POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

eMall – e-Mobility for All

RASD
Software Engineering 2

TEAM MEMBERS:

Andrea Piras - 10725972
Emanuele Santoro - 10676582
Andrea Sanguineti - 10739788

Academic Year: 2022-23

Contents

Contents **i**

1 INTRODUCTION **1**
1.1 Purpose . . . 1
1.1.1 Goals . . . 2
1.2 Scope . . . 2
1.2.1 World Phenomena . . . 3
1.2.2 Shared Phenomena . . . 3
1.3 Definitions, Acronyms, Abbreviations . . . 5
1.3.1 Definitions . . . 5
1.3.2 Acronyms . . . 6
1.3.3 Abbreviations . . . 6
1.4 Revision History . . . 7
1.5 Reference Documents . . . 7
1.6 Document Structure . . . 7

2 OVERALL DESCRIPTION **9**
2.1 Product Perspective . . . 9
2.1.1 Scenarios . . . 9
2.1.2 Class Diagram . . . 12
2.1.3 Dynamic Class Behaviour Models . . . 13
2.2 Product Functions . . . 14
2.3 User Characteristics . . . 15
2.4 Assumptions, Dependencies and Constraints . . . 16
2.4.1 Domain Assumptions . . . 16

3 SPECIFIC REQUIREMENTS **17**
3.1 External Interface Requirements . . . 17
3.1.1 User Interfaces . . . 17
3.1.2 Hardware Interfaces . . . 19
3.1.3 Software Interfaces . . . 19
3.1.4 Communication Interfaces . . . 19
3.2 Functional Requirements . . . 20
3.2.1 Use Case Diagrams . . . 20
3.2.2 Use Cases . . . 22
3.2.3 Sequence Diagrams . . . 30
3.2.4 eMSP Requirements . . . 40
3.2.5 CPMS Requirements . . . 42

3.2.6 Interaction Requirement . . . 43
3.2.7 Requirements mapping on Goals . . . 45
3.2.8 Domain Assumption mapping on Goals . . . 47
3.2.9 Explicit Goal Mapping . . . 47
3.3 Performance Requirements . . . 58
3.4 Design Constraints . . . 58
3.4.1 Standards Compliance . . . 58
3.4.2 Hardware Limitations . . . 58
3.5 Software System Attributes . . . 59

4 FORMAL ANALYSIS 60
4.1 Alloy Code . . . 60
4.1.1 Results . . . 65
4.2 Models . . . 65
4.2.1 First Model . . . 65
4.2.2 Second Model . . . 65
4.2.3 Dynamic Model . . . 67

5 EFFORT SPENT 68

6 REFERENCES 69
6.1 Reference Documents . . . 69
6.2 Reference Sites . . . 69

1 | INTRODUCTION

1.1. Purpose

Electric vehicles are the key technology to reduce the environmental impact of road transport, a sector that accounts for 16% of global emissions. Recent years have seen exponential growth in the sale of electric vehicles together with improved range, wider model availability and increased performance.

This trend is projected to continue in the future, and for this reason more charging stations are being built each year, in order to satisfy the increasing demand for energy for these electric vehicles.

In this context, it is fundamental that the communication between the charging stations and the drivers is managed in such a way that it introduces minimal interference and constraints on our daily schedule.

eMall is going to be a platform that permits the aforementioned communication through its subsystems eMSP (e-Mobility Service Provider), which is the app used by some users, like drivers, and the CPMS (Charge Point Management System), used by the owner of the charging stations or also called Charging Point Operator (CPO).

Our main goal is to develop both subsystems and in particular, describe how the eMSPs can communicate with different CPMSs. Through an eMSP, drivers can easily choose where to charge their vehicles and easily book the charging process and pay for it. On the other hand, CPOs can use their CPMS to manage their charging stations, in order to improve and speed up the energy management and the charging process.

1.1.1. Goals

ID	Description
G1	Allow Drivers to check nearby charging stations and see info about their prices, special offers and availability.
G2	Allow Drivers to create and delete bookings for charging their vehicle in a charging station.
G3	Allow Drivers to manage and monitor their charging process.
G4	Allow Drivers to pay CPOs for the charging process provided.
G5	Proactively suggest to the Drivers where to go to charge their vehicle.
G6	Allow CPOs to manage the charging prices and the criteria of energy acquisition in their charging stations.
G7	Allow CPOs to connect to all their charging stations and monitor them.

Table 1.1: Goals

1.2. Scope

The eMall platform objective is the integration between two subsystems: the eMSP and the CPMS.
Even though the eMSPs could be handled directly by third-party providers, in this project both subsystems are managed by eMall.
In this project, only the subsystems are described, since eMall is considered an already implemented service accessible via web and therefore only mentioned if the actors or the subsystems interact with it. Among the several stakeholders to consider, this document concerns only drivers and CPOs. Occasionally DSOs (Distribution System Operators) are mentioned, but they should not be considered part of the project scope since they do not use the platform and are therefore external entities.

1.2.1. World Phenomena

ID	Description
WP1	The driver wants to charge his vehicle.
WP2	The driver drives the vehicle to the booked charging socket.
WP3	The CPO wants to connect one of his charging stations.
WP4	The CPO decides to start using a new eMSP.
WP5	A DSO updates its energy price.

Table 1.2: World Phenomena

1.2.2. Shared Phenomena

ID	Description	W. C.	M. C.
SP1	The Driver inserts his personal data and a payment method in order to register himself on the platform	**X**	
SP2	The Driver creates a physical connection between the system and his vehicle.	**X**	
SP3	The Driver receives a suggestion when he needs to charge the vehicle.		**X**
SP4	The energy level of the battery vehicle changes.	**X**	
SP5	The Driver follows a new navigation route on his vehicle.	**X**	
SP6	The Driver checks information about the availability and locations of nearby charging stations	**X**	
SP7	The Driver books a charge in a charging station.	**X**	
SP8	The Driver receives a notification that the booking process was successful		**X**
SP9	The Driver receives a notification that the booking was successfully cancelled		**X**
SP10	The Driver plugs his vehicle in the charging socket	**X**	

SP11	The Driver gets notified when trying to start a charging process even though his vehicle is not well-connected		**X**
SP12	The Driver receives a notification when the charging process is complete.		**X**
SP13	The CPO checks the internal status of one of the managed charging stations	**X**	
SP14	The CPO sets a special offer.	**X**	
SP15	The CPO changes energy acquisition criteria for a charging station	**X**	
SP16	The CPO changes energy revenue criteria for a charging station	**X**	
SP17	In a charging station a socket becomes available after a charging process is finished	**X**	

Table 1.3: Shared Phenomena

1.3. Definitions, Acronyms, Abbreviations

1.3.1. Definitions

Definition	Description
Energy Provider	The entity providing energy to the charging station, it can either be a DSO or the internal batteries of that charging station.
Driver	The person that will use the eMSP application. For shortening reasons, only male pronouns are used to address the driver.
Charging Point Operator	The legal owner of charging stations that will use the CPMS. For shortening reasons, only male pronouns are used to address the driver.
Charging Station	Physical place managed by a CPO where drivers can charge their vehicle. It has one or more charging sockets. When a charging station is "available, " one of its sockets is available.
Charging Socket	Plug where drivers connect their vehicle to charge it. It can support different charging types. When a charging socket is "available" it means that it has no booked charging process or no one is connected to it.
Maximum Amount Of Time	The maximum amount of time a Driver can book a charge before the arrival time. It is needed to avoid a charging socket being occupied for a long time without being used. This variable can be decided when implementing the system
Charging Type	It defines the speed of the charging process.
User	Either the CPO or the Driver.

Table 1.4: Definitions

1.3.2. Acronyms

Acronyms	Description
eMall	e-Mobility for All
eMSP	e-Mobility Service Provider
CPO	Charging Point Operator
CPMS	Charge Point Management System
DSO	Distribution System Operator
FRE	Functional REquirement
NFRE	Non-Functional REquirement
RASD	Requirements Analysis and Specification Document
W.C.	World Controlled
M.C.	Machine Controlled
ID	IDentifier
API	Application Programming Interface

Table 1.5: Acronyms

1.3.3. Abbreviations

Abbreviation	Description
Gx	Goal number X
WPx	World Phenomenon number X
SPx	Shared Phenomenon number X
Dx	Domain Assumption number X
FREx	Functional Requirement number X
NFREx	Non-Functional Requirement number X
Ux	Use Case number X
opt	Optional
alt	Alternative

Table 1.6: Abbreviations

1.4. Revision History

Version Number	Date	Description
1.0	22/12/2022	First completed version of the documents.
1.1	23/12/2022	Rephrased and fixed minor bad wordings about bookings.

Table 1.7: Versions history

1.5. Reference Documents

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

1.6. Document Structure

Section 1 Introduction about the purpose and scope of the system. Discuss the main world and shared phenomena concerning our application's domain and goals. Furthermore, we sum up all the definitions and abbreviations in order to have a better comprehension of the following chapters.

Section 2 Introduction of different scenarios about the multiple interactions that the application could face. After that, there is a structural description of the system represented by various graphs such as class diagrams and state charts with all of their main characteristics. In the product functions section, there are multiple descriptions of all the possible functionalities present inside the application. In the last part, there is a list of the domain assumptions and the characteristics of the users who will exploit the application.

Section 3 This is the main part of the document, at the beginning, there is a description of the software and communication interfaces. After this, it's shown all the use case diagrams of the systems and the corresponding use cases, followed by the respective sequence diagram. Then, the lists of the requirements are presented along with their description, and some tables that map goals and domain assumptions with the respective requirements. In the end, there is a list of the performance requirements and design constraints of the system.

Section 4 This section contains a formal description of the important parts of the system using the Alloy language.

Section 5 This section shows how much time every student spent working on the documents

Section 6 This section is made to point out all the references and tools used during the creation of this document.

2 | OVERALL DESCRIPTION

2.1. Product Perspective

2.1.1. Scenarios

1. **Driver Registration**
   Paolino is a professor in Milan and decides to start using the eMSP application just published by his students on the app store. He downloads the app and then starts the signup process. The system asks him to insert various information about his personal data. Paolino fills all the text boxes and clicks confirm. The system then asks him for a valid payment method. After inserting it, a pop-up appears confirming its validity and finally, he receives a confirmation email from eMSP to conclude the registration. The system shows Paolino that the registration has been completed.

2. **Check nearby charging stations**
   Franco is going on a trip with his new electric car and wants to take a break. Since he would have had to stop at a charging station in any case, he wants to take advantage of this break and charge his car so he can get to the destination with no more stops. Franco, already registered to the eMSP, opens the application on his device, so he can check for charging stations in the selected area. Since Franco has planned to rest no more than half an hour, he selects from the eMSP application “Fast-charge” and gets a view of the map containing the charging stations supporting that charge type.

3. **Booking of a charge**
   Renato needs to have a completely charged car for a work trip abroad after the weekend. He opens the eMSP application on his phone to check the map of nearby charging stations. He then chooses one station supporting the "Slow Charge", since he is not in a rush. After selecting it, the application shows the selected station page, containing the price per kWh and if existing, the corresponding offer, for the selected charging type. Since the station is not yet available, the app shows the

remaining time before it is freed, which amounts to 60 minutes. As soon as the station becomes available he presses the "Book charge" button and selects the time of arrival. After receiving the notification that the booking was successful, along with the ID of the booked charging socket, Renato gets in his car and starts driving to the station.

4. **Car charging**
   It's 16:45 and Maristella has booked a charge for 17:00 for her truck. She arrives in time, parks her majestic truck and plugs in the charging cable to the charging socket. Finally, she presses the "Start charging" button displayed in the eMSP application on her device and starts the charging process. Her device now shows the estimated time until the recharge completes; meanwhile, Maristella decides to go to the station's bar nearby for having her tea time. She's having a good time when she suddenly remembers her dating invite. Maristella takes her phone, opens the eMSP application and presses the "Stop charging" button; finally she receives a notification telling her that the payment was successful, so Maristella goes back to the charging station, unplugs the car and leaves happily with her truck.

5. **Receiving a charging Suggestion/Advice**
   Cinzia is a professor in the province of Rome. She has been using the eMSP application for a while and has already authorized the system to read data about her calendar and her vehicle information such as navigation routes, location, and battery status. In the afternoon she has a meeting with one of her students to discuss about a very important project. During lunchtime, she receives a notification on her phone from the eMSP application, saying that there is a personalized suggestion for her. She clicks on the notification, which opens the suggested station info page. She notices that it's on her route to the school and has set an offer for the fast-charge sockets. Since her car's battery level is at 20% she decides to take advantage of the offer and starts the booking procedure.

6. **Applying offers for the charging prices**
   Irene is employed by Be-Charge, a well-known CPO. She manages charging stations in the province of Naples and she decides to set a special offer to attract possible new customers in the newcome charging station that opened last week. She logs in to the CPMS installed on her company computer and selects one of these charging stations. She then clicks on the "Set a new offer" button. The application asks her to fill out a form where she has to select the amount of discount and for which charging types and the expiration date for this offer. She sets a 10% discount for all the available charging types and sets the expiration date to 7 days from that moment and finally clicks on the "Submit" button. She repeats this procedure for the other new charging stations and then logs out of the application.

7. **Update energy criteria**
   Alice realizes that during the upcoming holidays a high percentage of the population will probably make road trips. She deduces that her charging stations will experience an increase in the number of customers. In order to withstand the high amount of energy that the charging stations will need to supply, she decides to update the criteria used by the CPMS to select the energy provider. She logs into the system and selects one of the most used stations. On its information page, she selects the "Update energy criteria" button. The system then shows the page with a form containing the available criteria. She chooses that the system will select the cheapest provider with enough capacity to power all sockets at full power and updates the revenue percentage for the energy sales from 10% to 15%. She then selects the "Update" button and waits for the system to send a confirmation. After repeating this procedure for all the charging stations that she expects will experience an increased influx, she logs out.

8. **Add a new charging station**
   Francesco is working in his office when he receives an e-mail from his supervisor stating that the construction of two more charging stations in Pavia has just finished and they need to be added to the system. Therefore he logs into the CPMS application and selects the “Add charging station” button. The application opens a prompt asking him to select the station from the list of detected ones and he selects one of the two. The application then shows a form asking for the new charging station information, such as name, location and number of sockets. After inserting them, he clicks the "Submit" button. He is then redirected to the energy criteria page, where he only sets the revenue percentage for the energy sales to 10%, leaving other fields empty. After receiving a confirmation notification he repeats the process for the other station and finally sends an e-mail to his supervisor to inform him that the task has been completed.

The previously presented scenarios are the ones regarding the description of the subsystems of this document. In order to clarify how the CPO can retrieve his credentials to access the CPMS and use it, another scenario is introduced. This scenario does **not** belong to the scope of this document.

9. **CPO registration**
   Marta just started a new company of charging stations in Foggia. In order to increase its number of customers, she decides to start using the eMall platform. She searches for eMall on her favourite search engine and, once found, she clicks on its website and then on "Register as CPO". The system asks her to insert various information about the organization and then checks its validity. She then receives a confirmation email with a download link to the CPMS software and the credentials

to log in. She downloads the software and installs it.

