POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

eMall – e-Mobility for all

Requirement Analysis and Specification Document - RASD

Author(s): **Fabio Lusha - 10882532**

Bianca C. Savoiu Marinas - 10684465

Academic Year: 2022-2023

Contents

Contents **i**

1 Introduction **1**
1.1 Purpose . . . 1
1.2 Scope . . . 2
1.3 Definitions, Acronyms, Abbreviations . . . 8
1.3.1 Abbreviations . . . 8
1.3.2 Definitions . . . 9
1.4 Reference Documents . . . 10
1.5 Document Structure . . . 11

2 Overall description **13**
2.1 Product perspective . . . 13
2.1.1 Domain model . . . 13
2.1.2 State charts . . . 15
2.1.3 Scenarios . . . 17
2.2 Product functions . . . 20
2.2.1 Data collection and management . . . 21
2.2.2 Communication and knowledge sharing . . . 22
2.2.3 Main functionalities . . . 22
2.3 User characteristics . . . 23
2.4 Assumptions, dependencies and constraints . . . 24

3 Specific requirements **25**
3.1 External Interface Requirements . . . 25
3.1.1 User Interfaces . . . 25
3.1.2 Hardware Interfaces . . . 25
3.1.3 Software Interfaces . . . 26

3.1.4 Communication Interfaces 28
3.2 Functional Requirements 28
3.2.1 Use cases 28
3.2.2 Unregistered and registered EVD's use cases 29
3.2.3 CPO's use cases 47
3.2.4 Requirements 62
3.2.5 Mapping on requirements 65
3.2.6 Mapping on goals 66
3.3 Performance Requirements 67
3.4 Design Constraints 69
3.4.1 Standard compliance 69
3.4.2 Hardware limitations 69
3.4.3 Any other constraints 70
3.5 Software System Attributes 70
3.5.1 Reliability 70
3.5.2 Availability 70
3.6 Security 70
3.6.1 Maintainability 71
3.6.2 Portability 71

4 Formal analysis using Alloy **73**
4.0.1 Resulting worlds 79

5 Effort spent **83**

6 References **85**

Bibliography **87**

List of Figures **89**

List of Tables **91**

1 | Introduction

1.1. Purpose

Widespread electrification of transport is the most efficient way to reach Europe's climate objectives for the sector and electric charging is the main asset to overcome the obstacles of the take-up of electric vehicles (EVs). EVs can reduce CO2 by an estimated annual 600,000 tons by 2030, going towards a carbon neutral Europe, and the importance of this aim raises the problem of having efficient systems that manage the charging services. The eMall is thought as an all-encompassing application that oversees the entire process from the user interaction to the effective recharge of the EV's battery.

The main goal we want to achieve with the eMall software is to help The EVDs (electric vehicle drivers) to have better access to recharge and to be able to book a charging point in order to avoid interference with his daily plans. Another important purpose of the system is to safeguard not only the users but also the providers of the service and this is made through privacy agreements and the actual interaction, that guarantees to supervise both interested parts, in order to get the best possible service and pay for it accordingly, having also a technical and economic exploitation of the charging infrastructures.

In this context there is an increase in the requested electric energy, but large amounts of power in short periods would require investments in the reinforcement of the distribution networks, which have not been designed to accommodate such load. It becomes necessary to introduce new systems and solutions to optimize the operation of the distribution networks. In this context we can identify the DSOs as the suppliers of electricity through the distribution networks. The DSOs interact with the eMall, and in particular with the CPMS (Charging Point Management System) module of the system to be. The CPMS, then, gives the information about the DSO's supply to the CPOs, which are important actors, that use the system in order to manage the charging service. A CPO is represented by an employee or a software, part of the business that owns some charging stations and wants to manage them through the eMall, deciding from where to acquire energy, and how to establish the prices, the special offers and other details about the stations.

The eMall is thought as a software that manages both the interaction with the businesses that offer the charging service and the interaction with the EVDs which want to use these services in order to charge their EVs. Therefore, the eMall provides a mobile application (eMma), which through its interface allows to the EVD to obtain the service, and provides, also, a web application that the CPOs use to manage the charging stations. The EVD interacts, as well, with the charging point interface (eMci), that communicates with the CPMS part of the eMall, in order to start the charging session from the station, plugging then the car to the compatible connector to effectively charge the EV.

Goals In the following table we present the main goals of the software to be. The goals capture the needs of the stakeholders, which are the EVDs and the CPOs.

Goal	Description
G1	The EVD is able to identify the charging stations nearby
G2	The EVD is able to visualize the tariffs of the charging stations
G3	The EVD is able to visualize any special offer available at the charging station
G4	The EVD is able to book a charge in a specific charging station for a certain time frame
G5	The EVD is able to start the charging process at a certain station
G6	The EVD is able to pay for the obtained service
G7	The CPO can decide from which DSO to acquire energy
G8	The CPO can decide the cost of charging
G9	The CPO can set special offers
G10	The CPO can decide whether to store or not energy in batteries
G11	The CPO can decide whether to use the energy available in the batteries

Table 1.1: Goals

1.2. Scope

World phenomena The portion of the real world where the machine is to be deployed and used is called the environment. Hence, scoping the problem by defining the environment is paramount, and this is the target of the next table. With the world phenomena we define the environment in which the software to be will operate, by clarifying some facts about the world and the users. In the following chapter, to the environment will be also given some boundaries, making some assumptions and describing some domains properties, but this is not an aspect tackled by the next table of phenomena.

World phenomena	Description
WP1	The EVD wants to charge the EV's battery
WP2	The EVD wants to plan where and when to charge the EV's battery, so he needs to know the position on the territory of the charging stations and if there are available and compatible charging points
WP3	The EVD wants to know the price and any special offers of the charging stations, to choose the one that better suits his needs
WP4	The prices of energy often vary in real world economy
WP5	The DSOs, as marketing strategy, have special offers during certain time periods.
WP6	The DSOs provide energy to the charging stations
WP7	The DSOs decide the energy price
WP8	The providers of the charging service (CPOs) make special offers during certain time periods
WP9	The CPOs decide the price of charging, following marketing trends, and depending on DSOs prices and business decisions
WP10	EVs may have an integrated rectifier that converts AC electricity to DC
WP11	Some type of chargers have an integrated rectifier that converts AC electricity to DC. They supply the EV directly with DC current
WP12	A charging of type X, provides electricity in mode C and is given through Z connectors
WP13	A charging station is owned and managed by one CPO
WP14	A CPO owns and manages one or more charging stations
WP15	The CPO buys energy from the DSOs
WP16	A charging station may be equipped with batteries
WP17	Charging stations equipped with batteries grant more flexibility to CPOs on how to choose between the energy stored in the batteries and the one offered by DSOs
WP18	Low voltage (3.7 - 11 kW) chargers need more time to charge the battery
WP19	Medium voltage (22-90 kW) chargers need less time to recharge a battery of capacity C than a low voltage charger

WP20	High voltage (> 90 kW) chargers need less time to recharge a battery of capacity C than a medium voltage charger
WP21	Batteries can only be charged with direct current (DC) electric power
WP22	Given a continuous supply of power W, and a battery with finite capacity C, than the charging time T is finite.
WP23	A battery can store a finite amount of energy, given by its capacity C.
WP24	The charging point of a specific charging station may be unusable because of maintenance or faults
WP25	The DSOs distribute and manage energy from the generation sources
WP26	Most electricity is delivered from the power grid as alternating current (AC)
WP27	During the day the electric power supplied to the station can vary
WP28	During the day a short-duration reduction in the voltage supplied to the electrical power systems may occur due to high current demand or faults in the system.
WP29	During the day a momentary increase in voltage may occur. This may happen when a heavy load turns off in a power system.
WP30	The DSOs operate and manage the electricity distribution networks
WP31	The DSOs solve grid problems, such as faults and network breaks
WP32	The DSOs use smart meters to detect interruptions and restore the supply of energy

Table 1.2: World Phenomena

Shared phenomena The shared phenomena define the interface through which the machine interacts with the world. The software monitors some shared phenomena, while controls others, and to show when the software takes the part of the controller and when the part of the observer we created two more columns in the next table, to keep track of the initiator of the action, in case the natural language turns out to be ambiguous. So, in the following assertions we present an interaction between the world (users and external

systems) and the machine (the eMall, especially its interfaces, such as the eMma, the eMci and the managerial web application).

