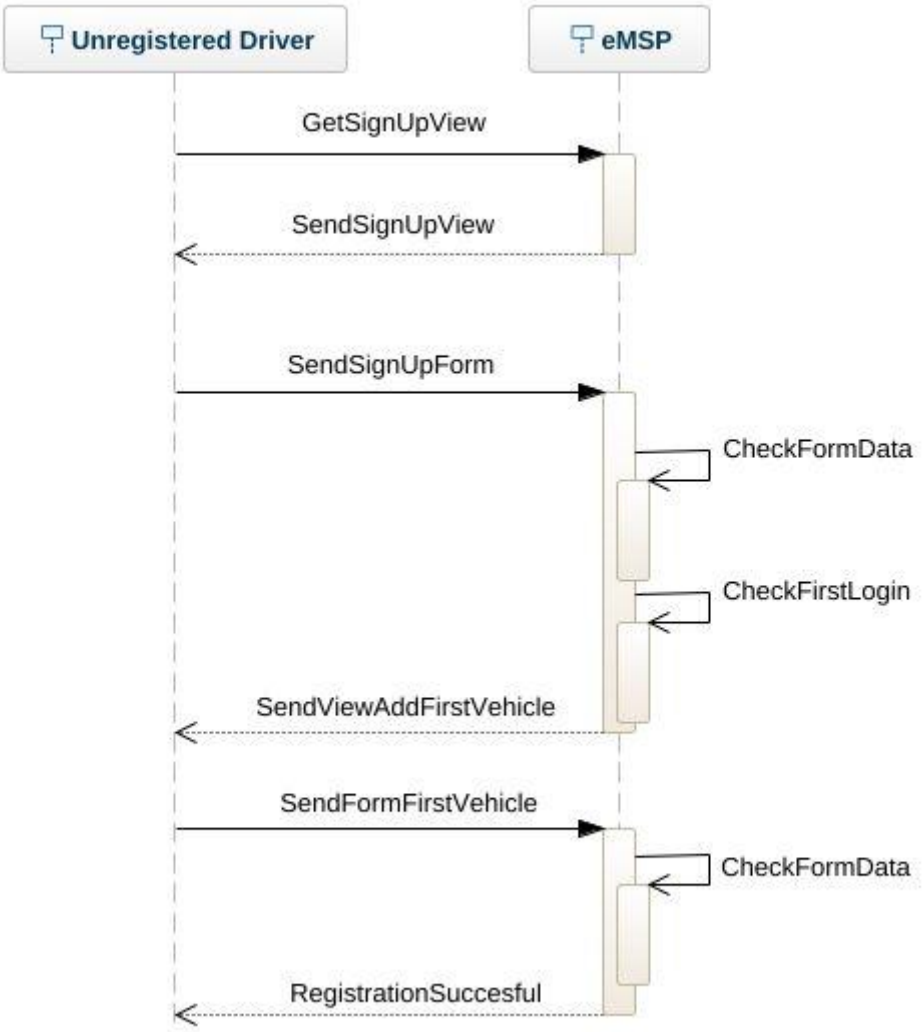


3.Driver



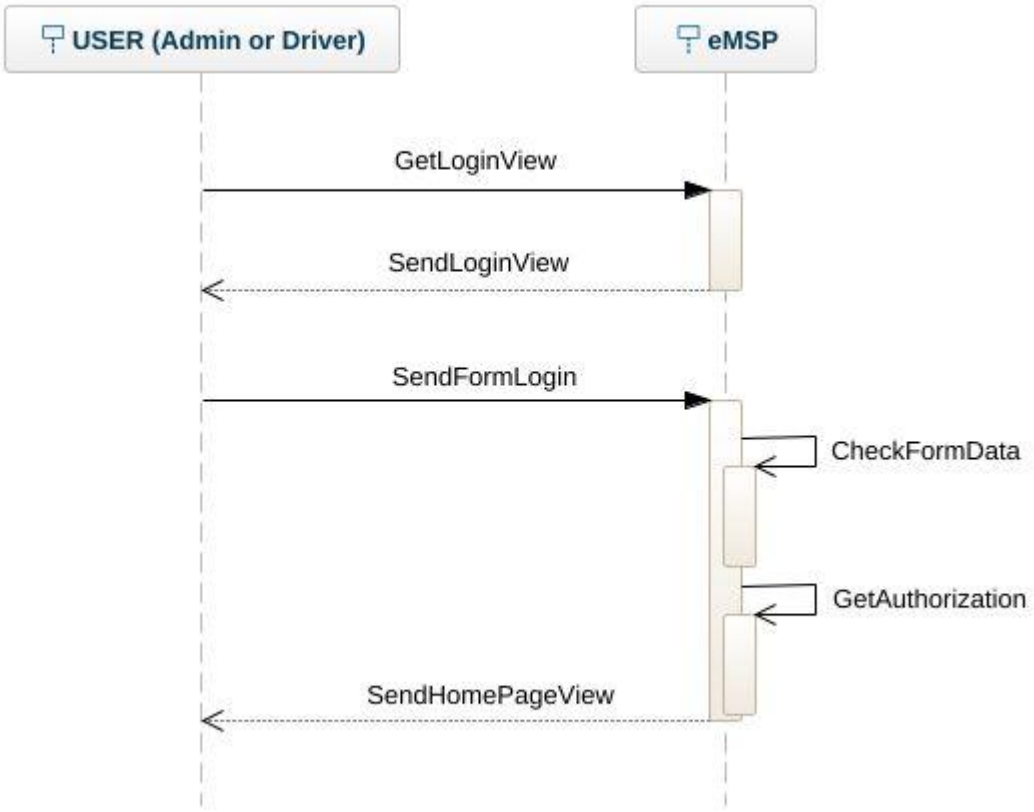
Sequence diagram

1. **Driver registration**

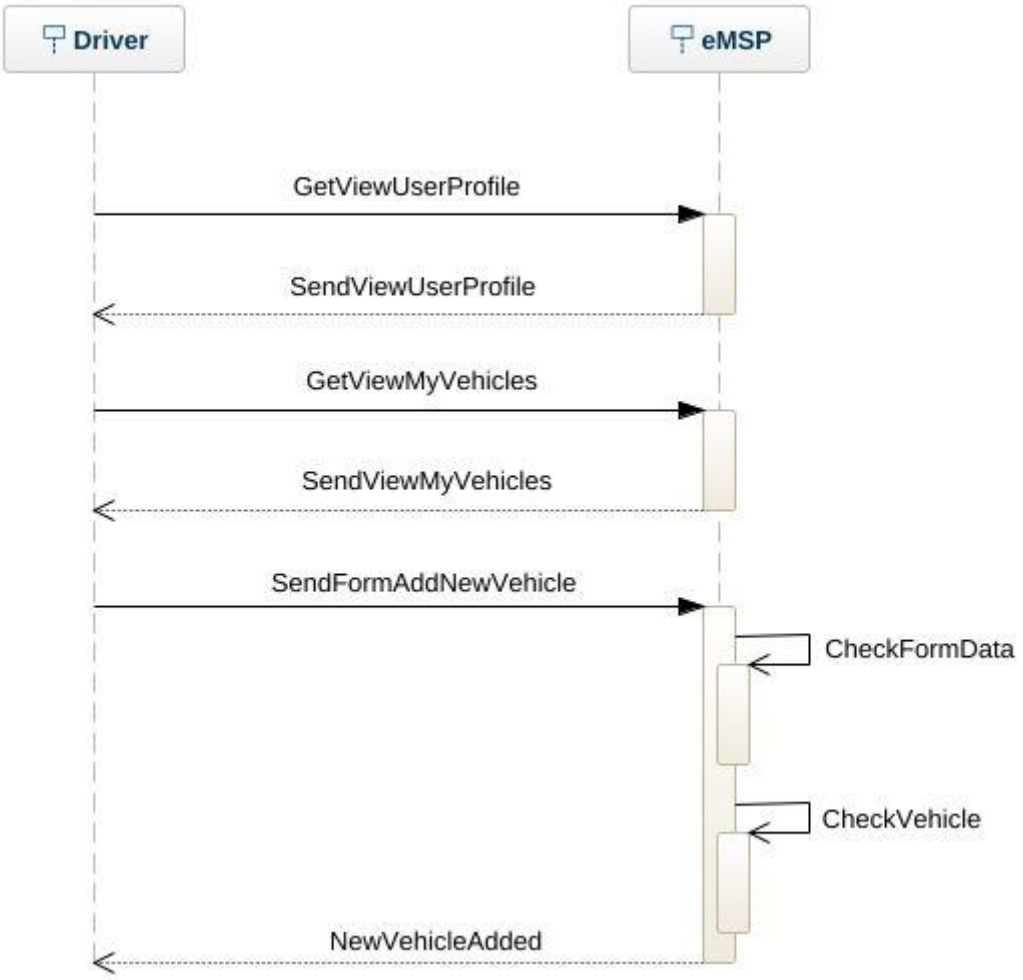


After the Driver enters the Profile Information he will be asked to enter his vehicle information which will be set as his active vehicle. In CheckFirstLogin is checked if the email has been already registered.