2.1.2. Class Diagram

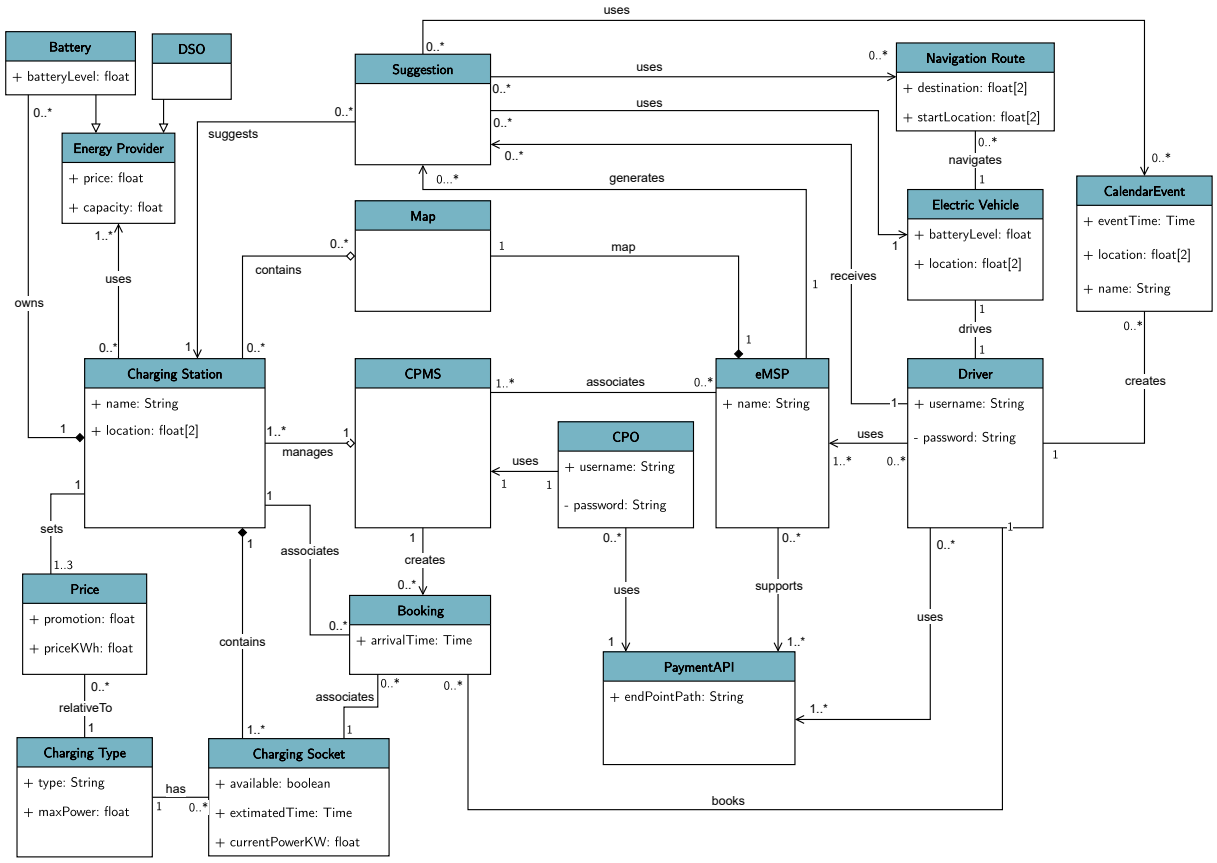

Figure 2.1: Class Diagram

In this class diagram:

- Locations are represented with an array of 2 floats representing latitudes and longitudes.
- The Time type is used to represent a timestamp.
- The endPointPath attribute of the PaymentAPI class represents the URL from where the API sends requests.

2.1.3. Dynamic Class Behaviour Models

The state diagrams listed below show the behaviour of the eMSP and the CPMS applications in their entirety.

- **eMSP**

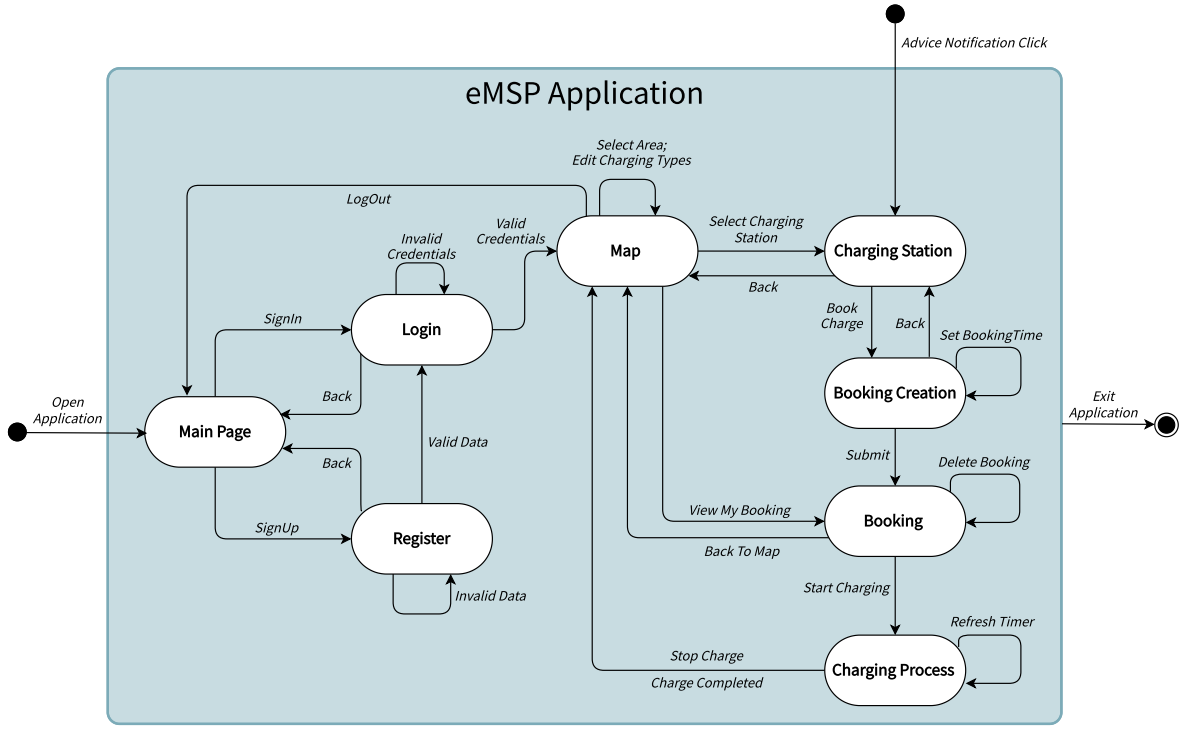

Figure 2.2: eMSP state diagram

- **CPMS**

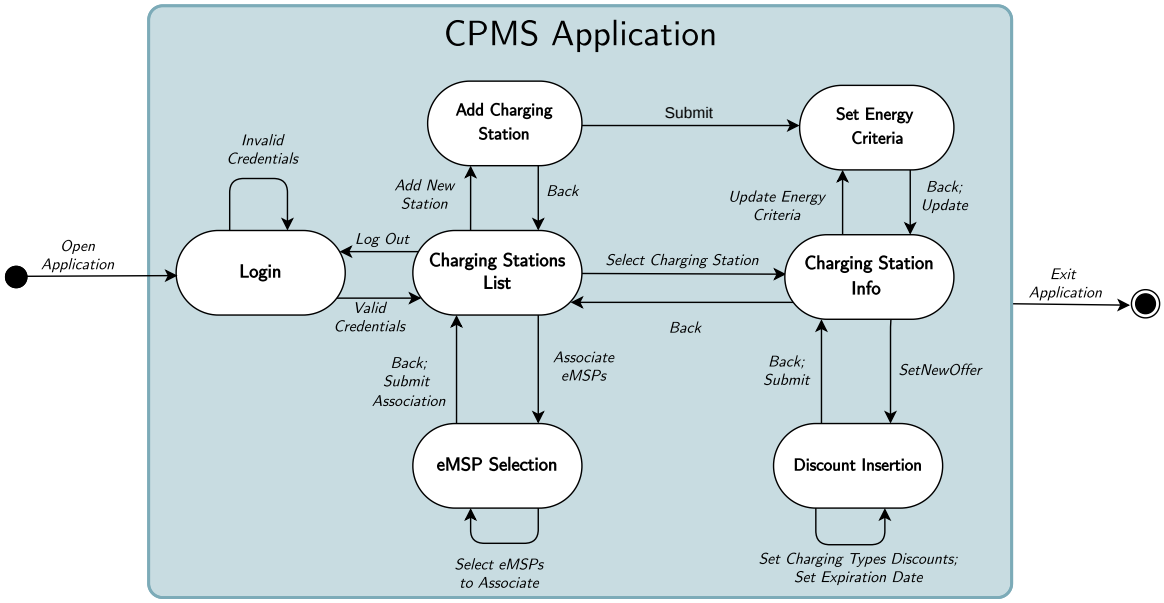

Figure 2.3: CPMS state diagram

2.2. Product Functions

In this section, the main functionalities of our system are presented and described in more detail.

Functionalities offered to the drivers through the eMSP:

- **Check charging stations in the surrounding area:** The driver can check nearby charging stations through a map of available ones. They have the possibility to select a specific station, in order to get more details about it, such as its availability, the charging price and any active offers.
- **Book a charge:** The driver can book a charge at a specific station up to a maximum amount of time in advance, through the details page of said station. They can select the type of charging socket, if available, that they want to use to charge the vehicle.
- **Start the charging process:** The driver, once they have driven to the charging station they booked in advance, can initiate the charging process through the application. If the vehicle is not well connected to the socket, the application will send a notification to the driver.
- **Proactive charge suggestion:** The driver can receive a suggestion to charge their vehicle based on their schedule and route to their destination. The system will evaluate the charging stations near the driver's route and if the prices are

advantageous and the charging can fit into their schedule it will send a suggestion to the driver through the application.

Functionalities offered to the CPOs through the CPMSs:

- **Charging Stations Management:** The CPO is able to manage all his charging stations: he can decide with which eMSPs to associate his CPMS and which charging stations will be managed by it.
  For those charging stations, the system will automatically handle the energy acquisition following the criteria set by the CPO, for example, based on the energy providers' prices, and their maximum power delivery capacity.
  For all the charging stations connected to the CPMS, the CPO will be able to set special offers for the charging prices and to visualize info about their internal status, such as the amount of energy available for the batteries, if available, the number of vehicles being charged and, for each charging vehicle, they can see the amount of power absorbed.

2.3. User Characteristics

The system can be exploited by the following actors:

1. **Unregistered Driver**
   A driver who needs to register to the eMSP platform before being able to use any of its functionalities.
2. **Driver**
   A driver that is registered on the eMSP platform and can use all its functionalities.
3. **CPO**
   A CPO registered on eMall who can use all the functionalities offered by its CMPS.

2.4. Assumptions, Dependencies and Constraints

2.4.1. Domain Assumptions

ID	Description
D1	The Driver's vehicle is electric and has a battery able to be recharged with all the charging socket.
D2	The Driver needs to know his personal data before signing up.
D3	Every time the driver books a charging process then he will show up in time at the charging station.
D4	When the Driver shows up during the time slot he booked, he'll always find his booked charging socket available.
D5	Every time the recharging process ends the driver leaves the station with his vehicle, which he first disconnects from the socket.
D6	If a vehicle is connected to a charging socket, then it delivers energy only after a driver starts the charging process booked for that socket.
D7	The energy deployed by the charging socket is only used to recharge the vehicle battery.
D8	Each charging socket has a unique ID relative to its charging station.
D9	The driver is able to create a connection between his device and his vehicle that permits to retrieve from it reliable data about the navigation system, the vehicle's battery status and location.
D10	There exists a standard communication protocol that permits the charging sockets and charging stations to communicate with the CPMS in order to notify it of events like vehicle connection and disconnection or data about the charging status (completed, in process, remaining charging time).
D11	There exists a uniform API that allows retrieval information about DSOs, such as price and energy capacity.
D12	There exists an external API that handles payments.
D13	There exists an API endpoint where the eMSP can retrieve the map of a certain area.
D14	There exists an API endpoint on eMall where the CPMS can retrieve a list of all eMSPs.
D15	There exists an API endpoint on eMall where the CPMS can confirm the credentials inserted by the CPO

Table 2.1: Domain Assumptions

3 | SPECIFIC REQUIREMENTS

3.1. External Interface Requirements

3.1.1. User Interfaces

The following mockups are presented here just to show an idea of the application that will be in use by the Drivers and the one available to the CPOs. In Figure 3.1 and 3.2 are shown some of the functionalities available to the eMSP, such as the Map and the charging station availability. On the other hand, by figure 3.3 it's possible to see the list of the CPO's managed charging stations, as well as the eMSP association functionality. The complete list of mockups (mobile app for eMSP and application for CMPS) will be available in the design document.

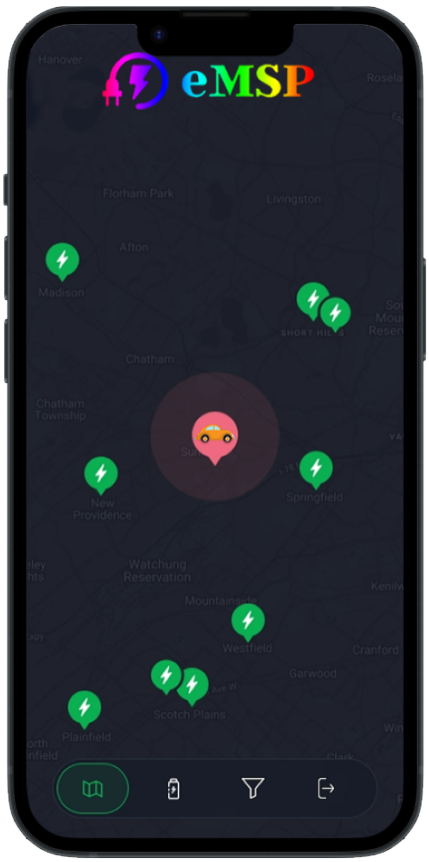

Figure 3.1: Map Mockup

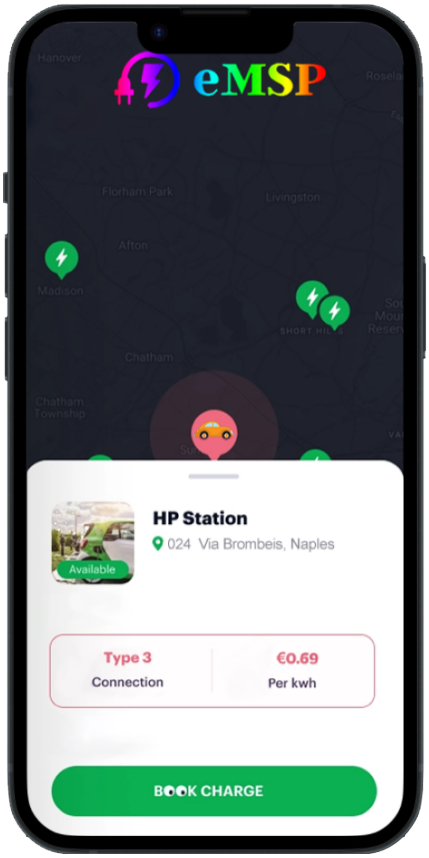

Figure 3.2: Booking Mockup

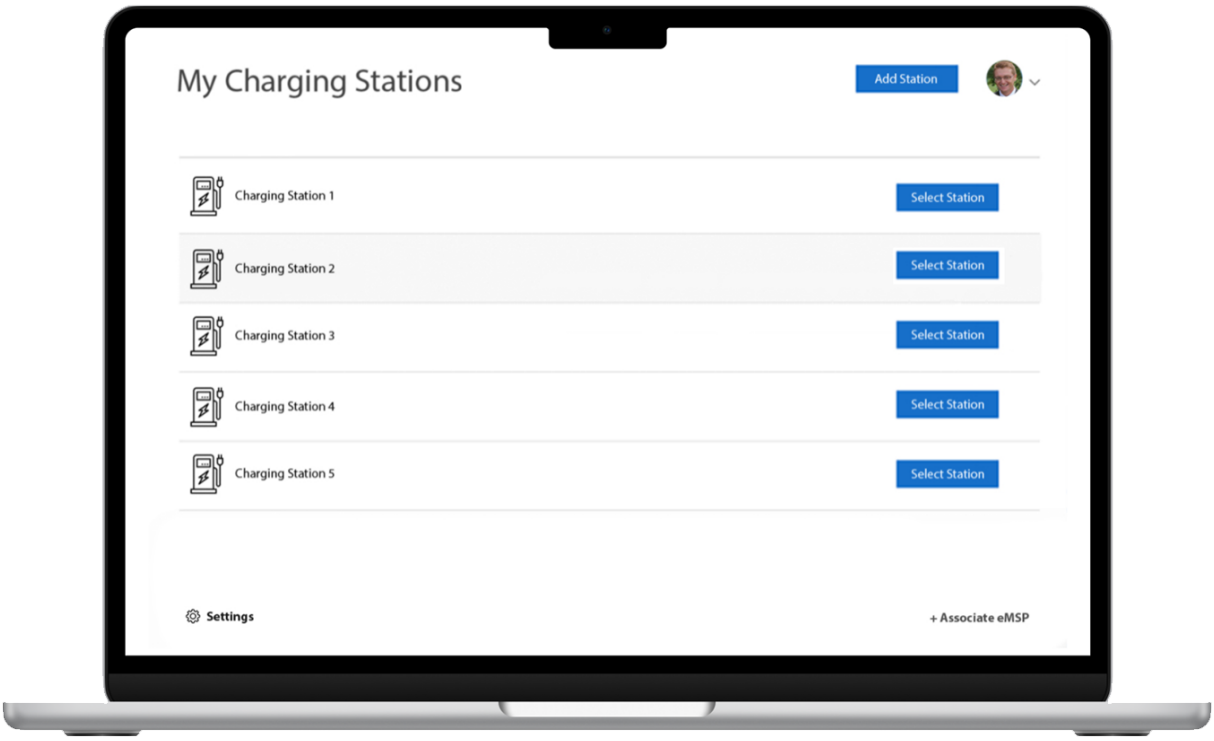

Figure 3.3: CPMS Mockup

3.1.2. Hardware Interfaces

- **Vehicle Interface**: Since the system requires to access info from the Driver's vehicle, such as the battery level, the location or the navigation route, it is necessary to establish a connection between the two of them. This should be done by the Driver through his device in order to let him receive suggestions.
- **Charging Station Interface**: In order to add and manage charging stations to the system there must be an interface between the two of them that permits the exchange of data through a standard protocol.
- **Charging Socket Interface**: In order to manage the charging process, check both its status and the connection between the vehicle and its connected socket, there must be an interface between the system and charging sockets that permits the exchange of data through a standard protocol.