Shared phenomena	Description	Controller	Observer
SP1	The eMall notifies the EVD when the charging process is finished	eMall	EVD
SP2	The EVD creates an account	EVD	eMall
SP3	The EVD in order to register inserts in the mobile app of the eMall the personal data (name, surname, email, password, payment details)	EVD	eMall
SP4	The EVD logs in using the email and the password	EVD	eMall
SP5	The EVD accepts the terms of service in order to use the eMma	EVD	eMall
SP6	The EVD shares its location with the eMall	EVD	eMall
SP7	The EVD confirms the payment from the mobile application of the eMall	EVD	eMall
SP8	The EVD deletes previously inserted EVs from its account	EVD	eMall
SP9	The EVD updates the specifications of the EVs on its account	EVD	eMall
SP10	The EVD adds a new EV to its account	EVD	eMall
SP11	The EVD updates personal data on its profile (such as email, payment)	EVD	eMall
SP12	The EVD inserts the maximum and minimum current supported by the EV	EVD	eMall
SP13	The EVD inserts the maximum power supported by the EV	EVD	eMall
SP14	The EVD inserts the inlet type of the EV	EVD	eMall
SP15	The EVD inserts whether the EV is equipped with a built-in rectifier	EVD	eMall

SP16	The EVD inserts the capacity of the battery in kWh	EVD	eMall
SP17	The eMall shows to the EVD the map of the charging stations nearby his location	eMall	EVD
SP18	The EVD chooses a charging station from the map	EVD	eMall
SP19	The eMall shows the user the rating of the charging station	eMall	EVD
SP20	The EVD inserts the expected time when he plans to start the charging process	EVD	eMall
SP21	The EVD inserts the expected time when he plans to end the charging process	EVD	eMall
SP22	The eMall shows to the EVD the list of available chargers of the charging station	eMall	EVD
SP23	The eMall shows the charger type and its connectors	eMci / eMma	EVD
SP24	The EVD chooses the charger he wants to use from the list of available ones	EVD	eMall
SP25	The eMall shows to the EVD the charger costs (per kWh, per minute, additional costs)	eMma/eMci	EVD
SP26	The eMall shows to the EVD the status of the charger	eMci	EVD
SP27	The eMall shows to the EVD the battery level of the connected EV	eMci	EVD
SP28	During the charging session the eMall shows to the EVD the power output of the charger	eMci	EVD

SP29	During the charging session the eMall shows to the EVD the remaining time to complete the charging process	eMci	EVD
SP30	The EVD starts the charging session from the charger	EVD	eMSP
SP31	*The CPMS asks the DSO about the current available energy sources, their prices, and special offers*	CPMS	DSO
SP32	The DSO dynamically changes the price of electricity	DSO	CPMS
SP33	The DSO changes dynamically the *energy sources* from which acquires energy	DSO	CPMS
SP34	The DSO makes special offers	DSO	CPMS
SP35	The CPO logs in	CPO	CPMS
SP36	The CPO selects the charging station for which to set the parameters (price, energy) of the charging service	CPO	CPMS
SP37	The CPO selects the DSO from which to acquire energy	CPO	CPMS
SP38	The CPMS shows to the CPO the *energy sources* and the relative current prices and special offers of the DSO	CPMS	DSO
SP39	The CPO sets the cost of charging	CPO	CPMS
SP40	The CPO can set a special offer	CPO	CPMS
SP41	The CPO selects the energy sources from which to acquire energy	CPO	CPMS
SP42	The CPMS shows if there are available batteries in the charging station	CPMS	CPO
SP43	The CPO selects the battery in which to store energy	CPO	CPMS

SP44	The CPO sets the amount of energy to store in the battery	CPO	CPMS
SP45	The CPMS dynamically shows to the CPO the number of EVs charging	CPMS	CPO
SP46	The CPMS dynamically shows to the CPO the charging stations consumption of energy	CPMS	CPO

Table 1.3: Shared Phenomena

1.3. Definitions, Acronyms, Abbreviations

1.3.1. Abbreviations

- **eMall**: e-Mobility for all
- **eMma**: e-Mall mobile application
- **eMci**: e-Mall charger interface
- **CPMS**: Charging Point Management System
- **CPO**: Charge Point Operator
- **eMSP**: Electric Mobility Service Providers
- **DMS**: Distribution Management System
- **DSO**: Distribution System Operator
- **EV**: Electric Vehicle
- **EVD**: Electric Vehicle Driver
- **EVSE**: Electric Vehicle Supply Equipment
- **HV**: High Voltage
- **LV**: Low Voltage
- **MV**: Medium Voltage
- **SCADA**: Supervisory Control and Data Acquisition

- **SCM**: Smart Charging Management
- **OMS**: Outage Management System
- **AC**: Alternating current
- **DC**: Direct current

1.3.2. Definitions

- **DSO**: typically the entity responsible for the operation and management of distribution networks – High, Medium and Low Voltage networks. For this purpose, the DSO typically owns systems such as Supervisory and Control Data Acquisition (SCADA)/ Distribution Management System (DMS) for the monitoring and general overview of the state of the network. It also owns other systems such as the Outage Management System (OMS) and Work Force Management System (WFMS) for addressing the network operation problems related with the continuity and quality of service.
- **CPO**: entity that technically manages all the EV infrastructure assets, depending of existing country regulation – this role can be assured by the DSO or other entity.
- **eMSP**: is the entity that can explore the economic side of the EV charging infrastructure, namely by selling energy for charging purposes.
- **CPMS**: is a software system that manages the charge point infrastructure – can manage the technical and economic aspects of the charging infrastructures.
- **EVD**: person or entity who owns an EV car and can use the public or private facilities for charging purposes.
- **EVSE**: Electric Vehicle Supply Equipment. It is an equipment that is able to charge EV batteries with AC or DC loads and with different rated powers depending on the type of equipment.
- **Private parking**: can be a condominium, industry or other entity who has private owned EV
- **Voltage sag**: a short-duration reduction in voltage of an electric power distribution system. It can be caused by high current demand or fault current elsewhere in the system.
- **Voltage swell**: the opposite of voltage sag. Voltage swell, which is a momentary increase in voltage, happens when a heavy load turns off in a power system.