2.User login to e-MALL

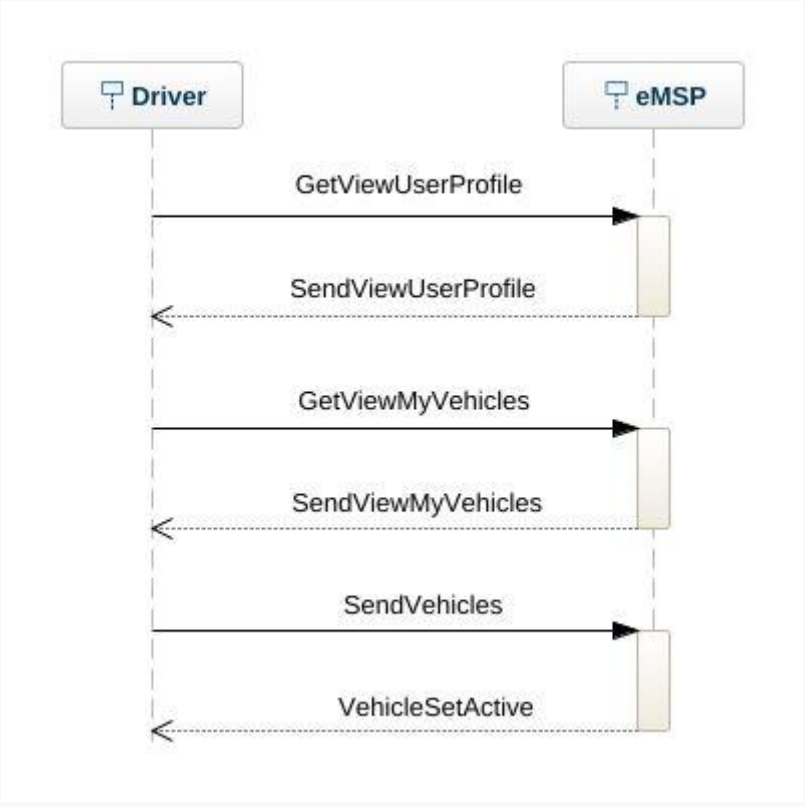


3.Driver Add a new vehicle



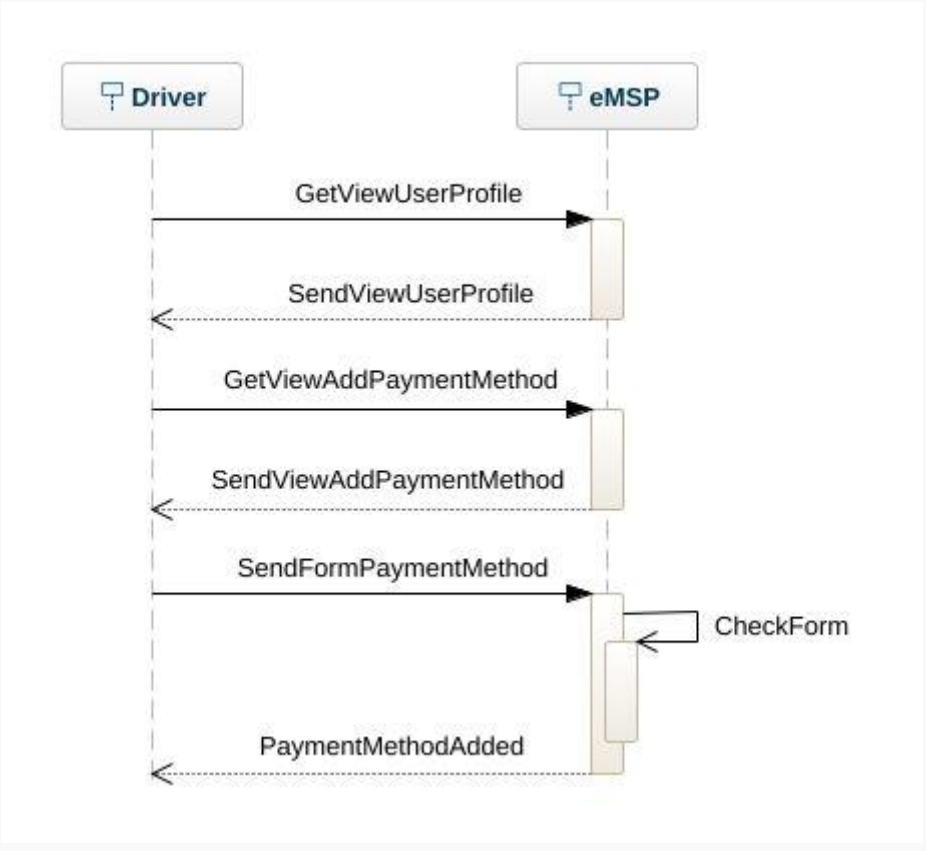
In CheckFormData is checked if the licence plate entered is correct.
In CheckVeichle is checked if the vehicle is already present in the eMall Database.

4.Driver Change Active Vehicle



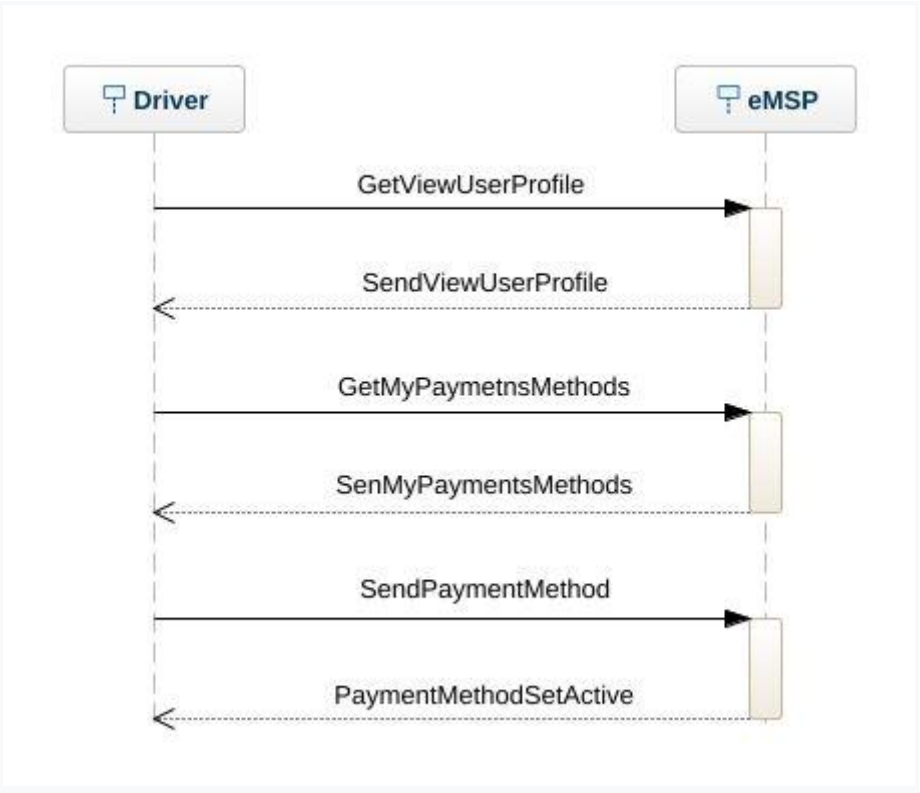
In the section MyVehicles in the UserProfile Page is present a button clickable that permits the Driver to change his Active Vehicle. The Active Vehicle will be useful for the system to make custom recommendations.