3.1.3. Software Interfaces

The CPMS and eMSP communicate with each other through a uniform API, in order to exchange information. In particular, the CPMS has to update all the eMSPs that are associated with him about socket availability and stations' charging prices; meanwhile, the eMSP has to communicate with the CPMS in order to let the Driver perform operations for recharging his vehicle and create/delete bookings. In order to do so, all the CPMSs and the eMSPs should have the same external interface protocol and attain themselves to this standard.

3.1.4. Communication Interfaces

The systems, both the eMSP and the CPMS, communicate with external APIs in order to provide all the different functions described in section 2.2.

- **Payment API** - In order for the Driver to pay the relative CPO for the given charging service, the eMSP uses an external API that allows and manages payments.
- **Maps API** - In order to show to the Driver where all the charging stations are located, the eMSP uses external API that provides up-to-date maps.
- **eMall API** - The eMall system provides the following APIs in order to let the eMSP and the CPMS communicate with it:
  - **Confirming credentials** - The CPO credentials used to access the CPMS are authenticated on the eMall system.
  - **eMSP List** - The CPMS retrieves a list of all the existing eMSPs from the

eMall system, in order to show them to the CPO.

- **DSO API** - The CPMS uses an external uniform API to retrieve the DSOs prices and energy capacity, in order to choose which one to get energy from.

3.2. Functional Requirements

3.2.1. Use Case Diagrams

1. **Unregistered Driver**

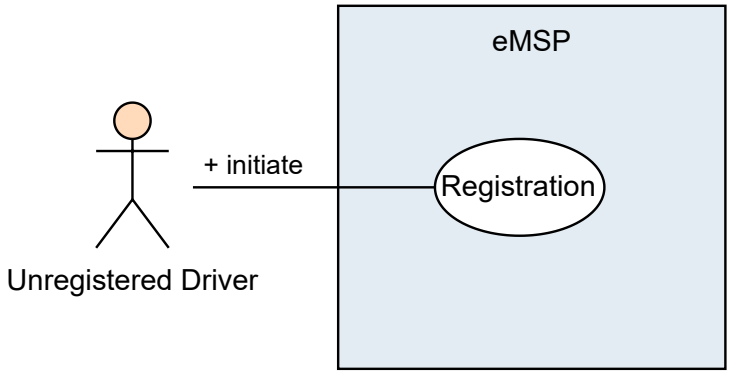

Figure 3.4: Use case diagram for an Unregistered Driver

2. **Driver**

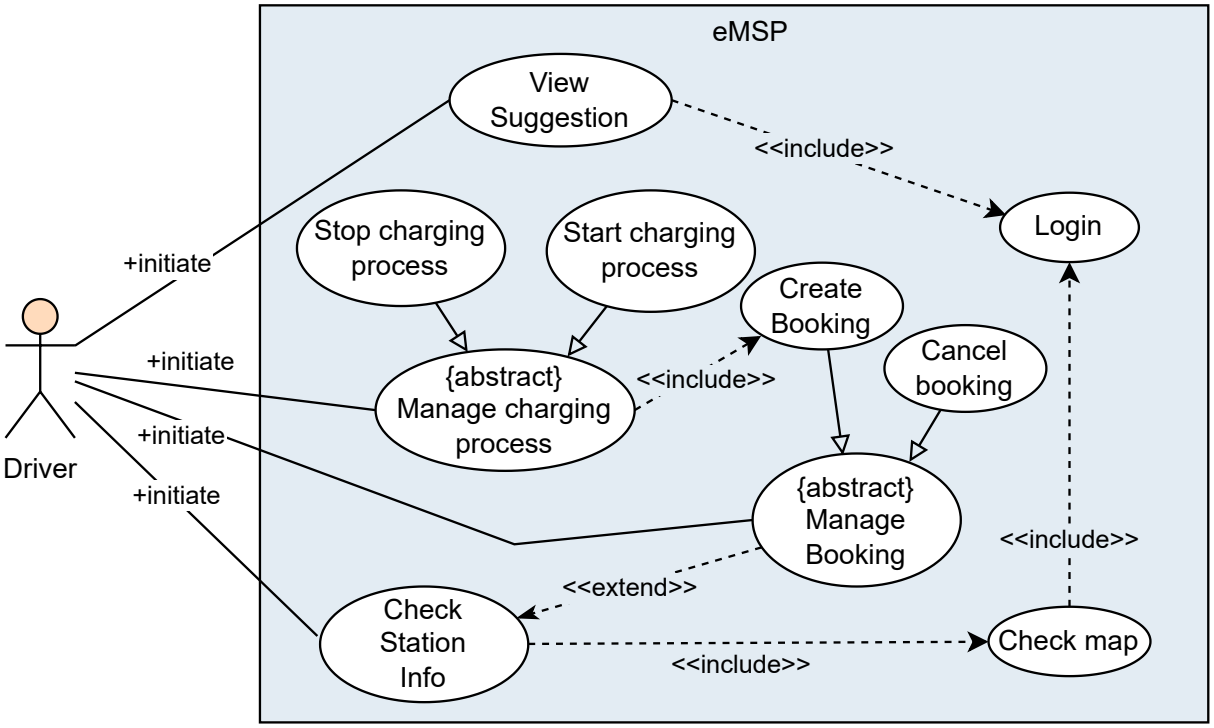

Figure 3.5: Use case diagram for a Driver

3. **CPO**

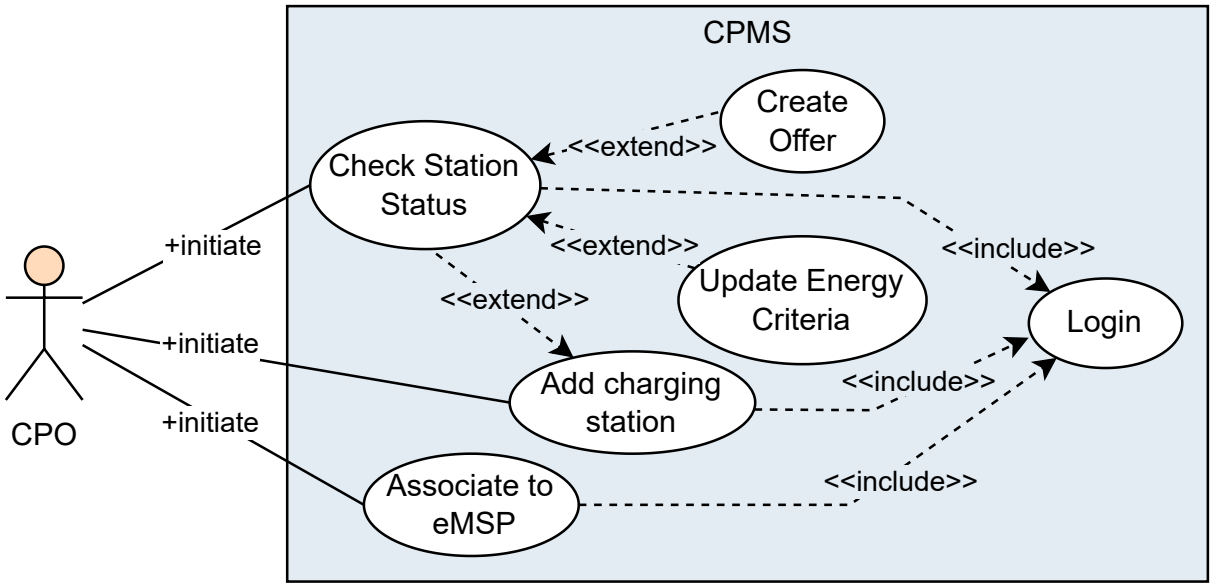

Figure 3.6: Use case diagram for a CPO

3.2.2. Use Cases

Note: In the following use cases we assume no internet connection failures.

1. **Driver Registration**

Use Case 1	**Driver Registration**
Actor(s)	Unregistered Driver
Entry Condition	The driver does not have an account and is on the initial view of the eMSP
Event Flow	(a) The Unregistered Driver press the "Register" button (b) The Unregistered Driver enters name, birth, date, email address and password (c) The Unregistered Driver submits the compiled form (d) The eMSP processes the request and shows a success message
Exit Condition	An account is created
Exception	The driver does not enter all mandatory data. The exception is notified to the Driver.

2. **Driver Login**

Use Case 2	**Driver Login**
Actor(s)	Registered Driver
Entry Condition	Driver is not logged in and on the eMSP main page
Event Flow	(a) The Driver press the "Login" button (b) The Driver submits his credentials such as email and password (c) The eMSP validates his credentials and shows a success message
Exit Condition	The driver is logged into the eMSP
Exception	Users enters invalid credentials. The exception is notified to the Driver.

3. **Driver checks map**

Use Case 3	**Driver checks map**
Actor(s)	Driver
Entry Condition	Driver is logged in
Event Flow	(a) The Driver opens the eMSP application and gets on the Map view (b) The Driver moves to area in which he wants to see all the charging stations (c) (opt) The Driver selects the charging type filter to apply to the map (d) The eMSP processes the request and returns the charging stations on the map that match the filters in that area
Exit Condition	All the stations are shown to the Driver
Exception	There are no available stations matching the filter in the requested area. The system sends a notification to warn the Driver.

4. **Driver checks charging station info**

Use Case 4	**Driver checks charging station info**
Actor(s)	Driver
Entry Condition	Driver is on the map page
Event Flow	(a) The Driver clicks on one of the shown charging stations (b) The eMSP processes the request and returns the info of that charging station, such as its charging prices for each charging type, the address and the availability (c) The Driver sees the received info
Exit Condition	Info of the stations are correctly shown to the Driver.
Exception	None

5. **Book a charge**

Use Case 5	**Book a charge**
Actor(s)	Driver
Entry Condition	Driver is checking the details of a station
Event Flow	(a) The Driver clicks on the "Book charge" button (b) The Driver selects the arrival time (c) The Driver clicks on "Submit" and sends the request (d) The eMSP notifies the user that he has successfully booked a charge and the ID of the charging socket
Exit Condition	The charge has been correctly booked
Exception	An exception is thrown to the eMSP by the CMPS. The exception message is notified to the Driver.

6. **Cancel a booked charge**

Use Case 6	**Cancel a booked charge**
Actor(s)	Driver
Entry Condition	Driver has booked a charge
Event Flow	(a) The Driver clicks on the Booked Charge button (b) The Driver clicks on the "Delete Booking" button (c) The eMSP notifies the user that the booked charge has been deleted
Exit Condition	The booked charge has been deleted
Exception	An exception is thrown to the eMSP by the CMPS. The exception message is notified to the Driver.

7. **Start charging process**

Use Case 7	**Start charging process**
Actor(s)	Driver
Entry Condition	The driver has booked a charge and is arrived at the charging station
Event Flow	(a) The Driver clicks on the Booked Charge button and checks the socket ID he has booked (b) The Driver plugs his vehicle to the socket with the ID booked (c) The Driver clicks on the "Start charge" button (d) The eMSP processes the request and notifies the user that the charging process has been started (e) The Driver sees the estimated time for the charge to end
Exit Condition	The charge is started
Exceptions	The charge can't be started because the vehicle is not well connected with the socket. The exception is notified to the Driver.

8. **Stop charging process**

Use Case 8	**Stop charging process**
Actor(s)	Driver
Entry Condition	The driver is checking the detail of his charging process
Event Flow	(a) The Driver clicks on the "Stop Charge" button (b) The eMSP processes the request and notifies the user that the charging process has been stopped (c) The eMSP notifies the user that the payment was successful
Exit Condition	The charge has been ended
Exception	The eMSP can't process the request of booking due to an error while communicating with the CMPS. The exception is notified to the Driver.

9. **Suggestion Notification**

Use Case 9	**Suggestion Notification**
Actor(s)	Driver
Entry Condition	The Driver has authorized the systems to read his data (calendar, vehicle's navigation routes, vehicle's location and vehicle's battery status) and receives a notification containing a suggested charging station where to charge his vehicle.
Event Flow	(opt - Driver can ignore the notification) (a) The Driver clicks on the notification. (b) The Driver is redirected to the station's info page.
Exit Conditions	• The Driver is on the station info page. • The Driver ignores the notification.
Exception	None

10. **CPO Login**

Use Case 10	**CPO Login**
Actor(s)	CPO
Entry Condition	The CPO is not logged in and is on the initial view of the CPMS
Event Flow	(a) The CPO presses the "Login" button. (b) The CPO submits his credentials such as email and password. (c) The CPMS validates his credentials and shows a success message.
Exit Condition	The CPO is logged into the CPMS and sees the list of his managed charging stations.
Exception	The CPO enters invalid credentials. The exception is notified to the CPO.

11. **View managed charging station status**

Use Case 11	**View managed charging station status**
Actor(s)	CPO
Entry Condition	The CPO is logged in
Event Flow	(a) The CPO clicks on one of the "Select Station" buttons of one of the listed charging stations. (b) The CPO visualizes info related to that station such as batteries energy (if available), the number of connected vehicles, their power absorption and their estimated remaining charging time.
Exit Condition	The station info is correctly visualized
Exception	None.

12. **Set an Offer**

Use Case 12	**Set an Offer**
Actor(s)	CPO
Entry Condition	The CPO is logged in and on a managed station page
Event Flow	(a) The CPO clicks on the "Set a new offer" button. (b) The CPO selects the amount of discount for each of the charging types wanted. (c) The CPO selects the expiration date for the offer. (d) The CPO clicks the "Submit" button. (e) The CPMS processes the information and shows a success message.
Exit Condition	The offer has been applied to the selected charging station
Exception	The CPO enters an invalid discount amount or an invalid expiration date. The exception is notified to the CPO.

13. **Update energy criteria**

Use Case 13	**Update energy criteria**
Actor(s)	CPO
Entry Condition	The CPO is logged in and on a managed station page
Event Flow	(a) The CPO clicks on the "Update energy criteria" button. (b) (opt) The CPO modifies the energy sale percentage revenue. (c) (opt) The CPO modifies the energy acquisition criteria. (d) The CPO clicks the "Update" button. (e) The CPMS processes the information and shows a success message.
Exit Condition	The energy criteria have been correctly updated for the selected charging station.
Exception	The new criteria are invalid. The exception is notified to the CPO.

14. **Associate eMSP**

Use Case 14	**Associate eMSP**
Actor(s)	CPO
Entry Condition	The CPO is logged in
Event Flow	(a) The CPO clicks on the "Associate eMSPs" button (b) The CPMS shows a list of all the existing eMSPs that are not associated yet (c) The CPO selects the eMSPs he wants to associate with (d) The CPO clicks on the "Submit Selection". (e) The CPMS processes the request and sends a success message
Exit Condition	The selected eMSPs are now associated with the CPMS.
Exception	The CPMS can't connect with one of the selected eMSP. The exception is notified to the CPO.