- **Socket outlet**: the port on the electric vehicle supply equipment (EVSE) that supplies charging power to the vehicle
- **Plug**: the end of the flexible cable that interfaces with the socket outlet on the EVSE.
- **Cable**: a flexible bundle of conductors that connects the EVSE with the electric vehicle
- **Connector**: the end of the flexible cable that interfaces with the vehicle inlet
- **Vehicle inlet**: the port on the electric vehicle that receives charging power
- *Inverter*: It is a power electronic device or circuitry that changes direct current (DC) to alternating current (AC).
- **Rectifier**: an electrical device that converts alternating current (AC) to direct current (DC).
- **eMma**: the eMSP subsystem responsible for the EVD interaction from the mobile app
- **eMci**: the eMSP subsystem responsible for the EVD interaction at the charging point
- ***additional costs****: overtime penalty, deposit for unregistered users*
- **Status of the charger**: can be free, occupied, booked and in maintenance
- **Smart meter**: is an electronic device that records information such as consumption of electric energy, voltage levels, current, and power factor; allow the reading of energy flow and real-time usage, and consequently permit the identification of interruptions in energy flow

1.4. Reference Documents

- `IEEE 29148-2018 International Standard - Requirements engineering`: defines the construct of a good requirement and provides attributes and characteristics of requirements; provides also additional guidelines for applying the requirements and requirements-related processes
- `RDD` assignment document
- `Electric Vehicle CPMS and Secondary Substation Management by F. Campos, Efacec, Portugal; L. Marques, Efacec, Portugal and K. Kotsalos, Efacec,`

Portugal (15 October 2018): used to define the interactions between the different parts of the system and the actors; models the EV public infrastructures, the eMSP, the DSO and the CPMS together with the APIs and protocols that allow their communication

- EV CHARGING: HOW TO TAP IN THE GRID SMARTLY? by Platform for Electromobility (May 2022): used to understand the environment of the problem and contextualize the analysis

1.5. Document Structure

This document mainly follows the guidelines of the IEEE 29148-2018 - International Standard - Requirements engineering, with some changes in the order of the contents. Also in the final part of the document is present an Alloy formal analysis of the described model, an additional section with respect to the standard. The document is composed by the following parts:

- An introduction to the domain in which the system will operate (world phenomena) and an initial description of the software to be, the eMall, specifying the goals to achieve
- The overall description of the functions that the eMall has to implement specifying the requirements and a domain model, its interaction with the different users describing with diagrams the most important shared phenomena, and the domain assumptions necessary to the system to be
- A thorough list of requirements both functional and non functional: giving a detailed description of the functional requirements using use cases, use cases diagrams and UML sequence diagrams to better specify the interactions; and characterizing the non functional requirements through software system attributes
- A formal analysis using alloy in order to show the soundness and correctness of the model described in the document, considering only a part the most important requirements of the system
- A section that contains the effort spent by the members of the group working towards the completion of this document

2 | Overall description

In this chapter a general overview of our software to be and its functionalities is given. In section 2.1 we will present a conceptual model of the domain we are working in, where, in addition to the real world objects that are significant in our domain and to our system, we include the main components of our system that will interface with the environment. Then, we proceed by presenting the state diagrams of the most important scenarios discussed in the following subsection. In section 2.2 we give a description of the main functionalities our system has to provide, but without going in much detail since we will delve into this aspects in the next chapter. In section 2.3 we provide an analysis of the target users of the system, which are the clients that will use the system or will interact with it. Finally, in section 2.4, we outline the assumptions, the dependencies and the constraints, necessary to be taken into account when implementing the software.

2.1. Product perspective

2.1.1. Domain model

We start off this chapter by analyzing the domain model (or *conceptual model*) we came up with to represent the domain we are working in. Being a conceptual model the diagram was not drafted with all the formalism specified in the UML notation; we actually used a pretty informal description, specifying multiplicity and reading direction only when strictly necessary. The model in figure 2.1 was drawn using the UML class diagram notation, and illustrates only the conceptual classes that are significant to the domain [1]. We also included the classes representing the component of our system to be that will interact directly with the environment.

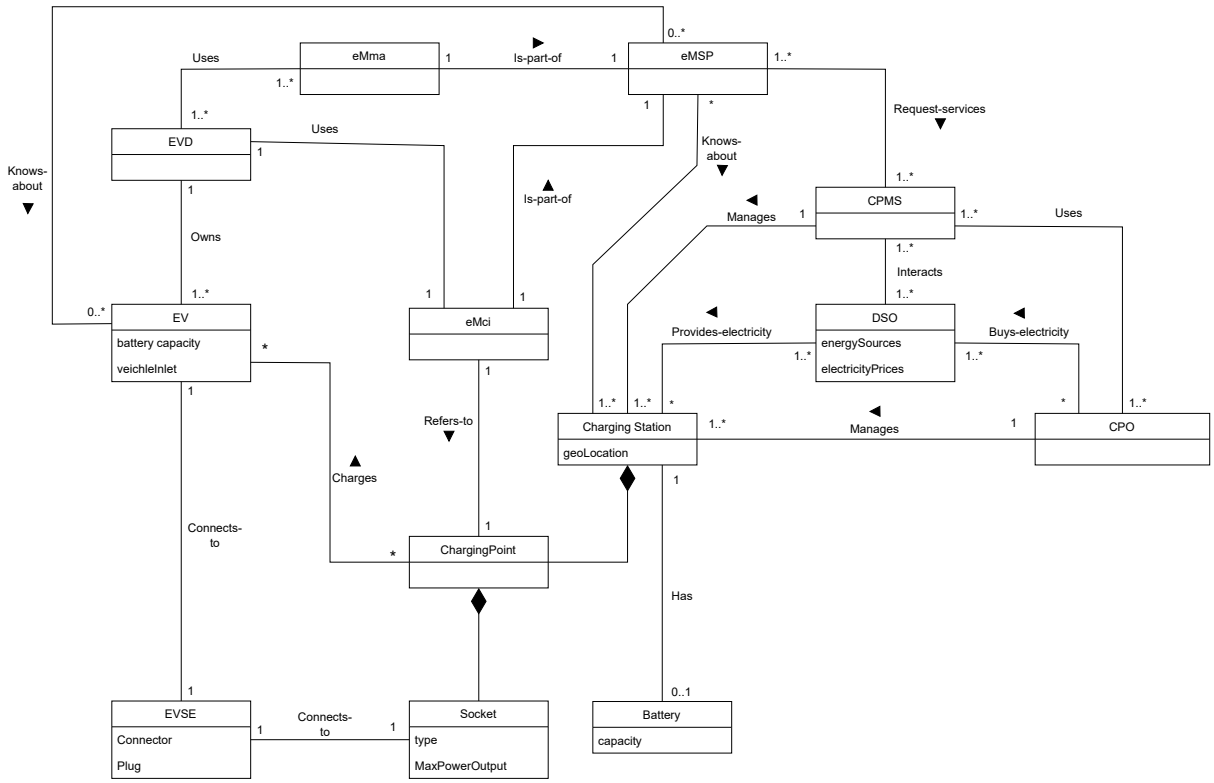

Figure 2.1: Domain model

We now proceed discussing some of the elements that may not be so immediate only by reading the domain model.

EVSE The EVSE, as mentioned in 1.3, is a general term that refers to the the equipment that allows a vehicle to be recharged. In this context we can see it as a universal adapter that can be an interface for the vehicle inlet and the socket of the charging point. In the diagram this is shown by the association *'Connects-to'* for both class EV and Socket. The multiplicity one-to-one for both associations is to show that at a specific moment a vehicle can be connected to a socket only through one single EVSE [1]. Obviously, an EV can connect to all the EVSEs that have a connector compatible with its inlet, and a Socket can connect to all the EVSEs that have a compatible plug. It also should be noticed that there are charging points that do not expose the socket directly, but they have an EVSE integrated, namely a cable with it's connector.

DSO In the model the DSO is represented as interacting with both CPO and the CPMS. This happens because we do not distinguish between the actual business and the Information System he uses. So the association between DSO and CPMS is to be intended as the interaction between the two software systems, meanwhile, the association between

DSO and CPO represents the interaction of the two businesses.