5. Driver Add new payment method



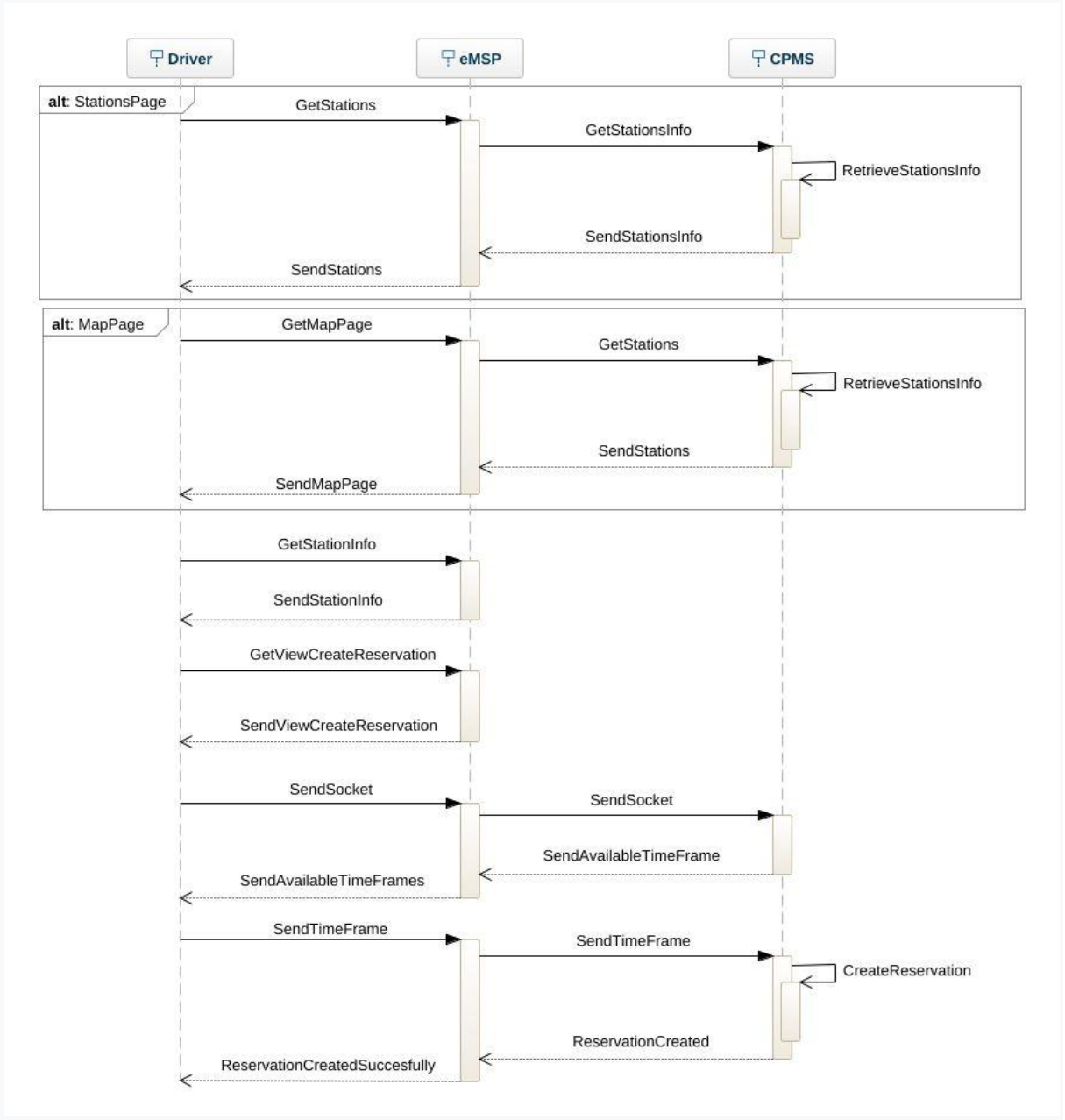
The Driver goes on the UserProfile Page and clicks on “AddPaymentMethod”. Then the Driver has to fill the form with all the mandatory data about his payment method and send the form. The eMSP with CheckForm checks if all the data inserted is correct and notifies the Driver of the successful addition of a new payment method.

6. Driver Change Active Payment Method



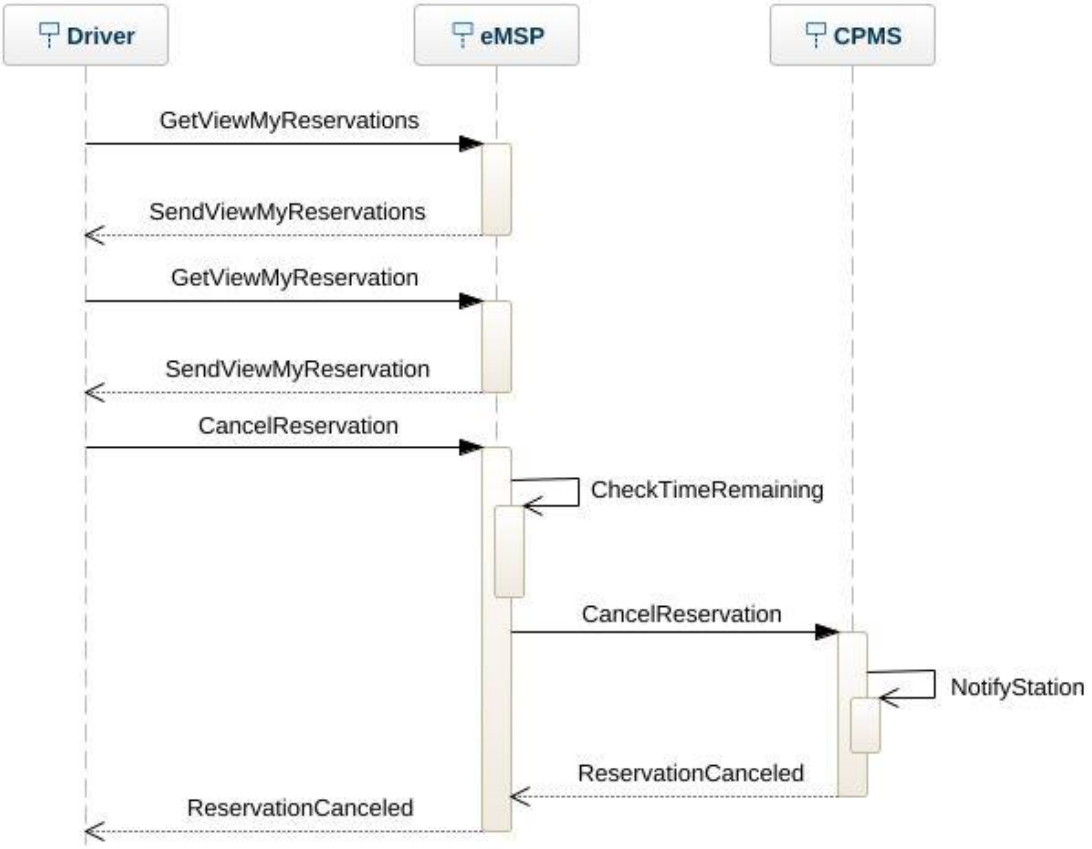
Here is described the process that the Driver must follow to change his active payment method. The Driver goes to the UserProfile Page and clicks on “ MyPaymentMethods”. A list of Driver’s payment methods is shown, afterwards the Driver chooses the payment method he wants and the eMSP will proceed to change the Driver’s ActivePaymentMethod.

7.Driver Make a reservation



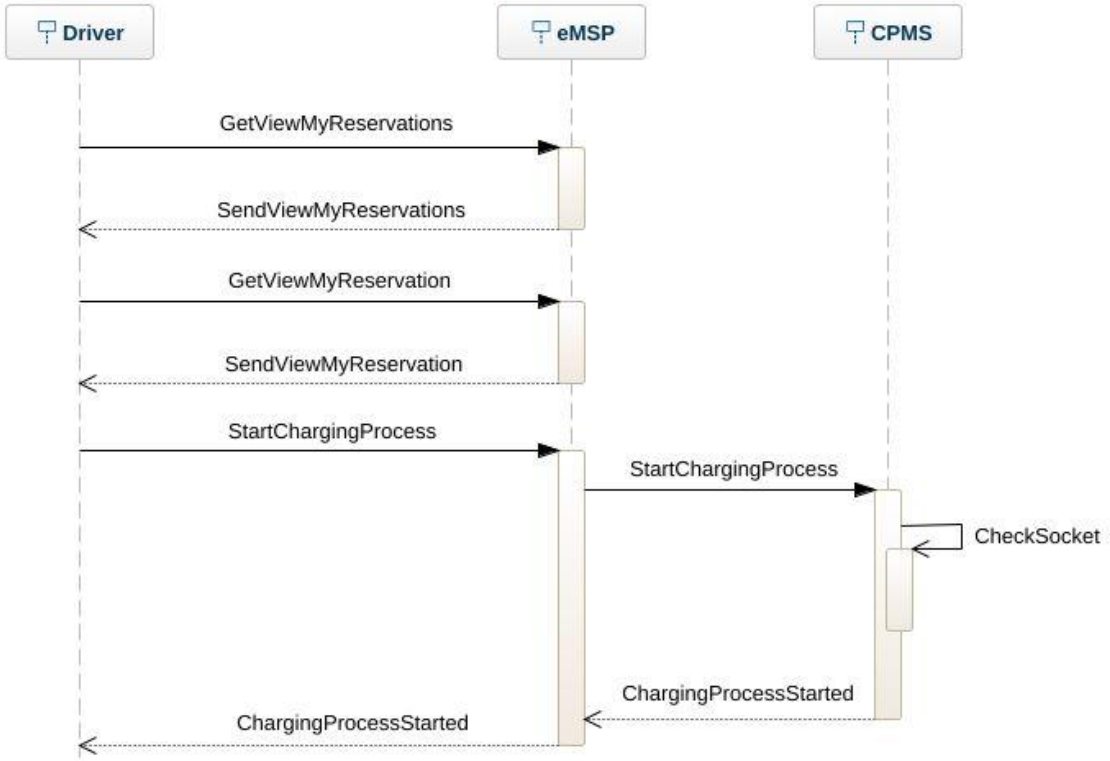
The two alternative sequences represent the possibility for the Driver to choose the desired station from the Map Page or from the Stations Page.The CPMS with RetrieveStationInfo retrieves all the information about the stations it manages.
Then the Driver clicks on “Create Reservation” on the desired station and decides which kind of socket he wants to book. Afterwards the system will show the time frames available for that kind of socket in that station. Eventually the Reservation is created.