15. **Add charging station**

Use Case 15	**Add charging station**
Actor(s)	CPO
Entry Condition	The CPO is logged in and has a charging station not yet connected to the system.
Event Flow	(a) The CPO clicks on the "Add charging station" button. (b) The CPMS shows a list of detected charging stations. (c) The CPO connects the new charging station to one of the available stations from the list. (d) The CPO inserts the new charging station information e.g. name, location and number of sockets. (e) The CPO clicks the "Submit" button. (f) The CPMS processes the information and shows a success message. (g) The CPMS shows the energy criteria page of the new charging station.
Exit Condition	The charging station has been correctly added and the system is showing the energy criteria page of that station.
Exception	The new station information is invalid. The exception is notified to the CPO.

3.2.3. Sequence Diagrams

Note: In the following sequence diagrams we assume no internet connection failures. Synchronous responses are a result of actions performed locally in the application.

1. **Driver Registration**

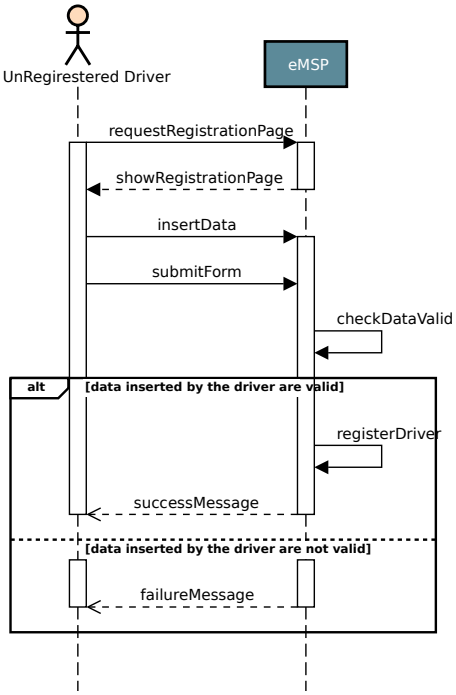

Figure 3.7: Registration of a driver

2. **Driver Login**

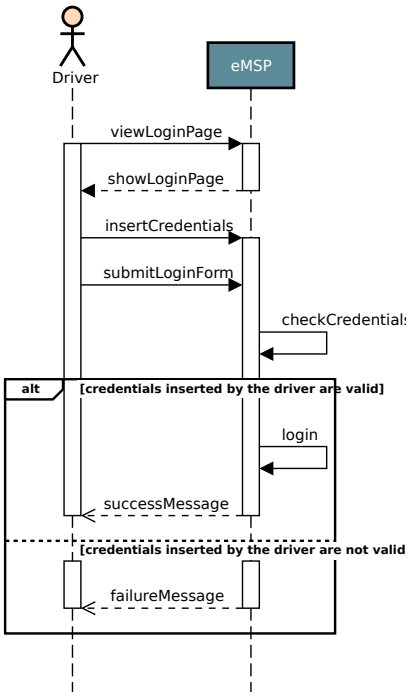

Figure 3.8: Login of a Driver

3. **Check Map**