ChargingPoint This class represents the actual physical device that contains the sockets to which the EV will connect to charge.

Battery In the association *'Has'* between ChargingStation and Battery we have stated that an instance of ChargingStation may have only one Battery. The reason behind this decision is that we opted for a high level of abstraction and with the class Battery we mean a generic mean of energy storage, without concerning ourselves with the actual physical properties or requirements of the item itself.

CPO The multiplicity one-to-many in the association *'Uses'* between CPO and CPMS is motivated by our view of the system. In our perspective, the CPMS is viewed as a software that is offered to different businesses that manage their charging stations, thus a business through the CPO can choose to manage the charging stations with different CPMS systems.

2.1.2. State charts

Among the main interactions with the eMall we have decided to represent here with state diagrams the ones that we consider the most interesting and complex uses of the system, from EVD point of view: the initiation of a charging station and the booking operations.

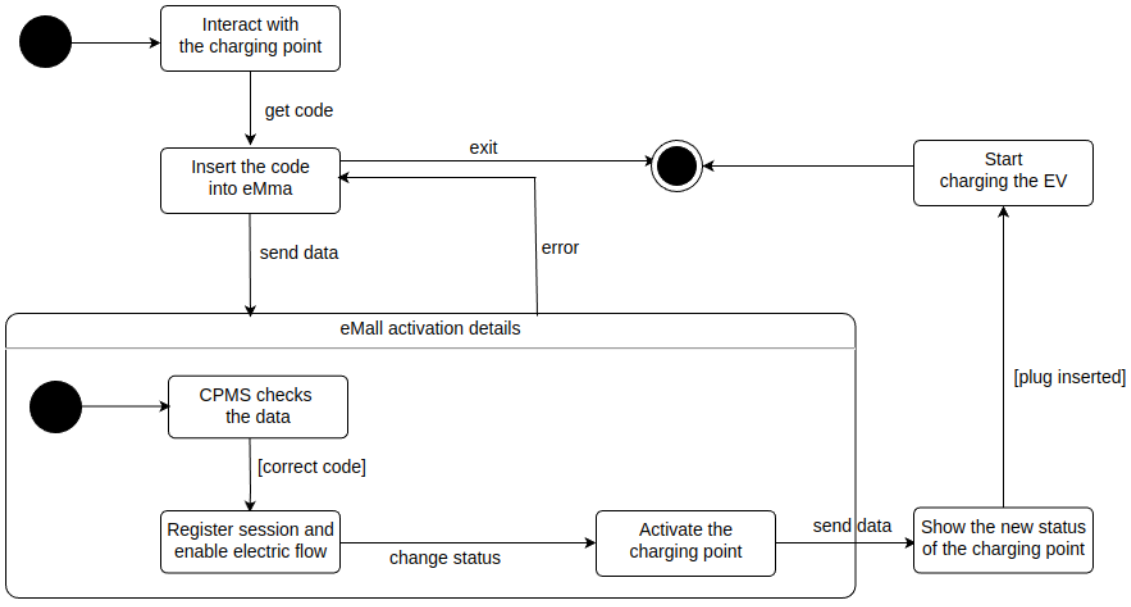

Figure 2.2: State diagram of the EVD that starts a charging session

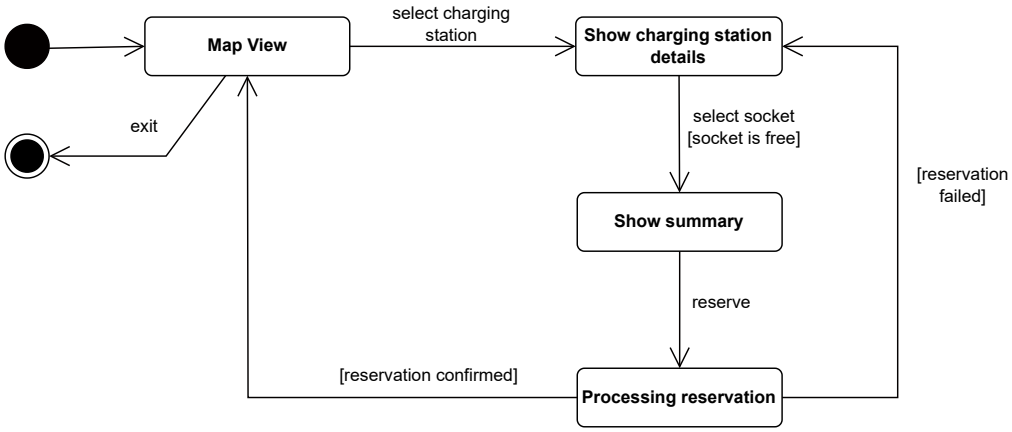

Figure 2.3: State diagram of the EVD that books a charging point

We also report another state diagram to represent the main interaction of the CPO with the managerial part of our software. We consider the case in which the CPO wants to modify some parameters regarding a certain charging station.

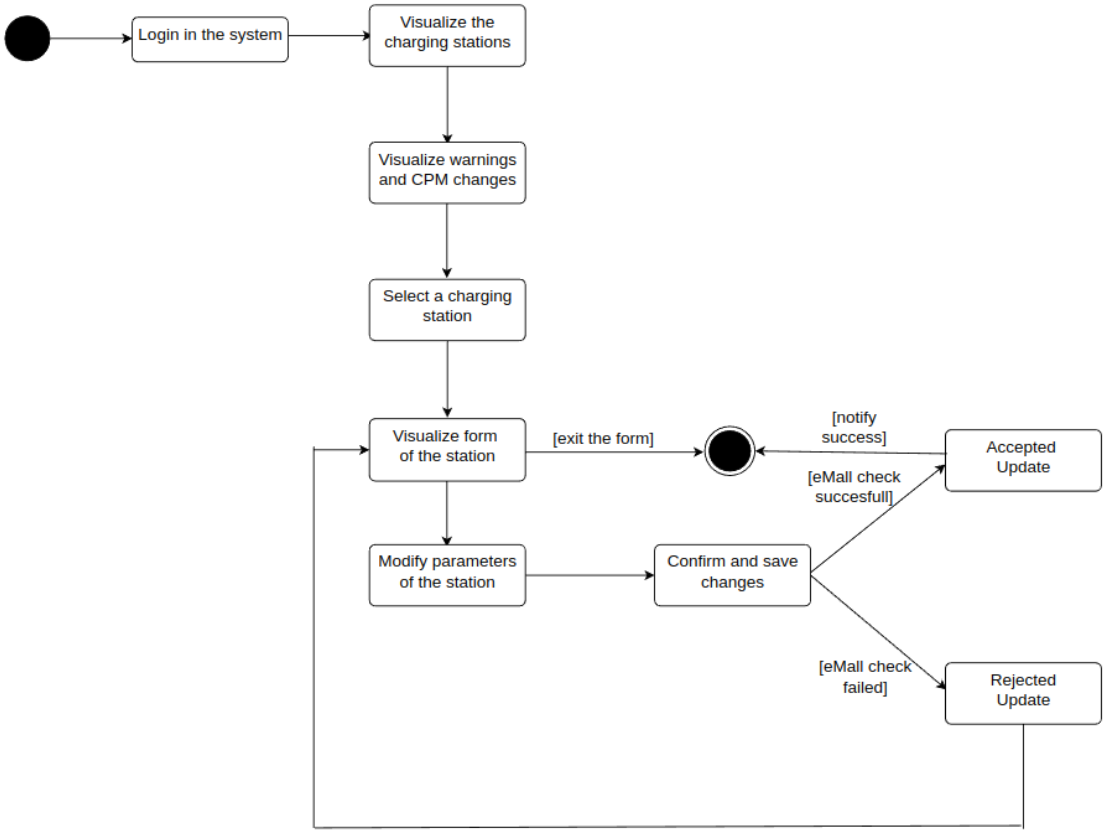

Figure 2.4: State diagram of the CPO that manages charging stations

2.1.3. Scenarios

Booking a charging point Edward, after getting in his electric vehicle, notices that the battery is at low percentage, so he plans to book a charge at a station nearby. He grabs the smartphone and opens the mobile app eMma to look for a charging station. When Edward opens the app he is greeted with the view of a map showing him the nearest charging stations to his location. The charging stations are represented with icons of different colors. The colors are used to distinguish totally occupied charging stations from those with free sockets where to charge. After moving around the map, Edward finds a charging station with free sockets suitable to his needs. He clicks on the icon symbolizing the charging station and a new view is drawn on the screen. The view visualizes information about the specific station, namely:

- The name of the charging station
- the rating of the charging station
- indication about the available sockets types and their number
- contact details
- address of the charging station
- any directions on how to handle the charging process
- reviews relative to the charging station

If the EVD is satisfied with the characteristics presented by the station he click on the button 'Book now' opening yet another view, which lists all the available sockets, pointing out the following information for each one of them:

- The type of charging (AC/DC)
- The type of the socket (type 1, type 2, CCS, CHAdeMO, etc.)
- the charging speed denoted in kW and km/h (km gained per one hour of charge)
- The price for kWh
- The price for unlocking the socket

Scrolling down on the app, additional information regarding the station and the charging process are shown, like:

- accessibility to the station

- any additional fees set by the CPO, such as the cost per minute for parking during the charging process and the cost per minute for parking after finishing the charging (penalty for occupying the spot and not using it)
- Taxation information (VAT etc)

Being satisfied by the features offered by this station, Edward selects the appropriate socket for his EV and equipment (any additional charging adapter) and clicks the button to reserve the spot for the next X minutes.

Update profile details Jay is an electric vehicle enthusiast, who bought himself a new EV, in order to reduce the negative impact on the environment. Given this new purchase Jay needs to update its profile on the eMma in order to take advantage of the eMall service at its most. He logs in to its account on the mobile app and from the main page navigates to his profile. On the profile page are visible the personal information and the details about the EVs. Furthermore, there is the button that allows to update the profile and this is exactly what Jay is looking for. After pressing the button 'Update' there are different possibilities and Jay chooses the one that states 'Add new vehicle'. Now he has to fill up a form with the EV's details, such as type, capacity of the battery, supported power and current and so on. After double checking the form Jay presses the 'Ok' button and the page reloads showing again the profile page that now states among the other vehicles also the new one. Considering the eventuality of making mistakes in completing the form it is always possible to come back to the EV details and change any present field.

Visualize charging history It has been nearly 6 months since Hannah bought her first EV and now she has fully grasped how the whole ecosystem around it works. In the past few months Hannah has tried quite a few different charging stations to explore how each one is managed and organized. Having tried all these charging stations Hannah is curious to see how many charging stations she has visited, how frequently and how much she has paid for the charging. With this objective in mind Hannah opens the eMma application in her mobile phone and heads to the history section of it. This section is divided in two parts: in the top half the app shows the imminent charging booking that Hannah has reserved, if present. In the bottom half of the screen, the app shows a chronologically ordered list of all the charges processed through eMma. Each entry in the list shows the date in which the charging was done, where it was done, for how long, how many kWh were charged, the type of the socket used and how much did it cost distinguishing between cost for kWh and total cost.

Start a charging session Adeline usually goes to the supermarket nearer to her house because it has a charging station in the parking area. Most of the times she finds an available charging point so she charges her EV while she does the grocery. Once stopped the car in the available spot Adeline wants to start a charging session. She interacts with the charging point interface, visualizes the information about the available charge with the respective power and cost and inserts the code shown on the screen in the mobile app of the eMall, the eMma. Once initiated the session from the mobile phone, the data are sent to the system, and in particular to the CPMS part of the software that checks the correctness of the inserted code and registers the session related to the user. Then, the CPMS enables the flow of electricity in order to actualize the charging from the charging point. If during the check of the data and during the activation operations there are no errors the status of the charging point changes and the session is activated. Now, Adeline can insert the specific plug, compatible with her car, in the EV in order to actually start the charging, which if not stopped earlier will terminate when the battery is full.

A new user registers into eMma Michael, proud owner of an EV for 5 years, has decided to try this new charging app, eMma, that is promoting itself as a better alternative to manage in a smart way the charging process of an EV. Michael decides to give eMma a chance, downloads the app and immediately initiates the procedure to create a new account. The first phase of the registering process is straightforward; the usual information about name, family name, email and password are requested. After completing this first phase eMma prompts the user with a message asking him the consent to use his geographical location and to accept the terms of service. Micheal gladly accepts because he wants the app to show him the charging station nearest to him based on his location. After granting the consent a new page is presented to Michael. This time it is a form to be completed with the information about Michael's EVs specifications and his EV gear, like charging adapters and cables. Michael understands that this information is needed so the app can work in a smart way, showing him only the charging stations that have sockets compatible with Michael EVs connector or adapters. Finally the process to complete the registering begins; a form where Michael has to add his electronic payment details. After completing this last stage, the app opens and shows Michael a map of the area around him where the charging stations are highlighted with icons of bright colors.

Visualize the charging stations map Daisy is an unusual user of the eMall, that didn't registered an account. Anyway the system allows the possibility to use the application as a guest, but the functionalities are limited. Daisy is only interested in visualizing the charging stations nearby, so she opens the main page of the app in order to look at

the map. The system retrieves, based on the location shared by the phone, the charging stations in the area and shows them on the map. Daisy can now explore the charging stations around clicking on them on the map, and she can see their rating with the relative reviews and can choose the service that better fits her needs. She can visualize the price and the available chargers with their type of connectors, but she is not able to book a charging session without an account. Once identified her preferred charging station Daisy closes the application, gets in her car and heads directly to charge her EV.

Manage the charging stations Nick is a CPO that on a typical day has to monitor the charging stations assigned to its department. After arriving to work and logging in the system with the company credentials Nick checks the list of charging stations and any new notification given by the CPMS part of the eMall about the DSOs decisions. He sees a warning regarding the recently deployed charging station in Rome and clicking on it the web application of the eMall shows a form with the various characteristics of the station. The parts that may have undergone a modification are highlighted in red and in this case the selection of the DSO has new options available and Nick clicks on it to explore the more convenient ones. He notices a change in one of the DSOs that now grants energy also through renewable resources, and given the green policy of the company he chooses this new kind of supply. Considering the price of 0.036/kWh provided by the DSO, in order to have a gain, according to the business modus operandi Nick sets the cost charging at 0.040/kWh. Once confirmed the DSO from which to acquire energy by looking carefully at the form he becomes aware of the fact that at the moment there are no EVD charging at the station and also sees that there are available batteries in which to store energy. One of the batteries is empty, so he selects it in order to store energy in it right away, until the full capacity C, given the off-peak moment. After all this operations Nick saves the changes and the eMall notifies him about the success of the procedure, that has an immediate effect on the system and his interaction with the world. Nick moves on to the next charging station of his list, checking up in similar way each one of the stations for which he is responsible.

2.2. Product functions

In this section, we briefly represent a list of the most important requirements of the `eMAll`, remainig on a high level of abstraction, since we will proceed to further discuss about them in much more detail in the next chapter.