8.Driver Cancel a reservation



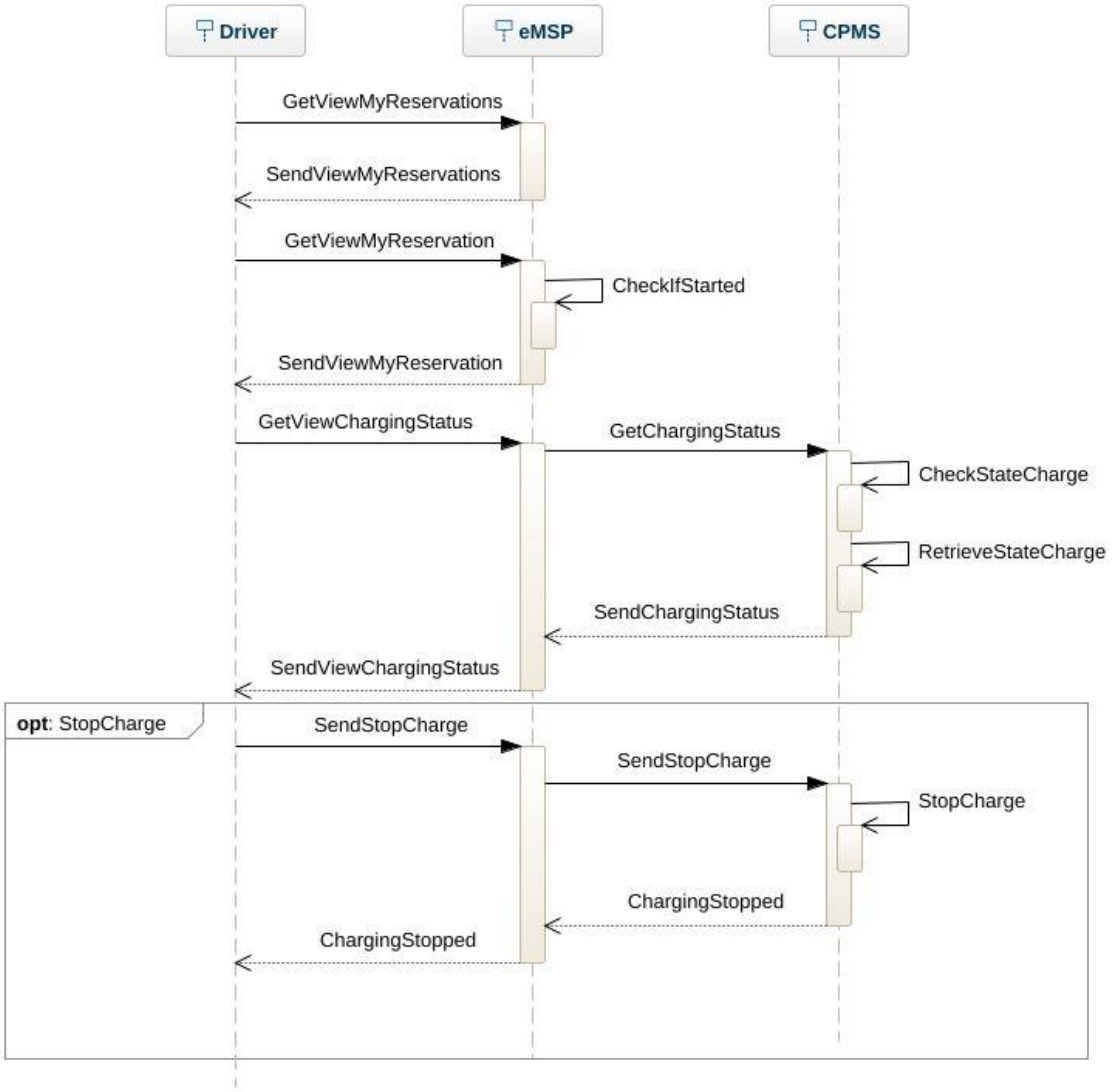
The Driver goes to the UserProfile Page and clicks on “MyReservations". A list of all the Driver's reservations is shown, afterwards the Driver clicks on the reservation he wants to cancel. With CheckTimeRemaining the eMSP checks if the reservation will start in less than fifteen minutes.

9. Driver Start the charging process:



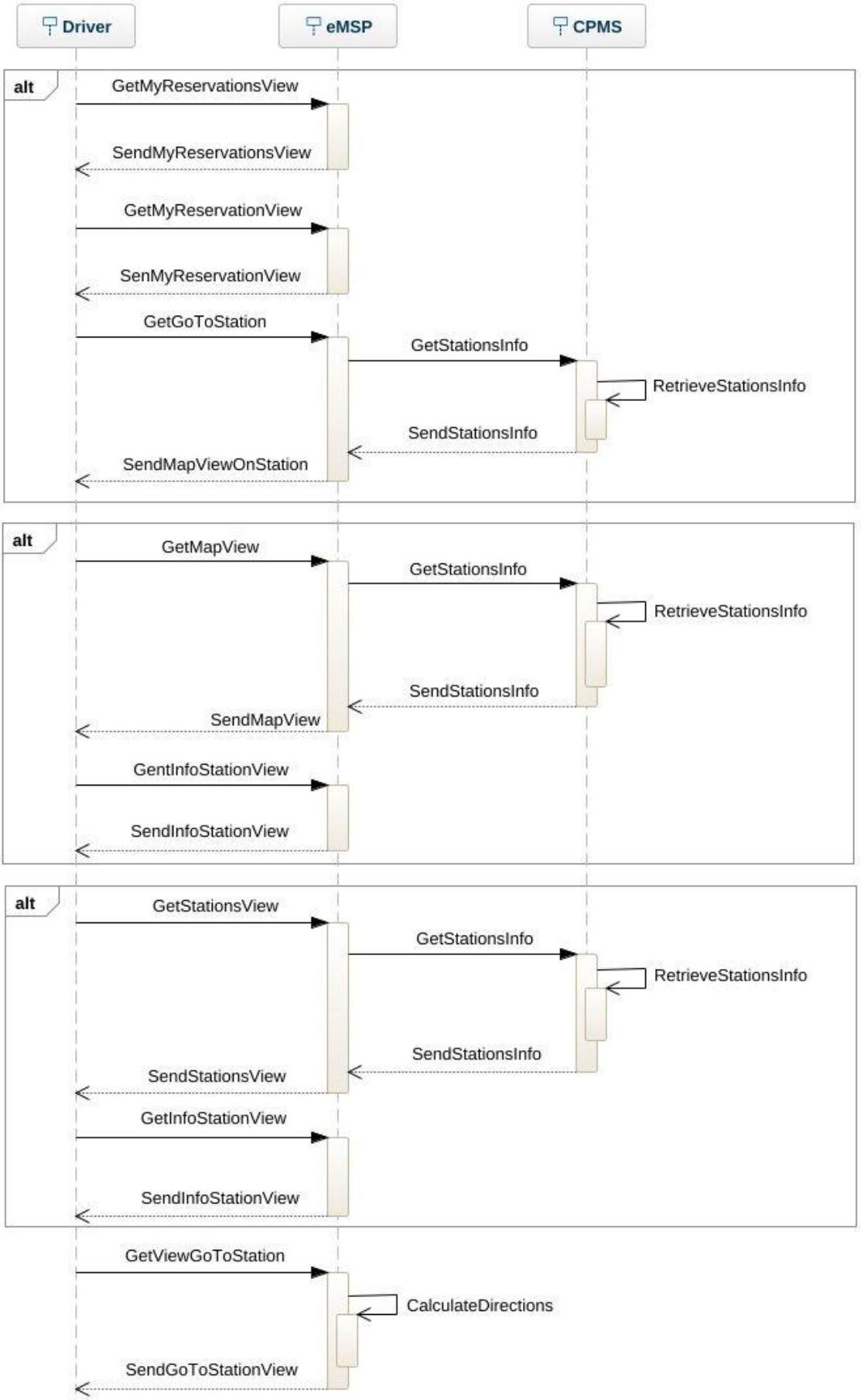
The CPMS will receive the request and with CheckSocket, will check if the socket is inserted correctly and if there is enough energy in the station. If so, the station will notify the CPMS of the start of the recharging process.