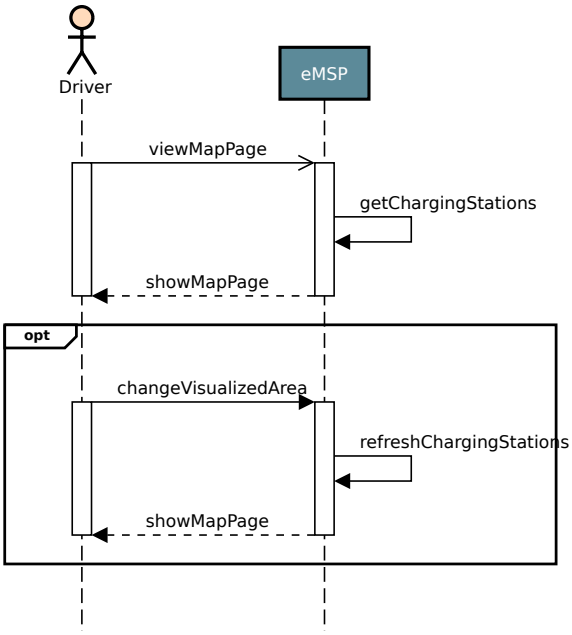

Figure 3.9: Check Map of the eMSP

4. **Check Station Info**

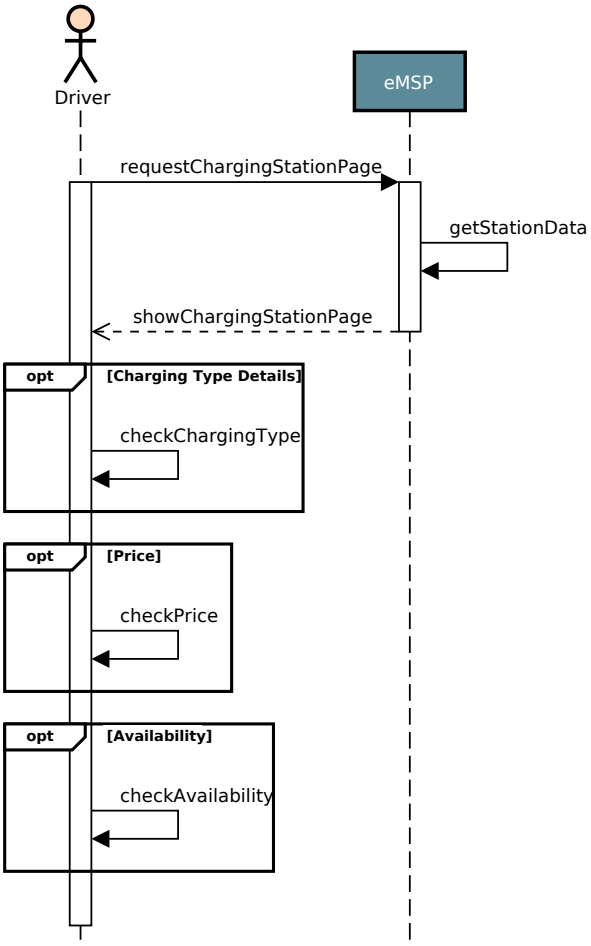

Figure 3.10: Check Station Info

5. **Booking**

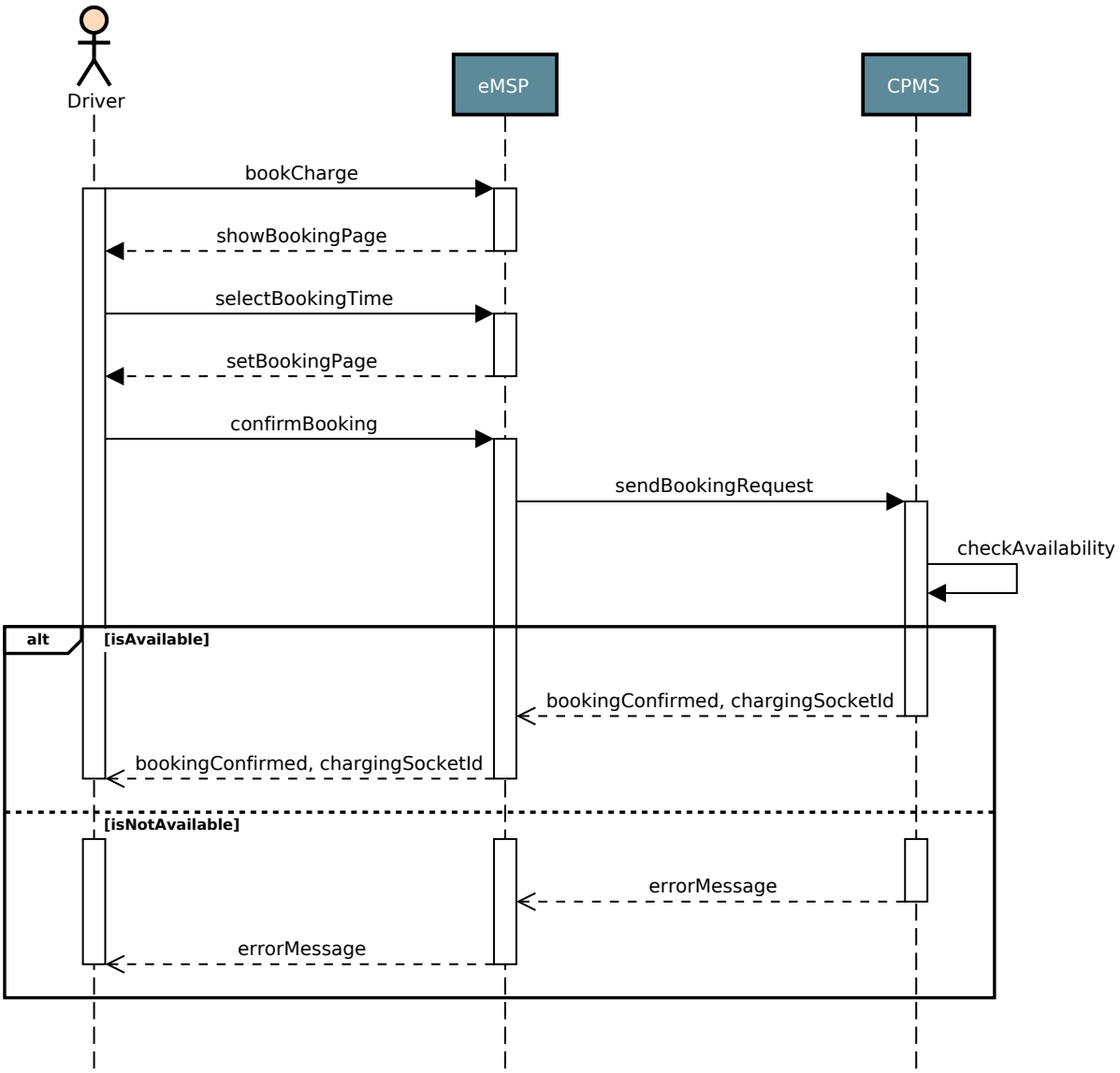

Figure 3.11: Book a charging process

6. **Delete Booking**

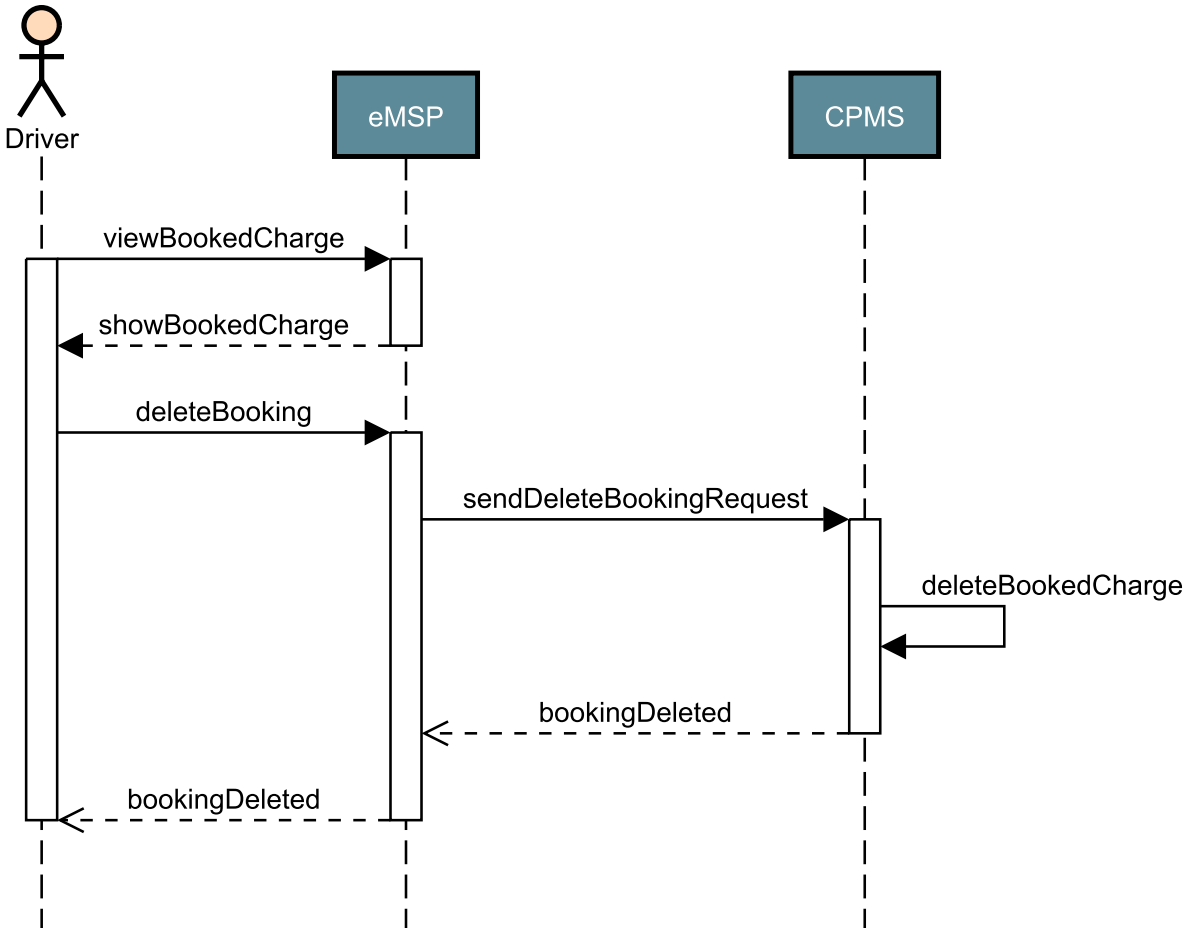

Figure 3.12: Delete a booked charging process

7. **Suggestion Notification**

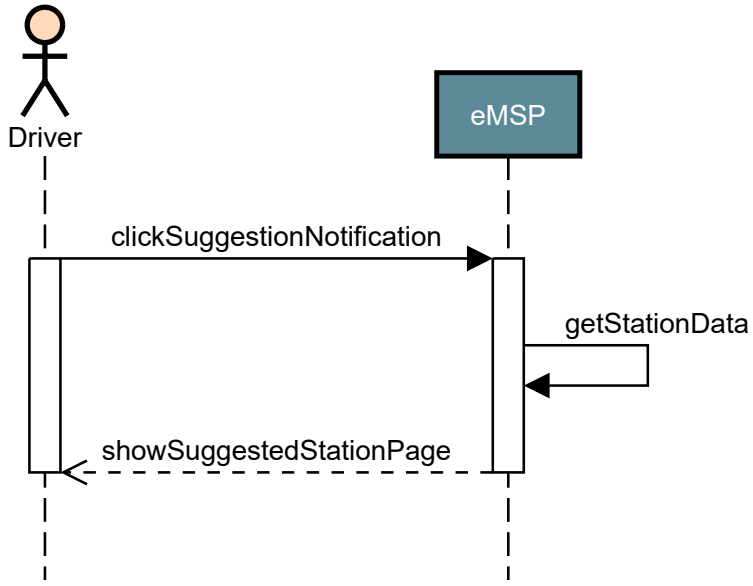

Figure 3.13: Driver receives and opens a suggestion

8. Charging Process

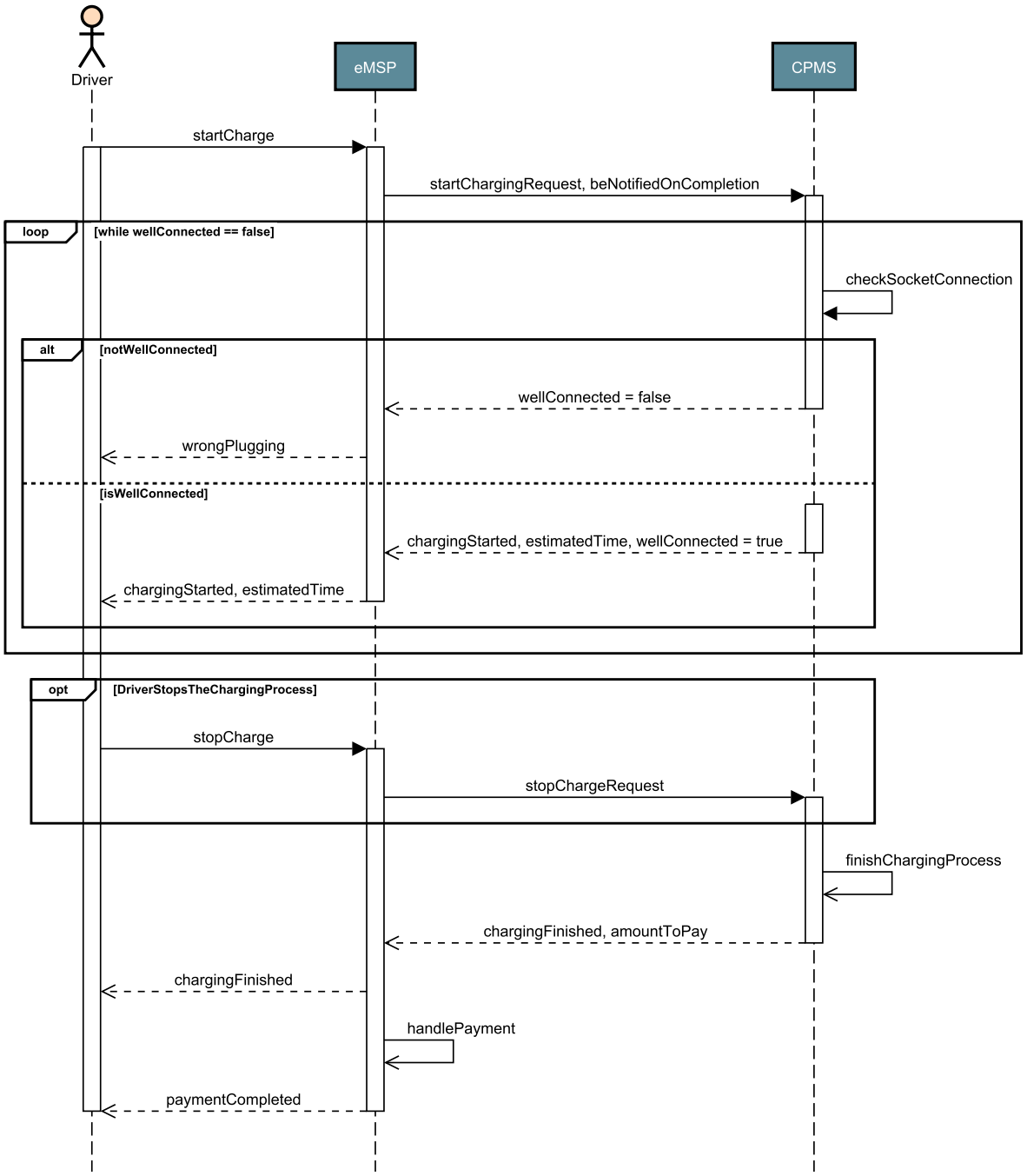

Figure 3.14: Process of a charging process

9. **CPO Login**

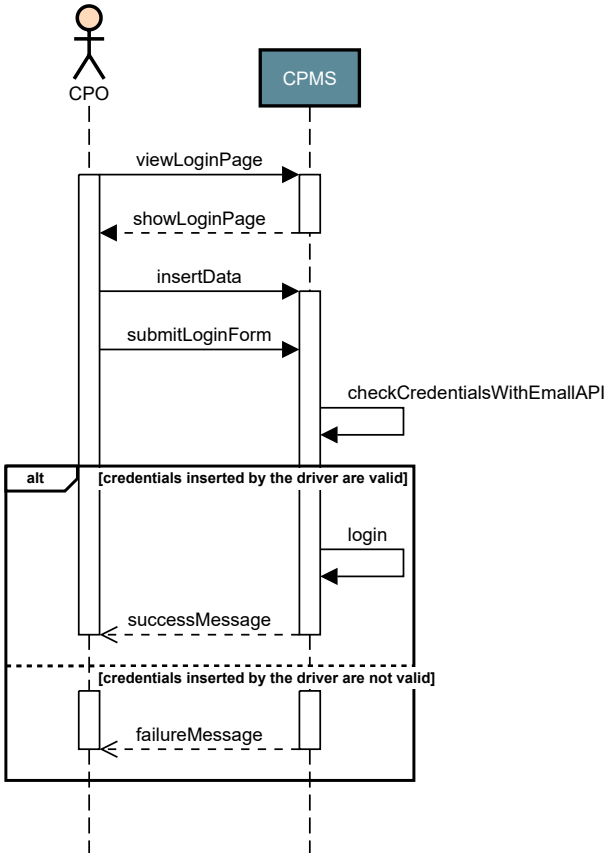

Figure 3.15: Process of a charging process

10. **View Managed Station Status**

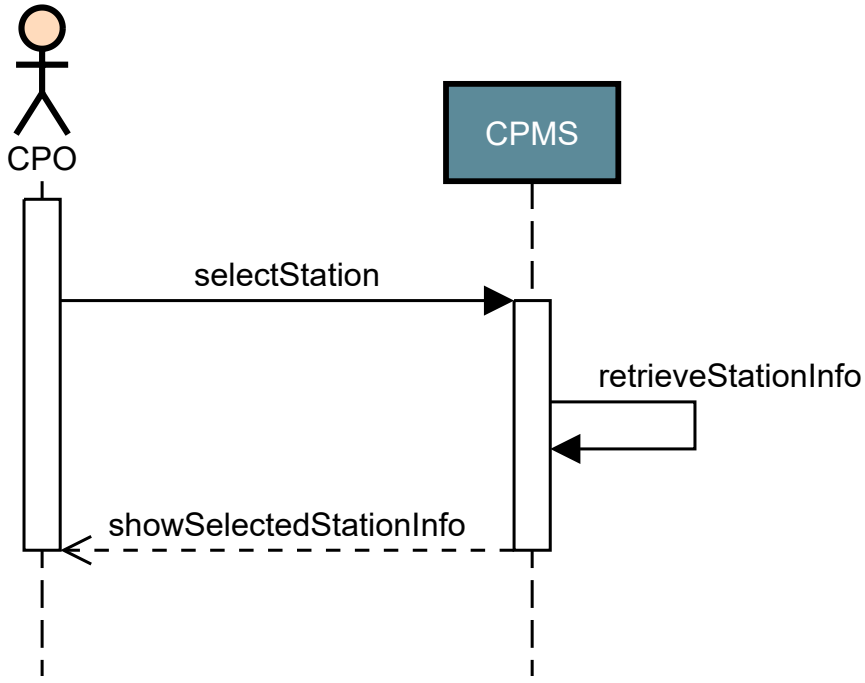

Figure 3.16: CPO views a managed station

11. **Add Station To CPMS**

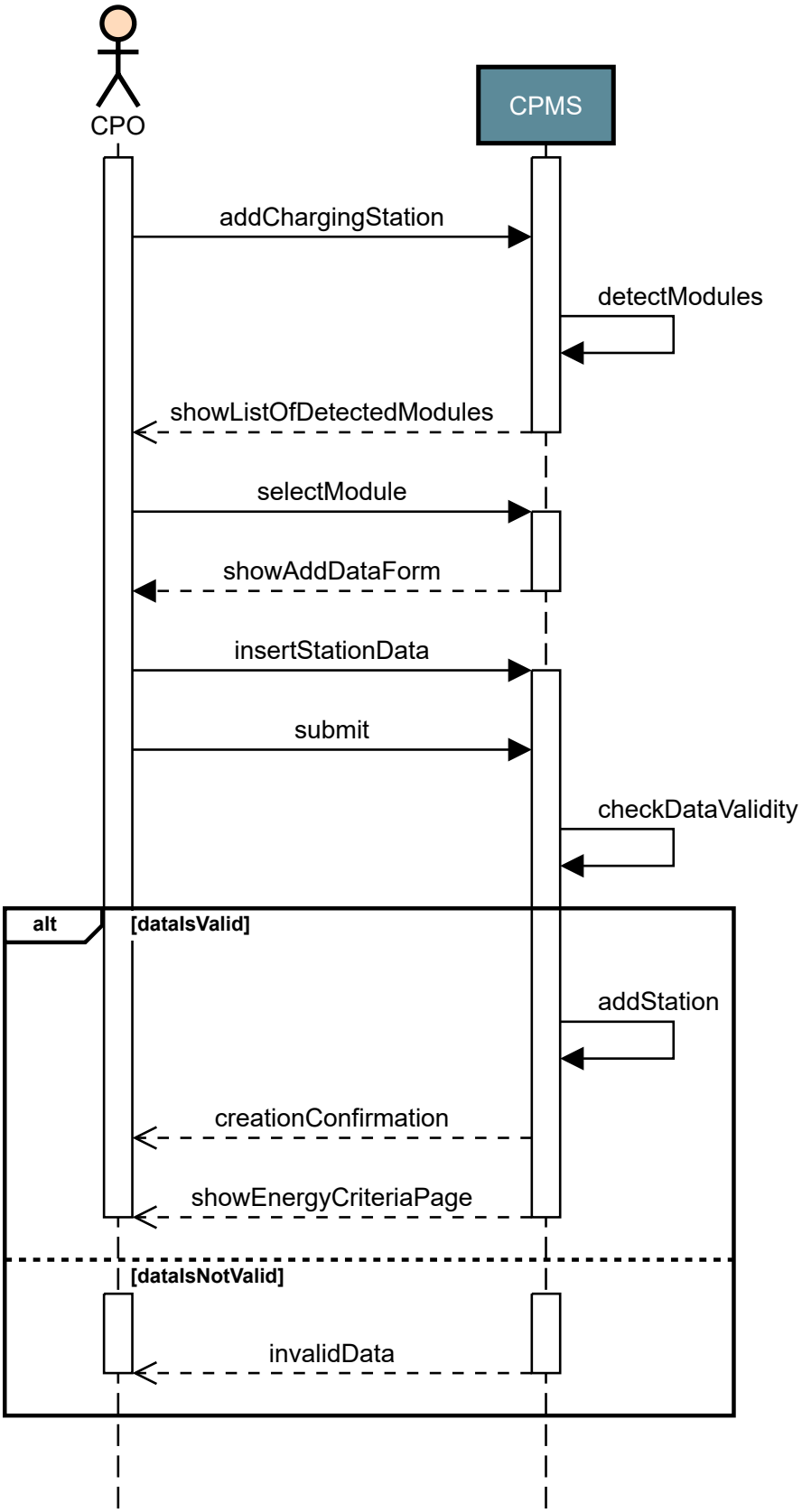

Figure 3.17: CPO adds a station

12. **Set Offer**

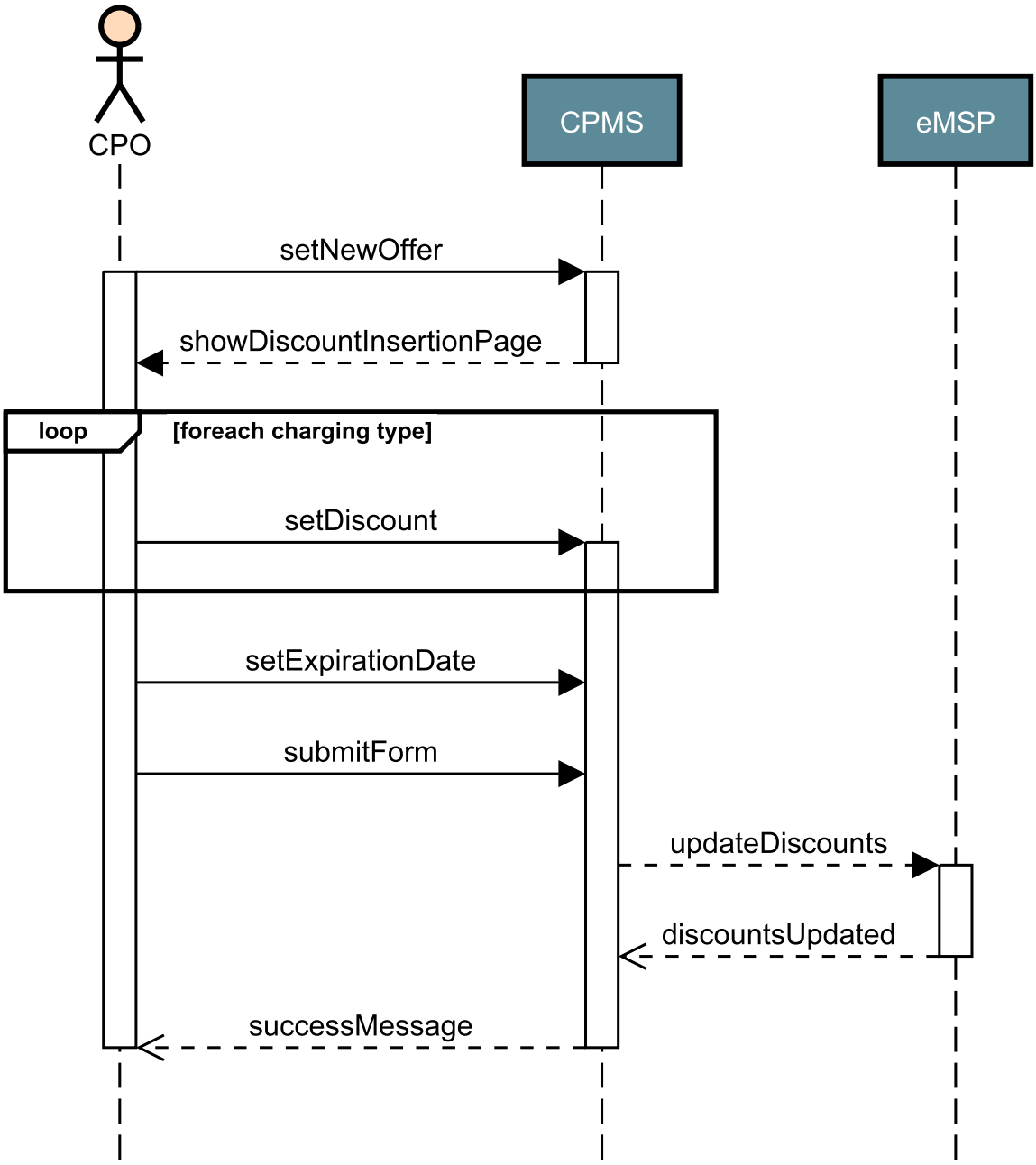

Figure 3.18: CPO sets a new offer

13. **Update Energy Criteria**

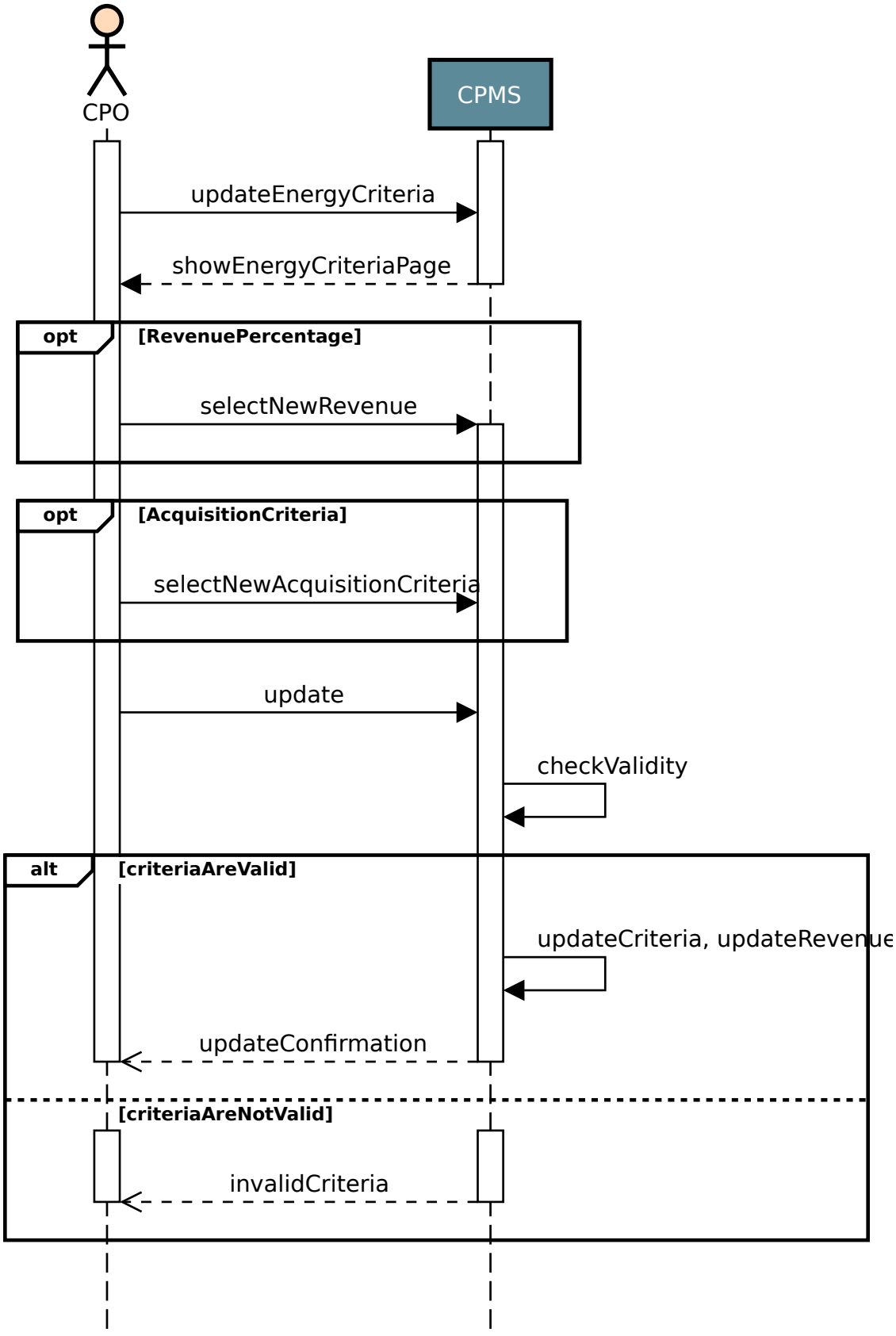

Figure 3.19: CPO updates energy criteria

14. **eMSP Association**

Note: In *sendCPOData* is also passed a payment method where the associated eMSP will send future drivers' payments.