2.2.1. Data collection and management

One of the main functionalities of the software is to store and manage different kinds of data coming from different sources:

1. The EVD using the eMma inserts into the system different kind of data. He inserts personal data, such as name, surname, and payment details; he also adds information about his EVs, like the maximum and minimum current supported, the connector type, the battery capacity and other relevant facts, like any additional EVSE he might own. The eMall allows the insertion of structured data and full-text elements that are subjected to checks in order to verify their correctness. The software maintains these data on the database in order to associate the bookings and the charging sessions to all registered EVDs, who can access all the functionalities of the system and are not subjected to the payment of a deposit every time they use a charging point

2. The DSO provides energy to the charging stations, and the information about the DSO's supply is automatically collected by the CPMS subsystem of the eMall through interfaces that interact with the external systems. The CPMS acquires the information and saves it on the database in order for it to be visible to the CPO, and updates these data periodically. The collected data deriving from the DSO's are essential for the businesses, which make their supply choices depending on the price, the availability and the kind of acquired energy

3. The CPO manages the charging stations and their supply, visualizing the information kept by the software and making data-driven decisions for each one of the charging stations owned by the company. The CPO can see the parameters of each station and change them based on the new prices and types of energy, based on the chosen DSO to acquire from and based on the new politics of the CPO's company. All the data updates done by the CPO are received by the CPMS and collected by the system, so the managerial part of the service constantly produces data, about the charging stations. These data are stored and then used by the software to inform the EVDs of the charging stations details. The eMall also keeps data about the charging points and about the presence of batteries in each charging station, and these are useful information that need to be collected in order to allow to the CPOs to manage the service accurately

4. The charging station itself is an important source of data. Information about the charging points usage, both in terms of frequency throughout different periods of time (day, week, month) and usage time (for how long a certain socket has been used

for each charging process) must be kept to enable the eMall system to conduct data analysis procedures (can give information about peak load hours) and empower the CPOs with relevant data for the business decision making process. Other information that can be tracked through the system include: client profiling (keep track of clients who visit the charging station), maintenance record, unused bookings profile

2.2.2. Communication and knowledge sharing

The eMall provides different tools to the EVD and to the CPO in order to take advantage of the service and obtain all the needed information from the system.. To be able to share this knowledge the subsystems of the software need to communicate among themselves and with the external entities. The offered tools are the following:

1. The eMma presents to the EVD all the information needed about the nearby charging stations. The application shows a map with the charging stations, and selecting a station the user can visualize further data, such as price, socket type, free charging points and other details. The eMma and the eMci are able to provide these information, because are part of the eMSP, which communicates with the CPMS to acquire the data about the charging stations
2. The web app available to the CPO, communicates with the CPMS part of the software getting the data about the electric supply offered by the DSOs, acquiring knowledge about the prices, the special offers and the available electric sources. The CPMS updates the information interacting periodically with the external service of the DSOs and shares the knowledge with the CPO

It is evident that among the functionalities of knowledge sharing and communication between the components involved, we also have as main features the following:

1. The eMma shows to the EVD the information about the nearby charging stations
2. The eMci shows to the EVD the data regarding the charging point in use
3. The CPMS gives to the CPO the knowledge of the DSOs changes and the last data saved for each charging station managed by the CPO

2.2.3. Main functionalities

Regarding the main functionalities that the EVD perceives, except for the ones already described, the most important ones remain:

1. The eMma allows to the EVD to book a charging point in a chosen time frame. Once

the booking is completed from the app, the EVD receives a confirmation notification and the booking with an associated code is added to the user history of charges. The system saves the data related to the registered EVD and to the booking, so the eMSP maintains a copy of the code provided to the user, the data associated to the charging station and the chosen time frame. The effective charging service will be provided when the user will correctly insert the received code into the che eMci of the specific charging point. The eMall, after checking the code, activates a charging session with respect to the EVD, having in this way that the system provides the functionality of charging the EV in the time frame previously booked

2. The eMall gives, also, the possibility to charge without booking. In this case the EVD interacts with the eMma and the eMci. From the two interfaces the data arrive to the eMSP, which creates the charging session and allows the user to use the service

2.3. User characteristics

The eMall has three main user classes:

1. **Unregistered EVD**: An EVD can register to the eMall or use the service without registration. In order to register, the user has to introduce personal data and the details of the EVs, so he creates a profile with an associated name and a password. By creating a profile is possible to take advantage of all the features provided by the service, having some privileges, but the eMall can also be used without any registration. The eMma can be downloaded on the phone and used as a 'guest' and in this case is still possible to visualize the map with all the nearby stations and their information. It is also possible to book a charging session from the application, but is necessary at least the insertion of the payment details and the payment of a deposit in advance in order to use this functionality. Even in the case of charging the EV without any booking, the unregistered EVD has to give a deposit before starting the charging session. Furthermore, the EVD without a profile doesn't have the history of charges, so there are some limitations in using the system

2. **Registered EVD**: An EVD is registered if creates an account inserting personal data and EVs details. The registered EVD interacts with the eMma and the eMci in order to use the main functionalities of the system: to book a charging session, to charge the EV without a booking, to visualize the nearby charging stations and to visualize and modify the personal profile and history. The EVD, registered or unregistered, can be unfamiliar with the use of mobile applications, so the software

needs to be user-friendly in order to guarantee a good service in all its aspects

3. **CPO**: A company that supplies the service is identified with the employees or the existing software, that interacts with the eMall system. In the interaction the part of the company is called the CPO and manages the charging stations provided by the company itself. The CPO is able to visualize all the stations and the respective charging points and can change the supply parameters, modifying the price of the charge, the storage of energy, the DSOs from which to acquire electricity and other details. All these changes are possible given the interaction of the CPO with the CPMS part of the eMall, which has the necessary knowledge, that is communicated to the company in order to administer the stations and offer the service properly

2.4. Assumptions, dependencies and constraints

Assumptions	Description
D1	The EVD has internet connection
D2	The EVD has a mobile phone with an integrated GPS module
D3	The EVD has the mobile application of the eMSP installed on his mobile phone
D4	The EVD inserts correct data in the mobile application
D5	The CPO inserts correct data in the web application
D6	The CPOs send the data of the charging stations to the eMall
D7	The end user payment from the mobile app is handled by external APIs.
D8	The EVD that creates an account inserts the personal data and the payment details during registration
D9	The CPO uses company credentials to access the web application of the eMall

Table 2.1: Assumptions

3 | Specific requirements

3.1. External Interface Requirements

3.1.1. User Interfaces

The eMall is modeled as a software with two possible user interfaces, one for the mobile application, which will be available to the users, and one for the web application available to the businesses, that offer the charging service.

The user interface of the eMall, that the EVDs interact with, is thought as a mobile application, the eMma, easy to use and intuitive, allowing users to quickly and easily access the features they need to charge their vehicles. The EVD needs to download the mobile app on his cellphone in order to interact with the eMall and take advantage of its functionalities. We want the application to be, also, visually appealing and easy to navigate, with well-designed buttons, menus, and other elements that make it easy for users to find the information they need and interact inserting the necessary data. Additionally, the UI should be responsive, ensuring that it works well on all mobile phones, regardless the screen size.

The other user interface is the one provided to the CPOs, which are in charge of managing the charging service for the businesses involved. In this case the interface is a web application, which we also thought as easy to use, with a clear visualization in order for the CPOs to be able to keep track of all the charging stations and manage them properly. The UI in this case offers more complex features, also, allowing to the user to modify the graphical parts and personalize the aspects of the application. Exactly like for the mobile application, in this case we want a web application that allows a fast interaction without performance issues, and that works on any browser.