10. Driver View charging status



The Driver goes to My reservation Page and clicks on the reservation. The eMSP with CheckIfStarted checks if the reservation has already started or is about to start. Then the Driver clicks on “Charging Status”. The CMPS with CheckStateCharge checks if the battery has been fully recharged and if so stops charging. The opt sequence represents the possibility for the Driver to manually stop the charging process.

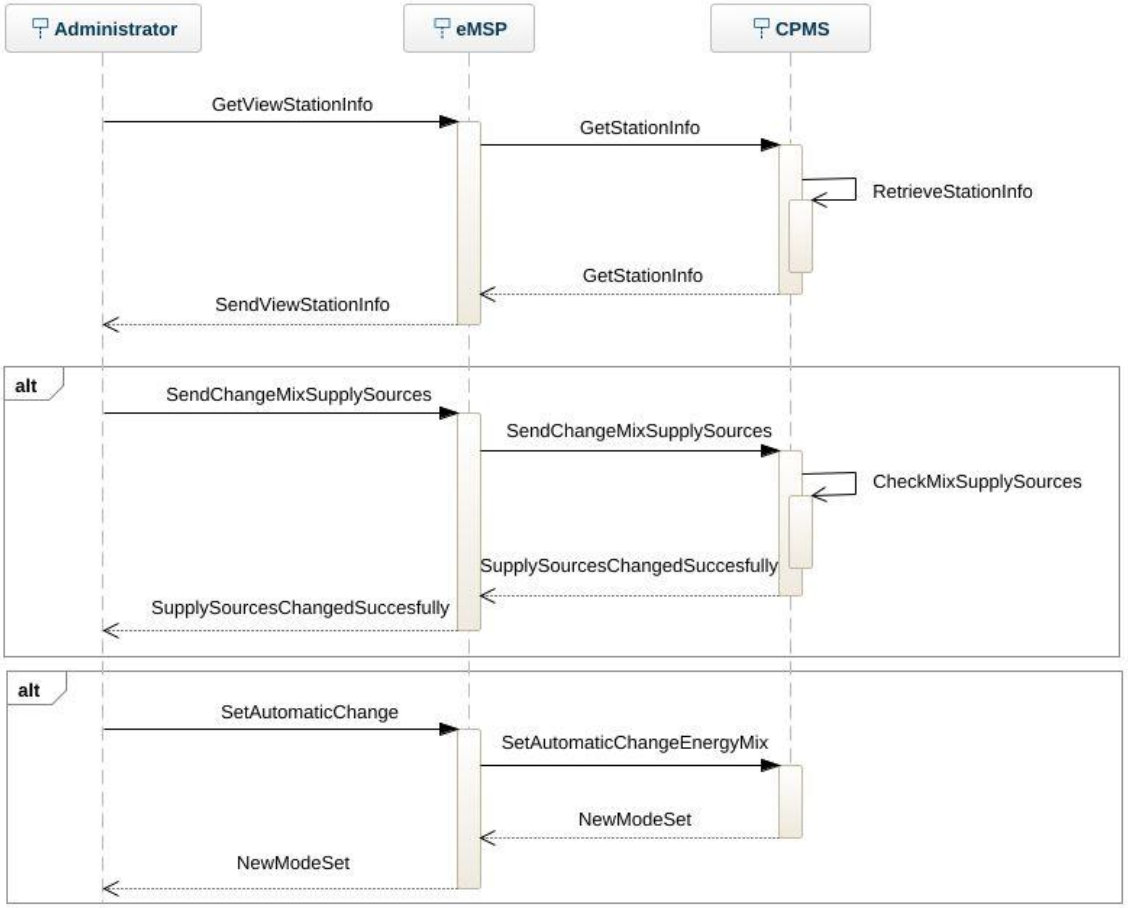
11. **Driver reach a station**



Here are shown the 3 differents alternatives that the Driver can

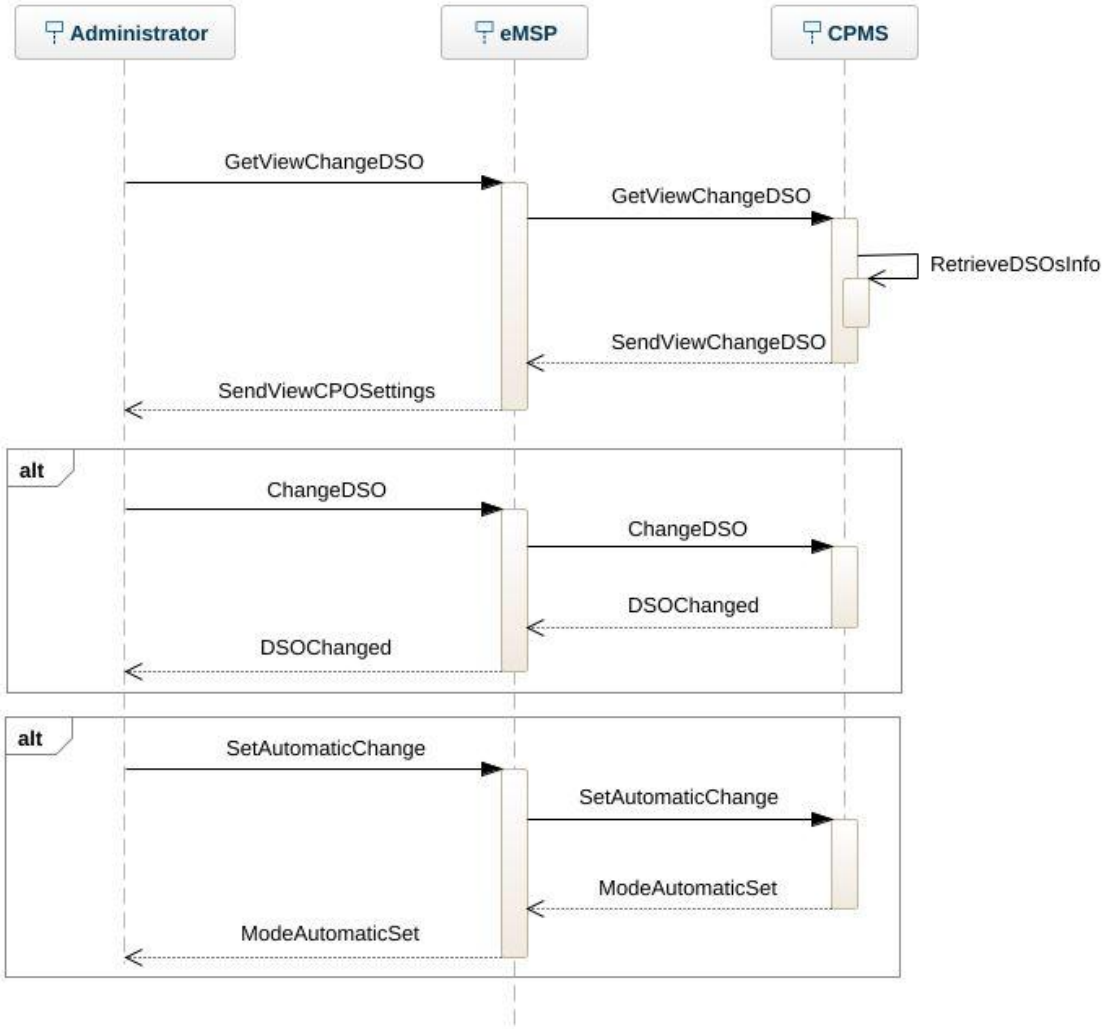
the 3 different alternatives that the Driver can follow to get the directions for the desired station.