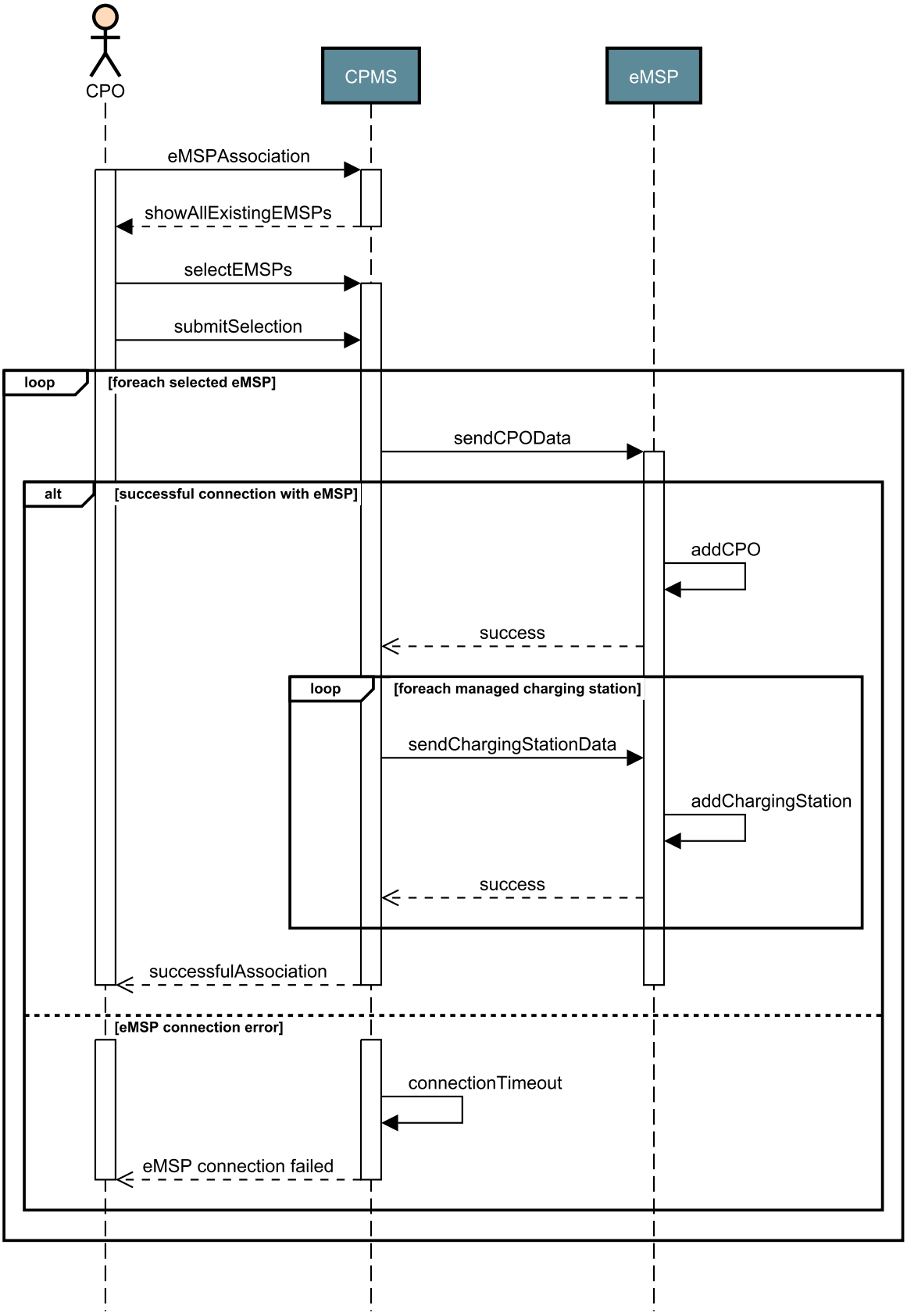

Figure 3.20: CPO associates a new eMSP

3.2.4. eMSP Requirements

In this section, the requirements for the two subsystems will be described as also their interactions. For both the Drivers and the CPOs it is required to be logged in to the system in order to perform any action different from

ID	Name	Description
FRE1	Driver Registration	An Unregistered Driver shall be able to register himself on the eMSP application.
FRE2	Driver Login	The Driver shall be able to login to the eMSP with his credentials.
FRE3	Internet Connection	The system shall notify the Driver if there is no internet connection.
FRE4	View Charging Stations	The Driver shall be able to see the location of all the charging stations in a certain geographic area.
FRE5	Charging Type Selection	The Driver shall be able to select the charging types he is interested in while searching for a charging station in order to filter them.
FRE6	Charging Cost	The Driver shall be able to check the cost per kWh for the specified charging types in a specific charging station.
FRE7	Charging Station Availability	The Driver shall be able to see which charging stations are currently available for the specified charging types.
FRE8	Future Availability	The Driver shall be able to see an estimated time in which a specific charging station will become available.
FRE9	Charge Booking	The Driver shall be able to book a charging socket in an available charging station, from 0 up to a max specified amount of time before.
FRE10	Booked Charging Socket ID	The system shall show the Driver the booked socket ID number.
FRE11	Delete Booking	The Driver shall be able to delete a booked charging process before its starting time.
FRE12	Confirm Booking	The Driver shall receive a notification that confirms the booking went well.
FRE13	Start Charging Process	As soon as the booked time starts the Driver shall be able to start the recharging process.
FRE14	Stop Charging Process	During the recharging process the Driver shall be able to stop it in advance.

FRE15	Time remaining	The Driver shall be able to check the estimated remaining time until the end of the charging process.
FRE16	Charge Finished	The system shall notify the Driver when the charging process is complete.
FRE17	Payment	At the end of the charging process, the system shall handle the payment between the Driver and the CPO through an external API.
FRE18	Generic Error Notification	Every time the system fails to elaborate an operation related to a Driver, then he shall receive a notification containing the error details.
FRE19	Successful Payment Notification	The system shall notify the Driver when a successful payment occurs.
FRE20	Check Vehicle Connection	The system shall be able to check if there exists a connection between the Driver's vehicle and the system itself.
FRE21	Vehicle's Remaining Charge	If the vehicle is connected, the system shall be able to retrieve the battery level of the Driver's vehicle.
FRE22	Vehicle's Location	If the vehicle is connected, the system shall be able to retrieve the Driver's vehicle location.
FRE23	Data Integration	The Driver shall be able to authorize the system to access his personal data such as his calendar, navigation system and location.
FRE24	Calendar Data	The system shall be able to retrieve data from the Driver's personal calendar.
FRE25	Navigation System	The system shall retrieve information about the Driver's navigation system.
FRE26	Charging Suggestion Creation	If the vehicle is connected to the Driver's device, the eMSP shall create charging suggestion notifications based on the status of the vehicle's battery, the Driver's schedule (his calendar and navigation system), special offers made available at some charging stations and the availability of charging sockets at those identified charging stations.
FRE27	Suggestion Notification	The system shall advise the Driver with a notification when and where he should recharge his vehicle's battery.

FRE28	Suggestion Interaction	The Driver shall be able to click on the charging suggestion notification sent by the eMSP and that shall redirect to the suggested charging station's info page.

Table 3.1: Table of eMSP Requirements

3.2.5. CPMS Requirements

ID	Name	Description
FRE29	CPO Login	The CPO shall be able to login to the CPMS with his credentials.
FRE30	Internet Connection	The system shall notify the CPO if there is no internet connection.
FRE31	View Charging Stations	The CPO shall be able to see the list of only all his owned charging stations.
FRE32	Add Charging Station	The CPO shall be able to add a new charging station.
FRE33	Batteries Energy	The CPO shall be able to check the remaining energy stored in the batteries in one of his own charging stations.
FRE34	Number Of Vehicles	The CPO shall be able to check the number of vehicles being charged in one of his charging stations.
FRE35	Power Absorption	The CPO shall be able to check the amount of power absorbed from each vehicle being charged in one of his charging stations.
FRE36	Charging Time Left	The system shall be able to give an estimation about how much time is needed to end the charging process for a particular vehicle being charged in a charging station.
FRE37	Energy Price Information	The system shall be able to retrieve the current energy price and the capacity for each of the available DSOs.
FRE38	Battery Energy Provider	The system shall be able to use the batteries as energy providers with prices equal to the cost that was used to recharge them.
FRE39	Energy Acquisition Criteria	The CPO shall be able to select the criteria that will be used by the CPMS in order to acquire energy, such as Energy provider with the lowest price, Energy Provider with biggest energy capacity.

FRE40	Energy Revenue Criteria	The CPO shall be able to select the revenue percentage amount from the energy sale of each charging station.
FRE41	Energy Acquisition Decision	The system shall be able, for each charging station, to automatically decide from which energy provider to retrieve the energy, based on the criteria chosen by the CPO.
FRE42	Energy Sale Price	The system shall be able, for each charging station, to automatically calculate the selling price of the energy based on the revenue criteria and on the price of the energy provider.
FRE43	Payment Calculation	The system shall be able to calculate when a charging process ends, the amount requested to pay based on the energy consumed and its selling price.
FRE44	Internal Battery Management	The system shall be able to automatically decide whether to store or not energy in the internal batteries, if available, of one of the managed charging stations.
FRE45	Special Offers	The system shall allow the CPO to apply discounts for the energy prices of a managed charging station until a set expiration date.
FRE46	eMSP list	The CPO shall be able to see a list of all the existing eMSPs.
FRE47	eMSP association	The CPO shall be able to decide which eMSP he wants to associate with his CPMS in order to permit communication between the two systems.

Table 3.2: Table of CPMS Requirements

3.2.6. Interaction Requirement

Here there is the description of the requirements that are in common between the two subsystems. To avoid having a large number of requirements, the following ones should be read in this way:

- **Format 1**: The eMSP shall (be able to) "perform an operation" on a single CPMS:
  - The eMSP shall communicate with the CMPS requesting to "perform an operation".
  - The CPMS shall return to eMSP the result of the "performed operation".

- **Format 2**: CPMS updates "data":
  - The CPMS shall detect when some "data" are updated.
  - The CPMS shall send to every associated eMSP the new updated "data".
  - The eMSP shall store the "data" received by the CMPS and show it to the driver when he requests it.

ID	Name	Description
FRE48	Book a charge	The eMSP shall be able to book a charge and receive the booked charging socket on a single CPMS.
FRE49	Delete a booked charge	The eMSP shall be able to delete a previously booked charge on a single CPMS.
FRE50	Start charge	The eMSP shall be able to start a charge on a single CPMS.
FRE51	Wrong plugging	The eMSP shall be notified when the vehicle is not well connected to the charging socket on a single CPMS.
FRE52	Stop charge	The eMSP shall be able to stop a charge on a single CPMS.
FRE53	Amount to pay	The eMSP shall be notified with the required amount of money to pay when the charging process ends on a single CPMS .
FRE54	Remaining Charge Time	The eMSP shall be able to ask for the estimated ending time for a specific charging process.
FRE55	Charging station availability	If CPMS can calculate the estimated time before each charging type becomes available, CPMS updates the availability of a charging station with the estimated time otherwise, CPMS updates only the availability.
FRE56	Charging station price	CPMS updates the price of a charging type in a station.
FRE57	Charging station offer	CPMS updates the offer of a charging type in a station.
FRE58	Charging station creation	CPMS updates about the data of the newly created charging station such as location, prices, supported charging types.

Table 3.3: Table of Interaction Requirements

3.2.7. Requirements mapping on Goals

R/G	G1	G2	G3	G4	G5	G6	G7
FRE1	X	X	X	X			
FRE2	X	X	X	X			
FRE3	X	X	X	X			
FRE4	X	X					
FRE5	X	X					
FRE6	X						
FRE7	X	X					
FRE8	X	X					
FRE9		X					
FRE10		X	X				
FRE11		X					
FRE12		X					
FRE13			X				
FRE14			X				
FRE15			X				
FRE16			X				
FRE17				X			
FRE18	X	X	X	X	X		
FRE19				X			
FRE20					X		
FRE21					X		
FRE22	X				X		
FRE23					X		
FRE24					X		
FRE25					X		
FRE26					X		
FRE27					X		
FRE28					X		
FRE29						X	X
FRE30						X	X
FRE31						X	X

FRE32						X	X
FRE33							X
FRE34							X
FRE35							X
FRE36			X				X
FRE37						X	
FRE38						X	
FRE39						X	
FRE40				X		X	
FRE41						X	
FRE42				X		X	
FRE43				X			
FRE44						X	
FRE45	X				X	X	
FRE46	X	X	X	X	X		
FRE47	X	X	X	X	X		
FRE48		X					
FRE49		X					
FRE50			X				X
FRE51			X				
FRE52			X				X
FRE53			X	X			
FRE54			X				X
FRE55	X	X			X		
FRE56	X				X		
FRE57	X				X		
FRE58	X	X			X		

Table 3.4: Mapping of Requirements on Goals

3.2.8. Domain Assumption mapping on Goals

R/G	G1	G2	G3	G4	G5	G6	G7
D1			X				
D2	X	X	X	X	X		
D3	X		X				
D4		X	X				
D5	X	X	X				
D6			X				
D7			X				
D8		X	X				
D9					X		
D10						X	X
D11						X	
D12				X			
D13	X						
D14	X	X	X	X	X		
D15						X	X

3.2.9. Explicit Goal Mapping

G1	**Allow Drivers to check nearby charging stations and see info about their prices, special offers and availability.**
FRE1	An Unregistered Driver shall be able to register himself on the eMSP application.
FRE2	The Driver shall be able to login to the eMSP with his credentials.
FRE3	The system shall notify the Driver if there is no internet connection.
FRE4	The Driver shall be able to see the location of all the charging stations in a certain geographic area.
FRE5	The Driver shall be able to select the charging types he is interested in while searching for a charging station in order to filter them.
FRE6	The Driver shall be able to check the cost per kWh for the specified charging types in a specific charging station.