3.1.2. Hardware Interfaces

The main components of our system to be: eMma, eMci, eMsp and CPMS all have different hardware needs thus, in the proceedings paragraphs we shall discuss each of

them individually.

eMma eMma is the mobile application so it must be able to run on a range of mobile platforms. In particular the scope of this part of the system regard only mobile platforms known as smartphone, so the system must be able to adapt to the smaller screen sizes and limited processing power of mobile devices, while still providing a user-friendly and intuitive interface. Furthermore, it is required that the mobile device be equipped with a GPS sensor to offer the geolocation service of our system.

eMci is the part of the system that will run on device mounted on the charging point. This kind of devices may not have common architecture, so a variety of platforms may be expected. We can consider this device to be an 'embedded device'. The characteristics we assume to hold for this devices are:

- Limited computing power
- connectivity module (wireless or wired) to connect to the internet
- connection to the internet

eMSP This part of the eMall is the one that must handle user request, coming from eMma, and communicate with the CPMS. This part of the system should run on a general purpose computer, so the hardware interfaces are the ones of common knowledge.

CPMS This part of the system is intended to have a web graphical interface to be viewed on a screen of a laptop or desktop PC, so it must be build for screens with size greater than 13 inches. The other part is of the CPMS is intended to be run on a general purpose computer, so like the eMSP no specific hardaware interfaces are requested in addition to those provided by a classical computer.

3.1.3. Software Interfaces

Like in the section above, we will proceed describing the software interfaces distinguishing it for each of the main component of our system.

eMma The mobile application of the eMSP will have to be compatible with the major platforms present in the market, namely iOS and Android. We do not care about other mobile platforms since with this two we cover the 99% of mobile users. If implementing the system for older versions of the two OSes becomes tiresome, then this older version

shall be not considered. Nonetheless, the system shall be compatible with at least Android 10 and iOS 16. Targeting this two versions of the OSes will cover 72% of iOS users[1] and nearly 70% of Android users[2]. For the interaction with other components vital to our system, like internet interface and GPS interface, the modules provided by each OS will be sufficient.

eMci Considering that this part of the system will run on an 'embedded device', interfacing with am embedded OS will be mandatory. Integrating an embedded OS in the device will make the development much easier and faster, given that most of device will be managed by the OS and the development team will only need to interact with the OS. The embedded OS should also integrate the driver for the network card and offer an interface to interact with it. Moreover, the system will have to communicate with the software running on the charging point to monitor it's status and the charging status when an EV is charging.

eMSP

1. The system will run on a general purpose computer so they must interact with the Operating Systems running on the device
2. It must keep track of user data, so it needs to interact with a persistence system like a DBMS
3. must be able to interact with multiple CPMSs, so the system shall support interoperability with different CPMSs

CPMS

1. he system will run on a general purpose computer so they must interact with the Operating Systems running on the device.
2. It must keep track of CPO instructions, charging station status, etc., so it needs to interact with DBMS
3. It shall interact with the software managing the technical aspect of the charging points, like handling the electricity flow, so the system must know how to instruct this software
4. It shall interact with the different API's provided by different DSOs

[1] `https://gs.statcounter.com/ios-version-market-share`
[2] `https://gs.statcounter.com/os-version-market-share/android`

5. If a system of energy storage is present, namely a battery, the CPMS shall interact with the software managing it, so it can give directives on how to proceed

3.1.4. Communication Interfaces

Here we will specify the main communication interfaces that can be used to solve the communication and integration of different components.

1. eMma and eMci are part of the eMSP that that run on separate machine from it. This means that they'll need to communicate using some kind of network protocol. In particular we shall use the internet stack of protocol. Surely up to the transport layer (TCP/UDP) the standard protocols shall be used, meanwhile the application layer shall be let to the architectural designer to decide weather to use a proprietary protocol or a standard one like `http`.
2. The eMSP needs to communicate with different CPMS. We can't make any assumption on how each different CPMS will provide their services so this part of the eMSP that handles the communication shall be constructed in the most general way. If, during further analysis, this implementation becomes too troublesome and complex than an assumption can be made and build the system such that it can communicate with only the CPMSs that use the `OCPI` protocol.
3. The CPMS needs to communicate with the software managing the charging points. An open standard protocol has been devised for this purpose and is widely adopted, the OCPP. Our system will be using this protocol to communicate with the charging points.
4. The CPMS needs to communicate with different DSOs to get information about the price, source and other characteristics of the supplied electricity and to conclude a purchase agreement. For this kind of communication there is no open standard protocol, so an ad-hoc interface needs to be build to be able to interface with different DSOs.

3.2. Functional Requirements

3.2.1. Use cases

In this section we present the most important use cases of the eMall. Every use case has also associated a sequence diagram, but we decided to present only the most relevant situations of the use case in the sequence diagrams, so they do not contain all the

possible alternatives. We also provide some use cases diagrams in order to better show the relationship between the actors and the actions that they can perform in the system, displaying the features and the capabilities of the application. The following use cases are also related to the scenarios explained in the last chapter, formalizing the situations previously described and further analyzing the present features. The use cases are also accompanied by a sequence diagram, which better represents the event flow, but we tried not to add too much details, that clusters up the diagram and makes it difficult to read.