12. Administrator modify energy mix



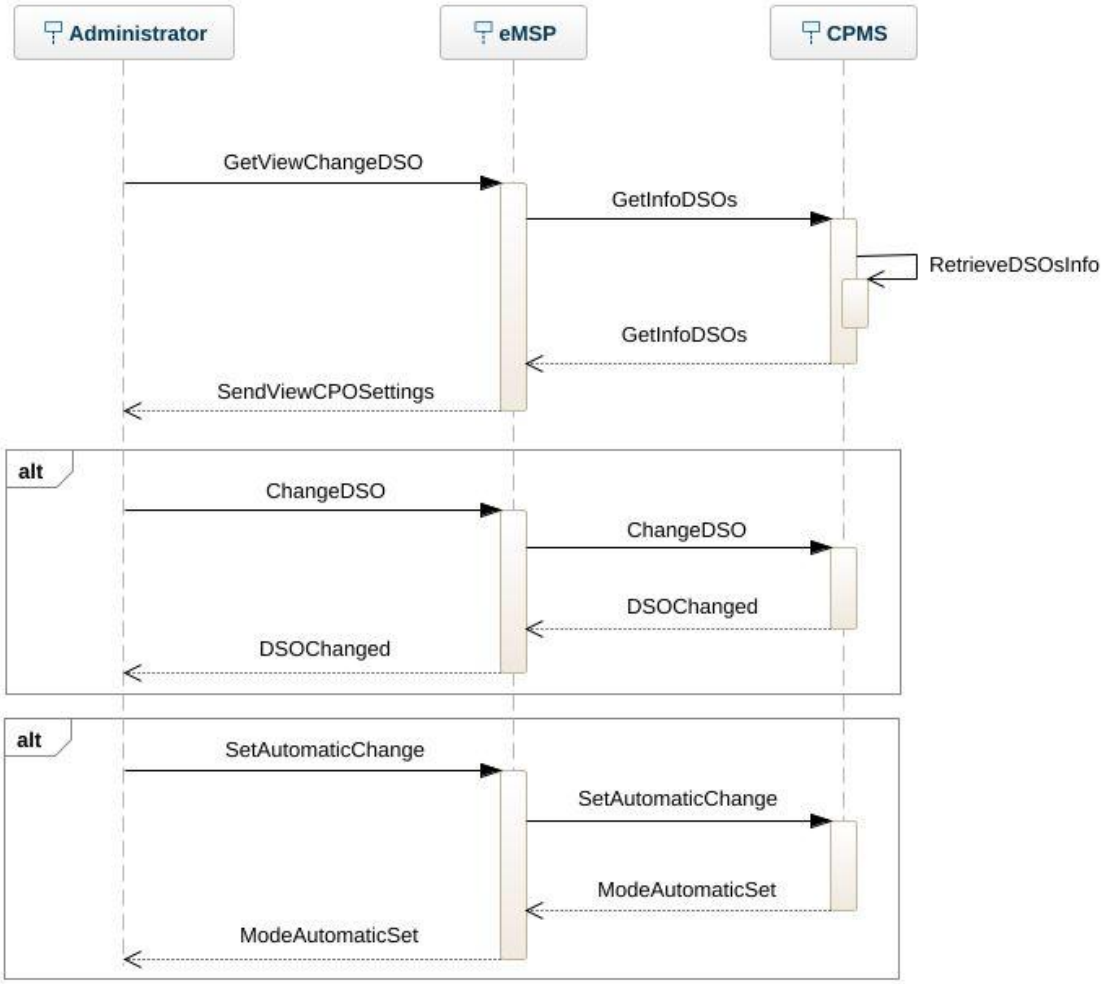
The CPMS uses RetrieveStationInfo to retrieve all the data about a station.
With CheckMixSupplySources it is checked if there is enough energy to make the change desired.
The two alternatives show the two possibilities of either selecting manually the Energy Mix desired or to set the automatic change.

13. Administrator Modify DSO



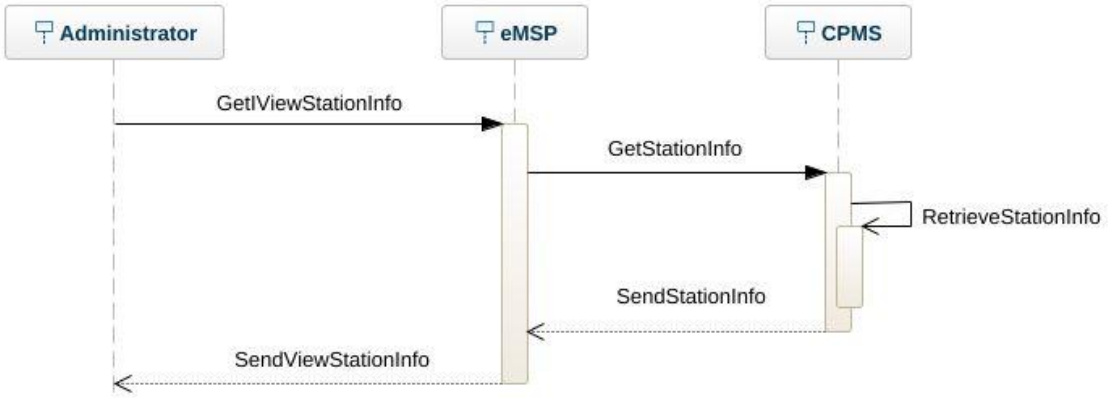
With RetrieveDSOsInfo the CPMS will retrieve all the data about his DSO. The two alternatives show the two possibilities of either selecting manually the DSO desired or to set the automatic change.

14 Administrator set an offer

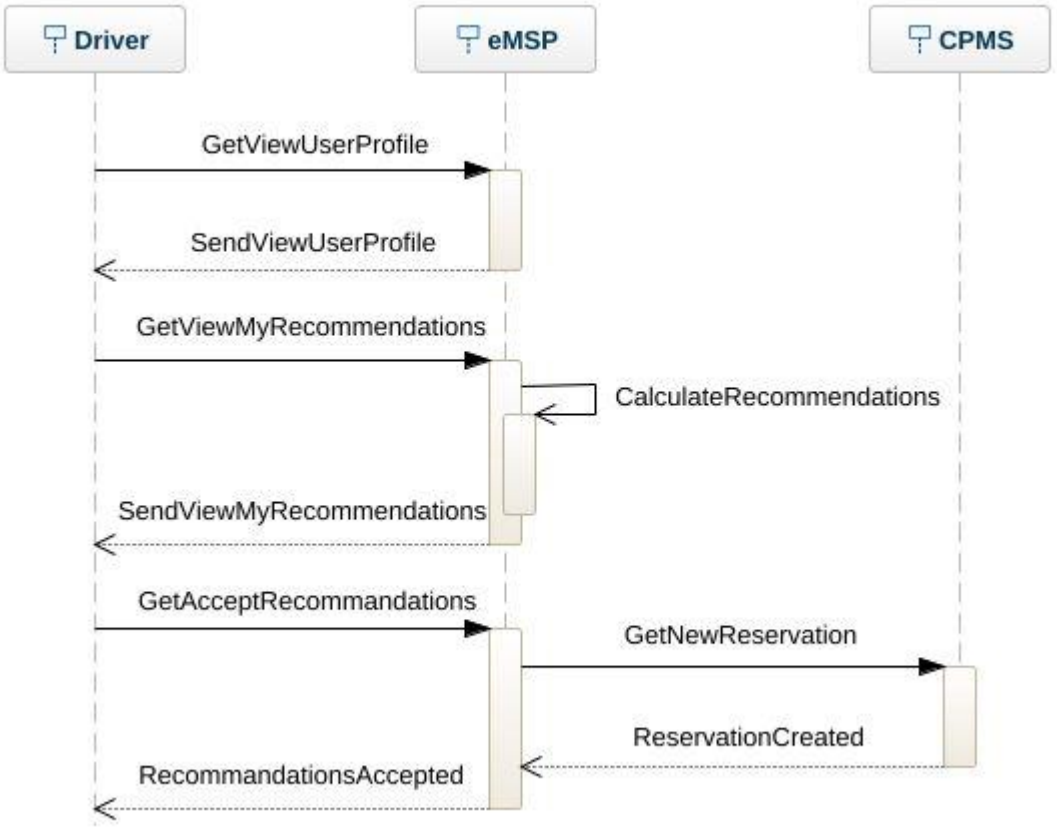


With CreateOffer the CPMS will set on the target station the new offer.

15Administrator view station Performance



16. Driver Accept a recommendation



The Driver goes on the UserProfile Page and clicks on “My Recommendations”. The eMSP handles the request and retrieves the recommendations made for the Driver. After the Driver clicks on “Accept”Recommendation”, the new reservation is created and inserted in the database.

3.2.5 Mapping on requirements

.