FRE7	The Driver shall be able to see which charging stations are currently available for the specified charging types.
FRE8	The Driver shall be able to see an estimated time in which a specific charging station will become available.
FRE18	Every time the system fails to elaborate an operation related to a Driver, then he shall receive a notification containing the error details.
FRE22	If the vehicle is connected, the system shall be able to retrieve the Driver's vehicle location
FRE45	The system shall allow the CPO to apply discounts for the energy prices of a managed charging station until a set expiration date.
FRE46	The CPO shall be able to see a list of all the existing eMSPs.
FRE47	The CPO shall be able to decide which eMSP he wants to associate with his CPMS in order to permit communication between the two systems
FRE55	If CPMS can calculate the estimated time before each charging type becomes available, CPMS updates the availability of a charging station with the estimated time otherwise, CPMS updates only the availability.
FRE56	CPMS updates the price of a charging type in a station.
FRE57	CPMS updates the offer of a charging type in a station
FRE58	CPMS update about the data of the newly created charging station such as location, prices, and supported charging types.
D2	The Driver needs to know his personal data before signing up.
D3	Every time the driver books a charging process then he will show up in time at the charging station.
D5	Every time the recharging process ends the driver leaves the station with his vehicle, which he first disconnects from the socket.
D13	There exists an API endpoint where the eMSP can retrieve the map of a certain area.

D14	There exists an API endpoint on eMall where the CPMS can retrieve a list of all eMSPs.

G2	**Allow Drivers to create and delete bookings for charging their vehicle in a charging station.**
FRE1	An Unregistered Driver shall be able to register himself on the eMSP application.
FRE2	The Driver shall be able to login to the eMSP with his credentials.
FRE3	The system shall notify the Driver if there is no internet connection.
FRE4	The Driver shall be able to see the location of all the charging stations in a certain geographic area.
FRE5	The Driver shall be able to select the charging types he is interested in while searching for a charging station in order to filter them.
FRE7	The Driver shall be able to see which charging stations are currently available for the specified charging types.
FRE8	The Driver shall be able to see an estimated time in which a specific charging station will become available.
FRE9	The Driver shall be able to book a charging socket in an available charging station, from 0 up to a max specified amount of time before.
FRE10	The system shall show the Driver the booked socket ID number.
FRE11	The Driver shall be able to delete a booked charging process before its starting time.
FRE12	The Driver shall receive a notification that confirms the booking went well.
FRE18	Every time the system fails to elaborate an operation related to a Driver, then he shall receive a notification containing the error details.
FRE46	The CPO shall be able to see a list of all the existing eMSPs.

FRE47	The CPO shall be able to decide which eMSP he wants to associate with his CPMS in order to permit communication between the two systems
FRE48	The eMSP shall be able to book a charge and receive the booked charging socket on a single CPMS
FRE49	The eMSP shall be able to delete a previously booked charge on a single CPMS
FRE55	If CPMS can calculate the estimated time before each charging type becomes available, CPMS updates the availability of a charging station with the estimated time, otherwise CPMS updates only the availability.
FRE58	CPMS updates about the data of the newly createdc harging station such as location, prices, supported charging type.
D2	The Driver needs to know his personal data before signing up..
D4	When the Driver shows up during the time slot he booked, he'll always find his booked charging socket available.
D5	Every time the recharging process ends the driver leaves the station with his vehicle, which he first disconnects from the socket
D8	Each charging socket has a unique ID relative to its charging station.
D14	There exists an API endpoint on eMall where the CPMS can retrieve a list of all eMSPs.

G3	**Allow Drivers to manage and monitor their charging process.**
FRE1	An Unregistered Driver shall be able to register himself on the eMSP application.
FRE2	The Driver shall be able to login to the eMSP with his credentials.
FRE3	The system shall notify the Driver if there is no internet connection.
FRE10	The system shall show the Driver the booked socket ID number

FRE13	As soon as the booked time starts the Driver shall be able to start the recharging process.
FRE14	During the recharging process the Driver shall be able to stop it in advance.
FRE15	The Driver shall be able to check the estimated remaining time until the end of the charging process.
FRE16	The system shall notify the Driver when the charging process is complete.
FRE18	Every time the system fails to elaborate an operation related to a Driver, then he shall receive a notification containing the error details.
FRE36	The system shall be able to give an estimation about how much time is needed to end the charging process for a particular vehicle being charged in a charging station.
FRE46	The CPO shall be able to see a list of all the existing eMSPs.
FRE47	The CPO shall be able to decide which eMSP he wants to associate with his CPMS in order to permit communication between the two systems
FRE50	The eMSP shall be able to start a charge on a singleCPMS.
FRE51	The eMSP shall be notified when the vehicle is not well connected to the charging socket on a single CPMS
FRE52	The eMSP shall be able to stop a charge on a single CPMS.
FRE53	The eMSP shall be notified with the required amount of money to pay when the charging process ends on a single CPMS .
FRE54	The eMSP shall be able to ask for the estimated ending time for a specific charging process.
D1	The Driver's vehicle is electric and has a battery.
D2	The Driver needs to know his personal data before signing up.

D3	Every time the driver books a charging process then he will show up in time at the charging station.
D4	When the Driver shows up during the time slot he booked, he'll always find his booked charging socket available.
D5	Every time the recharging process ends the driver leaves the station with his vehicle, which he first disconnects from the socket.
D6	If a vehicle is connected to a charging socket, then it delivers energy only after a driver starts the charging process booked for that socket.
D7	The energy deployed by the charging socket is only used to recharge the vehicle batteries.
D8	Each charging socket has a unique ID relative to its charging station.
D14	There exists an API endpoint on eMall where the CPMS can retrieve a list of all eMSPs.

G4	**Allow Drivers to pay CPOs for the charging process provided.**
FRE1	An Unregistered Driver shall be able to register himself on the eMSP application.
FRE2	The Driver shall be able to login to the eMSP with his credentials.
FRE3	The system shall notify the Driver if there is no internet connection.
FRE17	At the end of the charging process, the system shall handle the payment between the Driver and the CPO through an external API.
FRE18	Every time the system fails to elaborate an operation related to a Driver, then he shall receive a notification containing the error details.
FRE19	The system shall notify the Driver when a successful payment occurs.

FRE40	The CPO shall be able to select the revenue percentage amount from the energy sale of each charging station.
FRE42	The system shall be able, for each charging station, to automatically calculate the selling price of the energy based on the revenue criteria and on the price of the energy provider.
FRE46	The CPO shall be able to see a list of all the existing eMSPs.
FRE47	The CPO shall be able to decide which eMSP he wants to associate with his CPMS in order to permit communication between the two systems
FRE53	The eMSP shall be notified with the required amount of money to pay when the charging process ends on a single CPMS .
D2	The Driver needs to know his personal data before signing up.
D12	There exists an external API that handles payments.
D14	There exists an API endpoint on eMall where the CPMS can retrieve a list of all eMSPs.

G5	**Proactively suggest to the Drivers where to go to charge their vehicle.**
FRE18	Every time the system fails to elaborate an operation related to a Driver, then he shall receive a notification containing the error details.
FRE20	The system shall be able to check if there exists a connection between the Driver's vehicle and the system itself.
FRE21	If the vehicle is connected, the system shall be able to retrieve the battery level of the Driver's vehicle.
FRE22	If the vehicle is connected, the system shall be able to retrieve the Driver's vehicle location.
FRE23	The Driver shall be able to authorize the system to access his personal data such as his calendar, navigation system and location.

FRE24	The system shall be able to retrieve data from the Driver's personal calendar.
FRE25	The system shall retrieve information about the Driver's navigation system.
FRE26	If the vehicle is connected to the Driver's device, the eMSP shall create charging suggestion notifications based on the status of the vehicle's battery, the Driver's schedule (his calendar and navigation system), special offers made available at some charging stations and the availability of charging sockets at those identified charging stations.
FRE27	The system shall advise the Driver with a notification when and where he should recharge his vehicle's battery.
FRE28	The Driver shall be able to click on the charging suggestion notification sent by the eMSP and that shall redirect to the suggested charging station's info page.
FRE45	The system shall allow the CPO to apply discounts for the energy prices of a managed charging station until a set expiration date.
FRE46	The CPO shall be able to see a list of all the existing eMSPs.
FRE47	The CPO shall be able to decide which eMSP he wants to associate with his CPMS in order to permit communication between the two systems
FRE55	If CPMS can calculate the estimated time before each charging type becomes available, CPMS updates the availability of a charging station with the estimated time, otherwise CPMS updates only the availability.
FRE56	CPMS updates the price of a charging type in a station.
FRE57	CPMS updates the offer of a charging type in a station
FRE58	CPMS update about the data of the newly created charging station such as location, prices, supported charging types.
D2	The Driver needs to know his personal data before signing up.

D9	The driver is able to create a connection between his device and his vehicle that permits to retrieve from it reliable data about the navigation system, the vehicle's battery status and location.
D14	There exists an API endpoint on eMall where the CPMS can retrieve a list of all eMSPs.

G6	**Allow CPOs to manage the charging prices and the criteria of energy acquisition in their charging stations.**
FRE29	The CPO shall be able to login to the CPMS with his credentials.
FRE30	The system shall notify the CPO if there is no internet connection.
FRE31	The CPO shall be able to see the list of only all his owned charging stations.
FRE32	The CPO shall be able to add a new charging station.
FRE37	The system shall be able to retrieve the current energy price and the capacity for each of the available DSOs.
FRE38	The system shall be able to use the batteries as energy providers with prices equal to the cost that was used to recharge them.
FRE39	The CPO shall be able to select the criteria that will be used by the CPMS in order to acquire energy, such as Energy provider with the lowest price, Energy Provider with biggest energy capacity.
FRE40	The CPO shall be able to select the revenue percentage amount from the energy sale of each charging station.
FRE41	The system shall be able, for each charging station, to automatically decide from which energy provider to retrieve the energy from, based on the criteria chosen by the CPO.
FRE42	The system shall be able, for each charging station, to automatically calculate the selling price of the energy based on the revenue criteria and on the price of the energy provider.

FRE44	The system shall be able to automatically decide whether to store or not energy in the internal batteries, if available, of one of the managed charging stations.
FRE45	The system shall allow the CPO to apply discounts for the energy prices of a managed charging station until a set expiration date.
D10	There exists a standard communication protocol that permits the charging sockets and charging stations to communicate with the CPMS in order to notify it of events like vehicle connection and disconnection or data about the charging status (completed, in process, remaining charging time).
D11	There exists a uniform API that allows to retrieve information about DSOs, such as price and energy capacity.
D15	There exists an API endpoint on eMall where the CPMS can confirm the credentials inserted by the CPO

G7	**Allow CPOs to connect to all their charging stations and monitor them.**
FRE29	The CPO shall be able to login to the CPMS with his credentials.
FRE30	The system shall notify the CPO if there is no internet connection.
FRE31	The CPO shall be able to see the list of only all his owned charging stations.
FRE32	The CPO shall be able to add a new charging station.
FRE33	The CPO shall be able to check the remaining energy stored in the batteries in one of his own charging stations.
FRE34	The CPO shall be able to check the number of vehicles being charged in one of his charging stations.
FRE35	The CPO shall be able to check the amount of power absorbed from each vehicle being charged in one of his charging stations.
FRE36	The system shall be able to give an estimation about how much time is needed to end the charging process for a particular vehicle being charged in a charging station.
FRE50	The eMSP shall be able to start a charge on a single CPMS.
FRE52	The eMSP shall be able to stop a charge on a single CPMS.

FRE54	The eMSP shall be able to ask for the estimated ending time for a specific charging process.
D10	There exists a standard communication protocol that permits the charging sockets and charging stations to communicate with the CPMS in order to notify it of events like vehicle connection and disconnection or data about the charging status (completed, in process, remaining charging time).
D15	There exists an API endpoint on eMall where the CPMS can confirm the credentials inserted by the CPO

3.3. Performance Requirements

For the eMSP system, the performance requirements are the following:

- The system should be available 99% of the time.
- The response time for initiating/stopping a charging process should last a maximum of 4 seconds.
- In order to improve the overall user experience, all other operations not mentioned above should take no more than 15 seconds.

For the CPMS system, the performance requirements are the following:

- The system should be available 99% of the time.
- All operations should take no more than 15 seconds.

3.4. Design Constraints

3.4.1. Standards Compliance

- The system must receive from the driver permission to retrieve data regarding the position, battery status and navigation system of their vehicle.
- The system must manage the data retrieved from the user with respect to privacy laws.
- The system must keep data anonymous if they are used for external analysis or public disclosure.

3.4.2. Hardware Limitations

- Driver:
  - A mobile device with an internet connection and enough storage space to install and execute the eMSP application and capable of sending notifications to the user.
  - An electric vehicle that is able to connect to the driver's device and share its data: location, battery status and navigation system.
- CPO:
  - A computer device with an internet connection and enough storage space to install and execute the CPMS application.
  - Charging sockets with a universal plug in order to be able to connect to most electric vehicles.
- eMSP and CPMS Servers:

- Stable network connection to keep the server up.
- Enough processing power in order to handle all the requests made by the users.
- Enough storage space to install the application.

3.5. Software System Attributes

ID	Name	Description
NFRE1	Reliability	The systems should provide duplicates of the components and implement multiple storage systems, which implies a redundancy of data, with the aim to avoid accidental data losses.
NFRE2	Availability	There should be copies of the different components of the systems in order to have the possibility to keep the service up in case of maintenance operations or possible malfunctions.
NFRE3	Security	The systems shall encrypt its stored data to protect sensitive information, such as the users' payment methods or credentials, from potential attacks directed at the database. Moreover, to avoid sniffing and spoofing, the communication of important data throughout the application shall be done through the usage of some sort of encryption.
NFRE4	Maintainability	The systems should be developed using the best practices and modalities of *software engineering* in order to maintain them and have the possibility to expand their functionalities.
NFRE5	Portability	The eMSP application should be compatible with the most recent versions of iOS and Android for smartphones and tablets, regardless of screen size. The CPMS application should be supported on Windows and UNIX-like operating systems for computers.

Table 3.13: Non-Functional Requirements

4 | FORMAL ANALYSIS

In this chapter alloy analysis will be used in order to demonstrate the validity of the model, pointing out some peculiarities hard to notice given the previous UML representation.