Use case	Requirements

Driver registration	R1, R2, R24, R25
User login to e-MALL	R24, R25, R27, R28
Driver add new vehicle	R2, R24, R25, R27
Driver change active vehicle	R2, R24, R25, R27
Driver add new payment method	R3, R24, R25, R27
Driver change active payment method	R3, R24, R25, R27
Driver make a reservation	R4, R5, R6, R7, R8, R24, R25, R27
Driver cancel a reservation	R9, R16, R24, R25, R27
Driver start the charging process	R9, R11, R24, R25, R27
Driver view charging status	R9, R12, R13, R24, R25, R27
Driver reach a station	R4, R5, R6, R10, R24, R25, R27
Administrator modify energy mix	R17, R20, R21, R22, R23, R24, R25, R28
Administrator modify the DSO	R21, R22, R24, R25, R28
Administrator view station performance	R17, R18, R19, R20, R23, R24, R25, R28
Driver accept a recommendation	R14, R15, R24, R25, R27

3.3 Performance Requirements

The system must be reliable to handle an eventual incorrect procedure and must be supported by a fast internet infrastructure that is fundamental to guarantee to the users the possibility to evaluate only the charging stations actually available, without the risk of trying to make a reservation on a socket already booked. It is also extremely important to enable proper recommendations, based on information mostly retrieved by external APIs. This performance goal could be reached relying on an internet provider that ensures the user capacity and speed required.

Secondly, the system should be able to handle many concurrent users, at least 10 000, and many concurrent inputs.

3.4 Design Constraints

3.4.1 Standard compliance
All user data should be treated in compliance with GDPR (or the local privacy law), the authority that defines how the companies that work in the EU should collect, store and handle users' personal data.
Moreover, the application should function fully on all widely used web browsers and mobile stores.
Lastly, the regulation and guidelines of external APIs must be followed.

3.4.2 Hardware limitations
This system will be made available both as a website and as a mobile application. One of the requirements the hardware has to satisfy is the possibility to access the internet through a web browser (valid method for smartphone or PC) rather than to download the application from the online store for mobiles. Moreover, the hardware of the Driver must have the possibility to create schedules according to the user's duties and his vehicles must be provided with a localization system.

3.5 Software System Attributes
eMALL system has to be available 24h/7d, without considering the scheduled maintenance breaks during the year, and both mobile application and website must be reactive and usable.

3.5.1 Reliability
The service has to guarantee a high reliability, so the system has to be available for an adequate period without interruptions. Since the system reliability depends on its subsystems components, both eMSP and CPMS must respect these parameters. The reliability of external APIs is not considered in this document and is out of system's control. To guarantee this high reliability, there should be scheduled and periodical maintenance interventions to prevent unexpected downtimes. Additionally, a duplicate of the server should be run in parallel to guarantee the service in case of failure. There should also be a backup of each database, eMSP and CPMS.

3.5.2 Availability
Since eMALL service is based on the concept that each user should find the best charging option for his vehicle according to his temporary

conditions (schedule, localization, etc…) in the minimum time possible, the system's availability should be as high as possible, in particular during the most crowded hours. It is supposed to have an availability of 99.9%, which means 9 hours/year of downtime. This goal can be reached handling the complexity of the individual components, having a high quality maintenance and duplicating server and databases, as previously explained in relation to reliability.

3.5.3 Security

The system must ensure a high standard of security since it stores and handles private and critical users' information, such as password, localization and credit card details. To achieve this goal, critical data should be saved on a DB hashed and salted and every input and request by the user must be sanitised. Moreover, there should be an authorization check at every API endpoint within the system.

3.5.4 Maintainability

The system should be easy to maintain and to be modified to correct faults, improve performance or to be adapted to a changed environment. To achieve this goal, the system should be based on one or more software patterns, which (possibly supported by a clear documentation) guarantee and facilitate future extensions of the system, with the possibility to implement new features without the necessity of rethinking the whole concept.

3.5.5 Portability

Since the system can be supported both by mobile application and website, it should be properly designed to be visualised and to preserve its functionalities regardless of the device used (PC, different smartphone models).

4 Formal Analysis

4.1 Alloy code

The formal analysis of the system is represented through the use of Alloy code in order to verify the consistency of the model presented and the needed constraints.

```
// Ids, integer values and dates will not be considered for simplicity

-- SIGNATURES --


open util/boolean

sig Position {}
sig Recommendation{}

sig CPO {
	//cpoId: one CPOId
	availableDSO: some DSO,
	activeDSO: one DSO
}

sig DSO {
	//dsoId: one DSOId,
	//energyPrice: one Int
}

abstract sig SocketType{}
one sig SLOW extends SocketType {}
one sig FAST extends SocketType {}
one sig RAPID extends SocketType {}

abstract sig SocketStatus{}
one sig AVAILABLE extends SocketStatus {}
one sig UNAVAILABLE extends SocketStatus {}
one sig CHARGING extends SocketStatus {}

abstract sig RechargeStatus{}
one sig INPROGRESS extends RechargeStatus {}
one sig PAYED extends RechargeStatus {}

abstract sig User {
	//userId: one UserId,
	//name: one String,
	//surname: one String,
	//documentId: one DocumentId,
	//email: one Email,
	//password: one String
	}

sig CPOAdministrator extends User{
```

```
        cpo: one CPO
}

sig EVDriver extends User {
        paymentMethods: set PaymentMethod,
        vehicles: some Vehicle,
        recommendation: one Recommendation
}

sig Vehicle {
        //licensePlate: one String,
        type: one SocketType,
        //battery: one Int,
        isActive: one Bool
}

sig Socket {
        //socketId: one socketId,
        type: one SocketType,
        active: one Bool,
        status: one SocketStatus
}

sig ChargingStation {
        //stationId: one StationId,
        //energyMix: one Int,
        //price: one Int,
        //battery: one Int,
        position: one Position,
        cpo: one CPO,
        sockets: some Socket,
        options: some ChargingOption
}

sig PaymentMethod {
        //cardNumber: one String
        isActive: one Bool
}

sig Payment {
        //paymentId: one PaymentId
        //amount: one Int,
        paymentMethod: one PaymentMethod
} {
        all p : Payment | p in Recharge.payment
}

sig Recharge {
        payment: lone Payment,
        status: one RechargeStatus,
}
```

```
sig Reservation {
	//reservationId: ReservationId
	//fromDate: one Date,
	//toDate: one Date,
	socket: one Socket,
	evdriver: one EVDriver,
	vehicle: one Vehicle,
	recharge: lone Recharge
}

sig ChargingOption {
	//cost: one Int,
	type: one SocketType,
	isOffer: one Bool
}


-- FACTS


fact uniqueSocketsOfChargingStations {
	no disjoint c1, c2: ChargingStation | some s: Socket | (s in c1.sockets) && (s in
c2.sockets)
}

fact uniqueRechargeForReservations {
	no disjoint r1, r2: Reservation | some c: Recharge | (c in r1.recharge) && (c in
r2.recharge)
}

fact uniqueVehiclesOfEVDrivers {
	no disjoint d1, d2: EVDriver | some v: Vehicle | (v in d1.vehicles) && (v in d2.vehicles)
}

fact uniquePaymentsInRecharges {
	no disjoint r1, r2: Recharge | r1.payment = r2.payment
}

fact uniquePositions {
	no disjoint s1, s2: ChargingStation | s1.position = s2.position
}

fact uniquePaymentMethodsOfEVDrivers {
	no disjoint d1, d2: EVDriver | some p: PaymentMethod | (p in d1.paymentMethods)
&&  (p in d2.paymentMethods)
}

fact uniqueChargingOptions {
	no disjoint c1, c2: ChargingStation | one o: ChargingOption | (o in c1.options) && (o
in c2.options)
```

```
}


fact uniqueChargingOptionsInChargingStations {
	all disjoint o1, o2: ChargingOption, c: ChargingStation | (o1 in c.options && o2 in
c.options) implies (o1.type != o2.type or o1.isOffer != o2.isOffer)
}

fact uniqueRecommendations{
	no disjoint d1,d2: EVDriver | d1.recommendation = d2.recommendation
}

// No multiple Recharges happening at the same time
fact uniqueRecharges{
	no disjoint r1, r2: Reservation | r1.socket = r2.socket && r1.recharge.status =
INPROGRESS && r2.recharge.status = INPROGRESS
}

// All EVDrivers has at least one active vehicle in his vehicles
fact allEVDRiverHasVehicle {
	all u: EVDriver | one v: Vehicle | v in u.vehicles && v.isActive = True
}

// All EVDrivers has at least one active payment method in his paymentMethods
fact allEVDRiverHasPaymentMethod {
	all u: EVDriver | one p: PaymentMethod | p in u.paymentMethods && p.isActive =
True
}

// CPOs exists only if they own one or more charging stations
fact allCPOHasChargingStation{
	all c: CPO | some s: ChargingStation | c = s.cpo
}

// All CPOs has at least one active DSO in his availableDSO set
fact allCPOhasDSO {
	all c: CPO | one d: DSO | d = c.activeDSO && d in c.availableDSO
}

// Reservation associated to vehicle owned by the driver who created the reservation
fact onlyVehicleOfDriverInHisReservation{
	all r: Reservation, d: EVDriver, v: Vehicle | (d = r.evdriver && v = r.vehicle) implies v in
d.vehicles
}

// Reservations for a socket can be made only by drivers who owns a vehicle of compatible
recharging speed
fact allVehicleCompatibleWithSocket{
	all r: Reservation | one v: Vehicle | v =r.vehicle && (v.type = SLOW implies
r.socket.type = SLOW) && (   v.type = FAST implies r.socket.type = SLOW or r.socket.type =
FAST)
```

```
}

// All available sockets have an available ChargingOption associated to the station
fact onlyChargingOptionsForAvailableSockets{
	all c: ChargingStation, s: Socket | s in c.sockets implies (one o: ChargingOption |
o.type = s.type && o in c.options && o.isOffer = False)
}

// All ChargingOptions are present only if there is a corresponding socket of the same speed
fact allChargingOptionHasSocket{
	all c: ChargingStation, o: ChargingOption | o in c.options implies (one s: Socket |
o.type = s.type && s in c.sockets)
}

// Recharge status is payed only if associated with a payment
fact rechargeStatusPayed {
	all r: Recharge | r.status = PAYED iff (one p: Payment | p in r.payment)
}

// Payments associated to correct PaymentMethod
fact allPaymentAssociatedToDriver {
	all p: Payment | one d: EVDriver, s: Reservation, r: Recharge | p = r.payment && r =
s.recharge && d = s.evdriver && p.paymentMethod in d.paymentMethods
}

// Recharge is in progress only if the socket is in charging status
fact allRechargeInProgressIfSocketInCharging{
	all r: Recharge, t: Reservation, s: Socket | r = t.recharge && s = t.socket && r.status =
INPROGRESS implies s.status = CHARGING
	all s: Socket | s.status = CHARGING implies (one r: Recharge, t: Reservation |  r =
t.recharge && s = t.socket && r.status = INPROGRESS)
}

// Two vehicles can not be recharged at the same time
fact uniqueRechargeForEVDriver{
	no disjoint r1,r2: Reservation | r1.evdriver = r2.evdriver && r1.recharge.status =
INPROGRESS && r2.recharge.status = INPROGRESS
}

fact allSocketsConnected {
	all s: Socket | one c: ChargingStation | s in c.sockets
}

fact allPaymentMethodsConnected {
	all p : PaymentMethod | p in EVDriver.paymentMethods
}

fact allDSOConnected {
	all d: DSO | some c: CPO | d in c.availableDSO
}
```

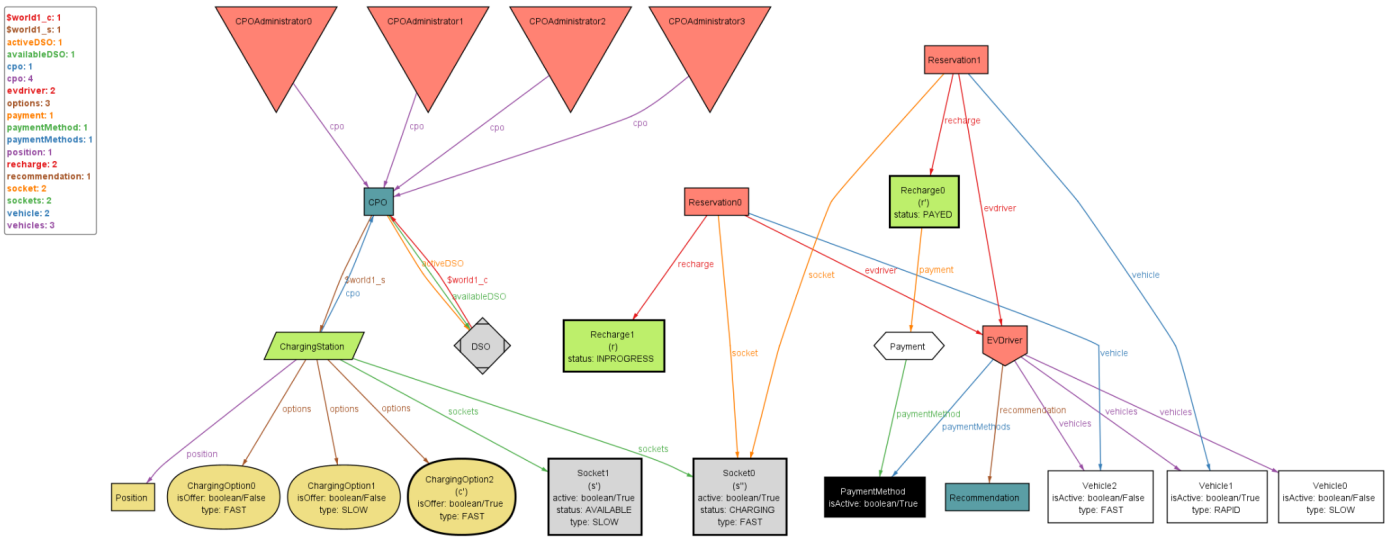
```
fact allVehiclesConnected {
	all v: Vehicle | one d: EVDriver | v in d.vehicles
}

fact allChargingOptionsConnected {
	all c: ChargingOption | one s: ChargingStation | c in s.options
}

fact allRechargesConnected {
	all r: Recharge | one s: Reservation | r in s.recharge
}

fact allRecommendationsConnected {
	all r: Recommendation | one d: EVDriver | r in d.recommendation
}

fact allPositionsConnected {
	all p: Position | one c: ChargingStation | p in c.position
}


-- DYNAMIC MODELING


pred createReservation [d: EVDriver, v: Vehicle, s: Socket, r: Reservation] {
	r.evdriver = d
	r.vehicle = v
	r.socket = s
}

pred addVehicle [d, d': EVDriver, v: Vehicle] {
	d'.vehicles = d.vehicles + v
}

pred addSocket [c, c': ChargingStation, s: Socket] {
	c'.sockets = c.sockets + s
}


-----------------------------------------------------------------------------------------------------------------------
--

pred general {
	#EVDriver >= 1
	#Socket >= 4
	some s: Socket | s.status = AVAILABLE
	some s: Socket | s.status = CHARGING
	#ChargingStation>=1
	#Vehicle >= 5
	#Recharge >= 1
	#Reservation >=3
```

```
        some r: Recharge | r.status = INPROGRESS
        some r: Recharge | r.status = PAYED
        #CPO>=1
        #DSO>=2
        some c: ChargingOption | c.isOffer = True

}

pred casual {}

pred world1 {
        #CPO>=1
        #EVDriver = 1
        #Socket >= 4
        some s: Socket | s.status = AVAILABLE
        some s: Socket | s.status = CHARGING
        #ChargingStation>=2
        #Vehicle >= 2
        #Reservation >=2
        #CPO=1
        some c: ChargingOption | c.isOffer = True
}

run world1 for 5
```

4.1.1 First model

In the first model, corresponding to predicate world1, we focus on the relations that intercurr between evdrivers, vehicles, sockets, reservations, recharges and payments. With this module we want to underline the following:

- Every Driver has at least an active payment method and an active vehicle
- Every recharge, if not in progress, is linked to a unique payment associated to a payment method of the corresponding Driver
- Every vehicle is associated to a reservation only if it is compatible with the charging speed offered by the chosen socket



4.1.2 Second model

In the first model, corresponding to predicate world2, we focus on the relations that intercurr between CPO administrators,dsos, charging stations, and charging options. With this module we want to underline the following:

- Every CPO has only an active DSO between all available ones while multiple CPOs can have the same active DSO
- Every charging station always offers a standard charging option for every type of socket connected to it

4.1.3 Dynamic Modelling

Some dynamic behaviour of the system have also been modelled:

- Creation of a reservation
- Addition of a vehicle for an evdriver
- Addition of a socket for a charging station

5 Effort spent

5.1.1 Umberto Colangelo

Section	Time spent
Introduction	7 h
Overall description	6 h
Specific requirements	17 h
Formal analysis	1 h
Reasoning	10 h

Total time	41 h

5.1.2 Roberto Cialini

Section	Time spent
Introduction	5 h
Overall description	6 h
Specific requirements	5 h
Formal analysis	19 h
Reasoning	10 h

Total time	45 h

5.1.2 Vittorio La Ferla

Section	Time spent
Introduction	7 h
Overall description	6 h

Specific requirements	18 h
Formal analysis	2 h
Reasoning	10 h

Total time	43 h