4.1. Alloy Code

```
sig PaymentApi {}

sig Time {timeStamp : one  Int} {timeStamp ≥ 1}

abstract sig EnergyProvider{}

sig DSO extends EnergyProvider{}

------------------- DRIVER
sig Driver {
    drives: one ElectricVehicle,
    calendar: set CalendarEvent,
    uses: some eMSP,
    driverPayment: uses -> one PaymentApi,
    receives: set Suggestion,
    books: set Booking
}

sig Booking {
    station: one ChargingStation,
    socket: one ChargingSocket,
    time: one Time,
}{one this.~creates and one this.~books and socket in station.sockets}

sig ElectricVehicle {
    routes: set NavigationRoute
} { one this.~drives}

sig NavigationRoute {} { one this.~routes}

sig CalendarEvent {}{one this.~calendar}

------------------- CPO
sig CPO {
    utilizes: one CPMS,
    cpoPayment: eMSP -> lone PaymentApi
}
```

```
sig CPMS {
    manages: some ChargingStation,
    creates: set Booking,
    associates: set eMSP
}{one this.~utilizes}

sig ChargingStation {
    sockets: some ChargingSocket,
    price: ChargingType -> lone Price,
    energyProvider: some EnergyProvider
}{one this.~manages}

sig Battery extends EnergyProvider{}{one this.~energyProvider}

sig Price {}

sig ChargingSocket {
    type: one ChargingType
} {one this.~sockets}
// represents a charging type, such as slow, fast, rapid...
sig ChargingType {}

-------------------- eMSP
sig eMSP {
    supports: some PaymentApi,
    generates: set Suggestion,
    map: one Map,
}
sig Map {
    contains: set ChargingStation
}{one this.~map}

sig Suggestion {
    event: set CalendarEvent,
    navigation: set NavigationRoute,
    vehicle: one ElectricVehicle,
    suggestedStation: one ChargingStation
}{one this.~generates and one this.~receives}

-------------------- FACTS
// for each driver and eMSP there is only one payment method used for that eMSP and that
    ↪ payment method is supported
fact driverUsesOnlySupportedPayments{
    all d:Driver, e:eMSP | e in d.uses implies (one p:PaymentApi |
        p = e.(d.driverPayment)) and (e.(d.driverPayment) in e.supports)
}

//  for each CPO and eMSP there is only one payment method used for that eMSP and that
    ↪ payment method is supported
fact CPOUsesOnlySupportedPayments{
    all c:CPO, e:eMSP | (e in c.utilizes.associates iff (one p:PaymentApi |
    p = e.(c.cpoPayment))) and (e.(c.cpoPayment) in e.supports)
}

// suggestions are only based on data from the driver receiving that suggestion
fact useOnlyDataFromReceivingDriver{
    all s:Suggestion | (
        (all c:CalendarEvent | c in s.event iff c.~calendar = s.~receives) and
        (all n:NavigationRoute | n in s.navigation iff
```

```
        n.~routes.~drives=s.~receives) and
        (all v: ElectricVehicle | v = s.vehicle iff v.~drives= s.~receives)
        )
}

// an eMSP can suggest only a charging station present in its map
fact onlySuggestStationInMap{
    all s: Suggestion | s.suggestedStation in s.~generates.map.contains
}

// an eMSP can send suggestions only to drivers using that eMSP
fact onlySuggestionsForAssociatedDrivers{
    all s: Suggestion | s.~generates in s.~receives.uses
}

// if a price exists then it's associated to some charging station
fact everyPriceConnected{
    no p: Price | p not in ChargingType.(ChargingStation.price)
}

// if a charging type is supported by some charging station then there exists a single
    ↪ price set by that charging station for that charging type
fact ChargingTypeMatchesSocketType {
    all s: ChargingStation, t: ChargingType | (t in s.sockets.type iff (one p:
        ↪ Price | p = t.(s.price)))
}

// a Driver can do a booking only for a charging station in a map of one of his used
    ↪ eMSPs, associated with the CPMS of said station
fact onlyeMSPAssociatedBookings{
    all b: Booking, d: Driver | b in d.books implies
        (some e: eMSP | e in ((b.station).~contains).~map and e in d.uses and e in (b.~
            ↪ creates).associates)
}

// every booking made on the same socket must be on different times
fact DifferentBookingsDifferentTimes{
        all disj b1, b2: Booking | b1.socket = b2.socket implies b1.time ≠ b2.time
}

// every charging station appears in a map iff the CPMS managing that charging station is
    ↪  associated to that map's eMSP
fact ChargingStationsInEmspMap {
    all s: ChargingStation, m: Map | s in m.contains iff (m.~map in (s.~
        ↪ manages).associates)
}

------------------- ASSERTIONS
assert NoUnmatchedSocketAndCharging {
    no c: ChargingStation | some t: ChargingType | (t in c.sockets.type and no p:
        ↪ Price | p in t.(c.price)) or (t in (c.price).Price and
        no s: ChargingSocket | s in c.sockets and s.type = t)
}
--check NoUnmatchedSocketAndCharging

assert NoDriverWithUnsupportedPayment {
    no d: Driver | (some e: eMSP | e in d.uses and (no p: PaymentApi | p in e. (d.
        ↪ driverPayment) and p in e.supports) or
        (e.(d.driverPayment) not in e.supports))
```

```
}
--check NoDriverWithUnsupportedPayment

assert NoCPOWithUnsupportedPayment {
    no c: CPO | some e: eMSP | e in c.utilizes.associates and (no p: PaymentApi |     p in
        ↪ e.(c.cpoPayment) and p in e.supports)
        or (e.(c.cpoPayment) not in e.supports)
}
--check NoCPOWithUnsupportedPayment

assert NoBookingWithoutAssociationOrUsage{
    no b: Booking | no e: eMSP | (e in ((b.station).~contains).~map and e in         (b.~
        ↪ creates).associates) or e not in (b.~books).uses
}
--check NoBookingWithoutAssociationOrUsage

assert NoBookingsAtSameTime {
        no disj b1, b2: Booking | b1.socket = b2.socket and b1.time = b2.time
}
--check NoBookingsAtSameTime

assert NoSuggestionWithNotAssociatedChargingStations {
    no s: Suggestion|  s.~generates not in (s.suggestedStation).~manages.associates
}
--check NoSuggestionWithNotAssociatedChargingStations

------------------- DYNAMIC MODELLING
// add a booking
pred addBooking[d: Driver, d': Driver, b: Booking]{
    d'.drives = d.drives
    d'.calendar = d.calendar
    d'.uses =  d.uses
    d'.driverPayment = d.driverPayment
    d'.receives = d.receives
    d'.books = d.books + b
}
-- run addBooking for 5

// receive a suggestion
pred receiveSuggestion[d: Driver, d': Driver, s: Suggestion]{
    d'.drives = d.drives
    d'.calendar = d.calendar
    d'.uses =  d.uses
    d'.driverPayment = d.driverPayment
    d'.receives = d.receives + s
    d'.books = d.books
}

-- run receiveSuggestion for 5

// add a charging station
pred addChargingStation[c: CPMS, c': CPMS, s: ChargingStation]{
    c'.manages = c.manages + s
    c'.creates = c.creates
    c'.associates = c.associates
}
-- run addChargingStation for 5
// associate CPMS to new eMSP
pred associateToEeMSP[c, c': CPMS, e: eMSP]{
```

```
    c'.associates = c.associates + e
    c'.manages = c.manages
    c'.creates = c.creates
}
-- run associateToEeMSP for 5

------------------- STATIC MODELLING
pred world1 {
    # ChargingStation = 3
    # ChargingSocket = 4
    # Driver = 3
    # Price = 3
    # uses  = 4
    # ChargingType = 3
    # PaymentApi = 3
    # associates = 3
    # eMSP = 3
    # CPO = 2
    # CPMS = 2
    # DSO = 2
    # Battery = 1
    # Booking = 0
    # NavigationRoute = 0
    # Suggestion = 0
    # CalendarEvent = 0
    # Time = 0
}
-- run world1 for 5
pred world2 {
    # ChargingStation = 3
    # ChargingSocket = 4
    # Driver = 2
    # Price = 2
    # uses  = 2
    # ChargingType = 3
    # PaymentApi = 1
    # associates = 2
    # eMSP = 1
    # CPO = 2
    # CPMS = 2
    # DSO = 1
    # Booking = 2
    # Battery = 0
    # NavigationRoute = 2
    # Suggestion = 2
    # CalendarEvent = 2
    # Time = 2
}

run world2 for 5
```

4.1.1. Results

No assertion gives a counterexample instance, hence the model should be well-posed.

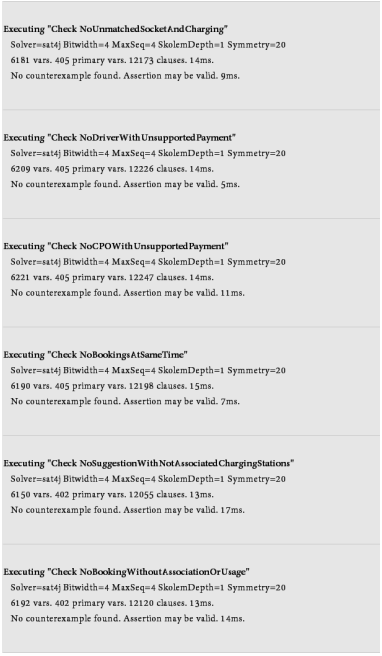

Figure 4.1: Assertions Results

4.2. Models

4.2.1. First Model

In figure 4.2 the most important relations of the model are represented. In particular, it's possible to notice that the map of an eMSP shows only the charging stations managed by a CPMS associated with that eMSP.
Another interesting relation shows that both Drivers and CPOs have to use a payment method (PaymentAPI) supported by the eMSP to which they are directly (for Drivers) or indirectly (for CPOs, through their CPMS) linked.

4.2.2. Second Model

In figure 4.3 it is possible to see how the bookings and suggestions work. In particular, Drivers can create a booking, through a used eMSP, on a station's socket only if that station is managed by a CPMS associated with said eMSP. Also, the suggestions can be received by Drivers based on their vehicle, navigation routes and calendar events. Similarly to Bookings, they can be suggested a charging station only if the eMSP that generates the suggestion is associated with the CPMS that manages said station.

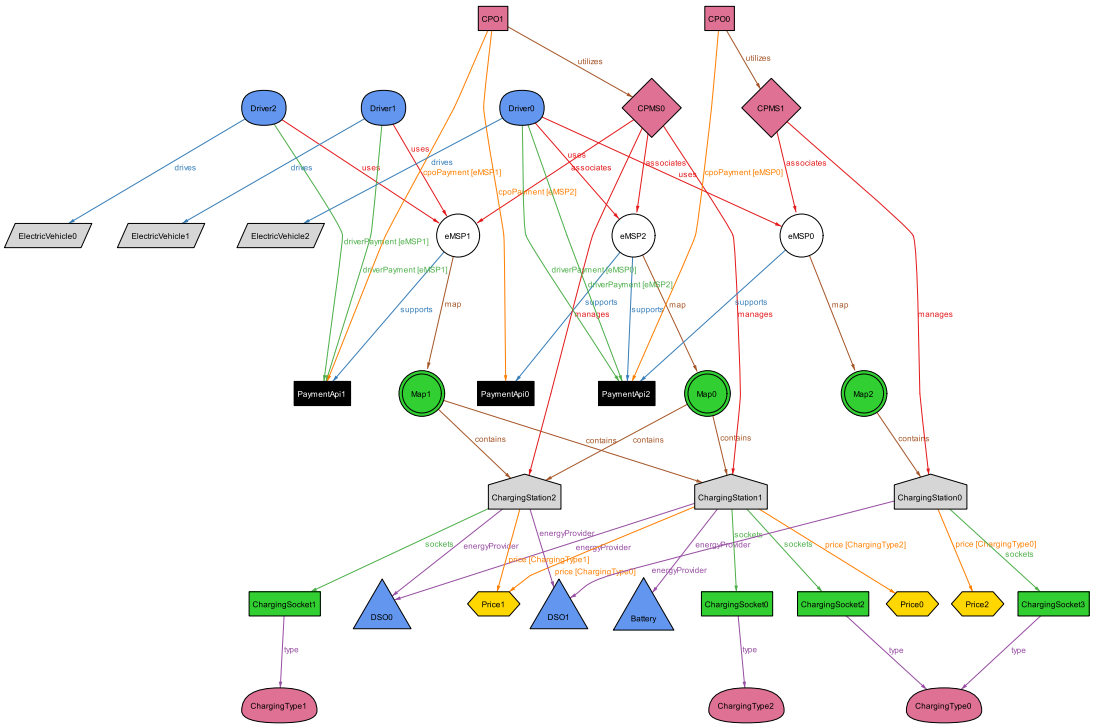

Figure 4.2: World 1

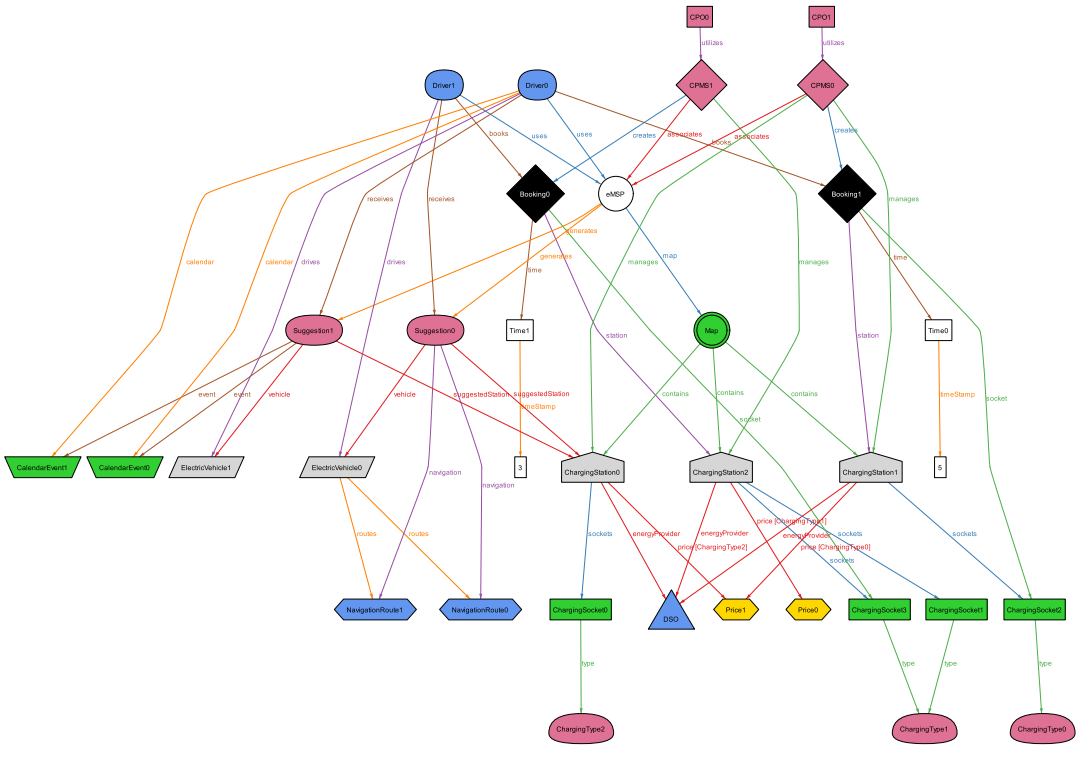

Figure 4.3: World 2

4.2.3. Dynamic Model

Here below there is an example instance found running the *addBooking* predicate

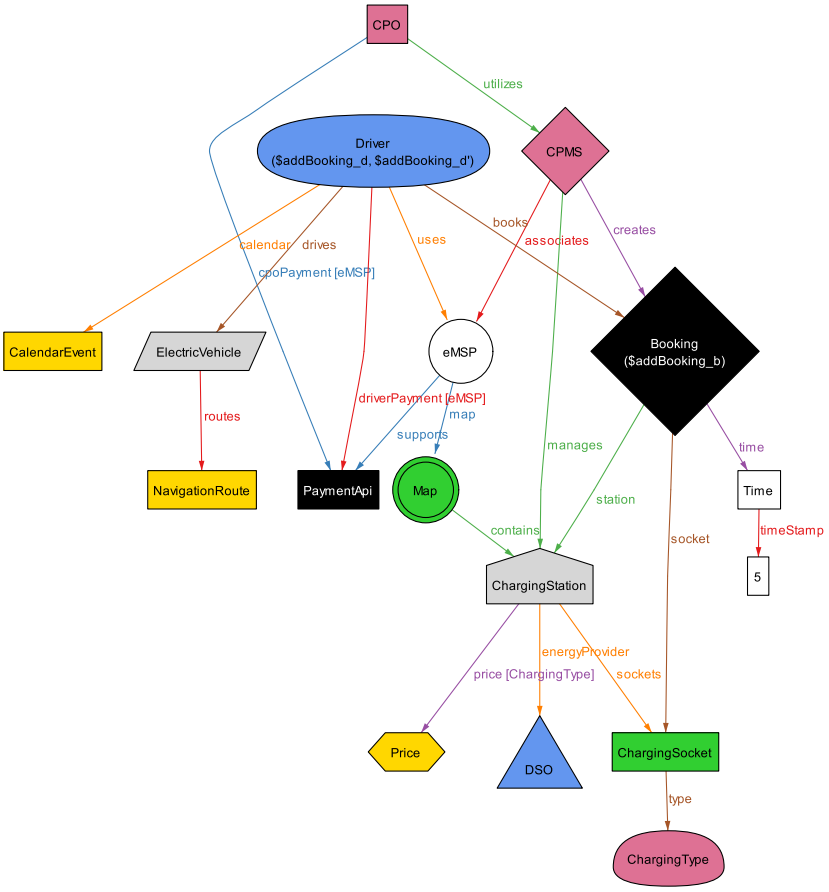

Figure 4.4: Booking Example

5 | EFFORT SPENT

Here is reported the amount of time spent writing this document.

Andrea Piras	Hours
General reasoning	12
Introduction	5
Overall description	8
Specific requirements	19
Formal analysis	6
Total	50

Emanuele Santoro	Hours
General reasoning	12
Introduction	4
Overall description	9
Specific requirements	18
Formal analysis	5
Total	48

Andrea Sanguineti	Hours
General reasoning	12
Introduction	6
Overall description	11
Specific requirements	18
Formal analysis	6
Total	53

6 | REFERENCES

6.1. Reference Documents

- Specification document: Assignment RDD A.Y. 2022-2023
- OCPI_2.1.1-RC1
- OCPP_1.6
- Course slides

6.2. Reference Sites

- https://www.overleaf.com
- https://app.diagrams.net/
- https://sequencediagram.org
- https://online.visual-paradigm.com
- https://www.figma.com
- https://github.com/Angtrim/alloy-latex-highlighting