3.2.2. Unregistered and registered EVD's use cases

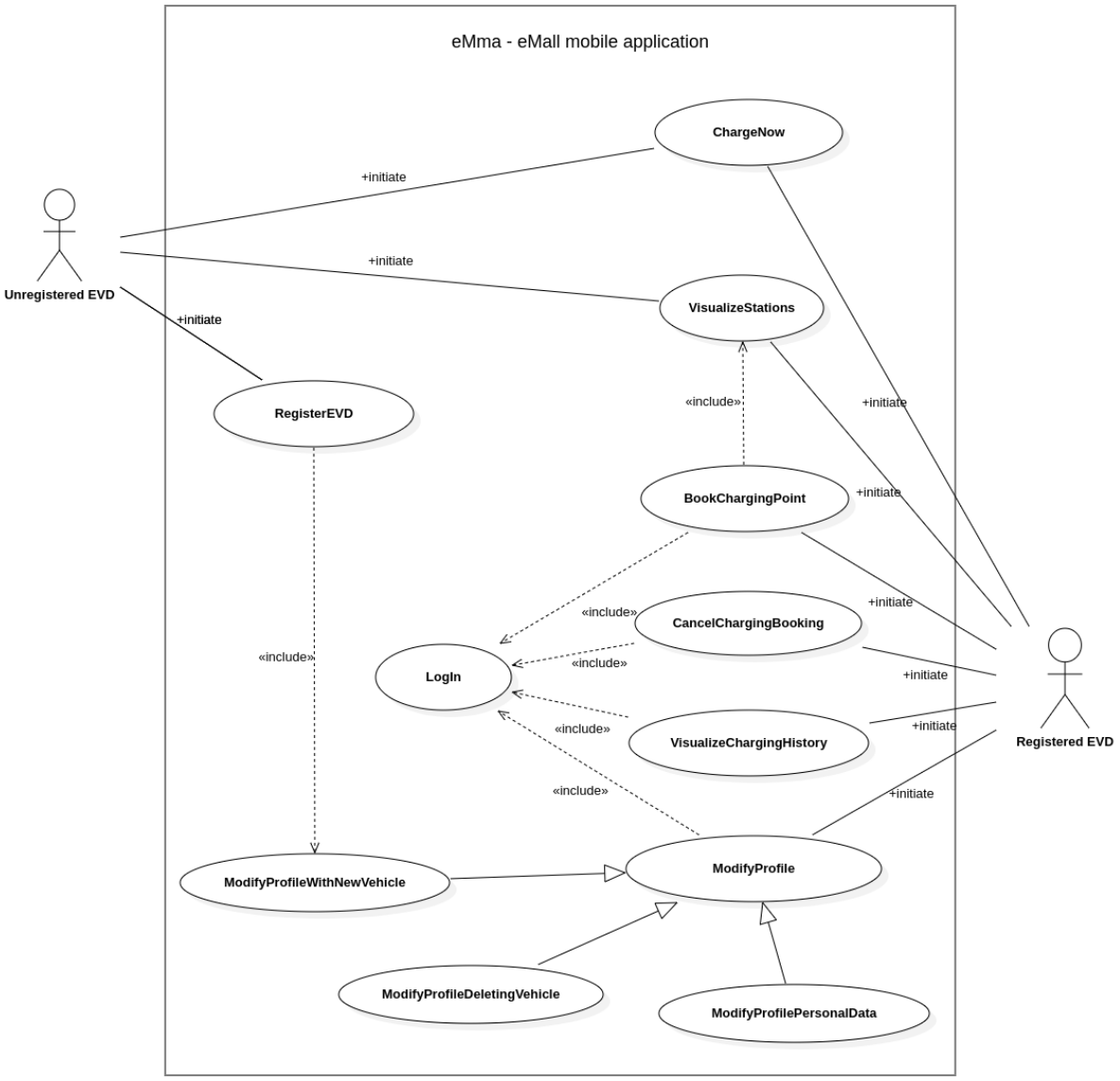

Figure 3.1: Use cases diagram of the registered and unregistered EVD

User registration

Use case name	RegisterEVD
Actor	Unregistered EVD
Entry condition	True
Event flow	1. User opens the eMma mobile application 2. User starts the registering process 3. Users enters his personal data: name, surname, email 4. System check if the email is in the correct format 5. User creates a new password and confirms it a second time 6. System checks the security property of the password 7. System verifies the ownership of the email 8. System asks the user for the consent to use his geographical location 9. User agrees 10. System asks the user to agree to terms of service 11. User agrees 12. «include» "ModifyProfileWithNewVehicle" from step 6 13. The system asks the user to insert the payment details 14. The user inserts the payment details 15. The system checks the correctness of the inserted data 16. System creates new account and logs in the user
Exit condition	A valid account is created and the system logs in the user
Exceptions	a. If the email confirmation process fails eMma shows an error and asks for a new email b. If the inserted password doesn't respect security requirements eMma will ask the user for a new password c. If the user doesn't confirm the ownership of the email the registration process halts and after 10 minutes the system deletes user's details d. If the user doesn't agree to the term of service the registration process halts and after 10 minutes the system deletes user's details e. If at any time the user wants to exits the form, the application allows it, but all the data inserted so far are lost if the operation is not completed

continue on the next page

Special requirements	Every time the user agrees, a new page is loaded in less than 2 seconds, in order for the application to be perceived as fast

Table 3.1: RegisterEVD

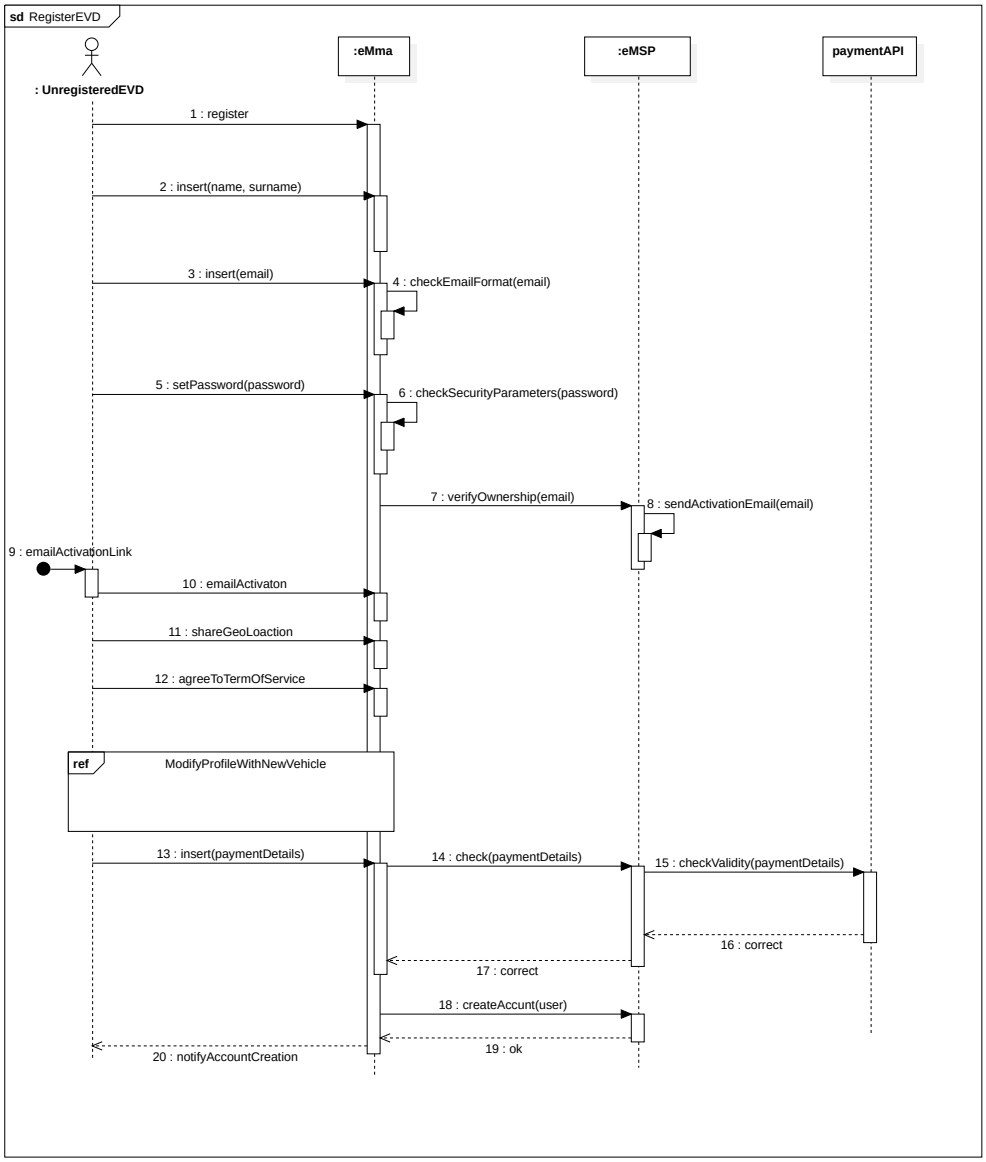

Figure 3.2: RegisterEVD sequence diagram

Log in the system

Use case name	LogIn
Actor	Registered user (EVD or CPO)
Entry condition	The user is not logged in the system and wants to log in
Event flow	1. The user accesses the eMma 2. The eMma shows the log in page 3. The user inserts the log in details 4. The system checks the correctness of the log in details 5. The eMma logs in the user and shows the homepage
Exit condition	The user is logged in and is shown the homepage of the eMma
Exceptions	If the credentials are not correct the user receives an error message and the log in is not successful
Special requirements	After inserting the credentials, the log in details must be checked and the eMall homepage must be shown in less than 2 seconds

Table 3.2: LogIn

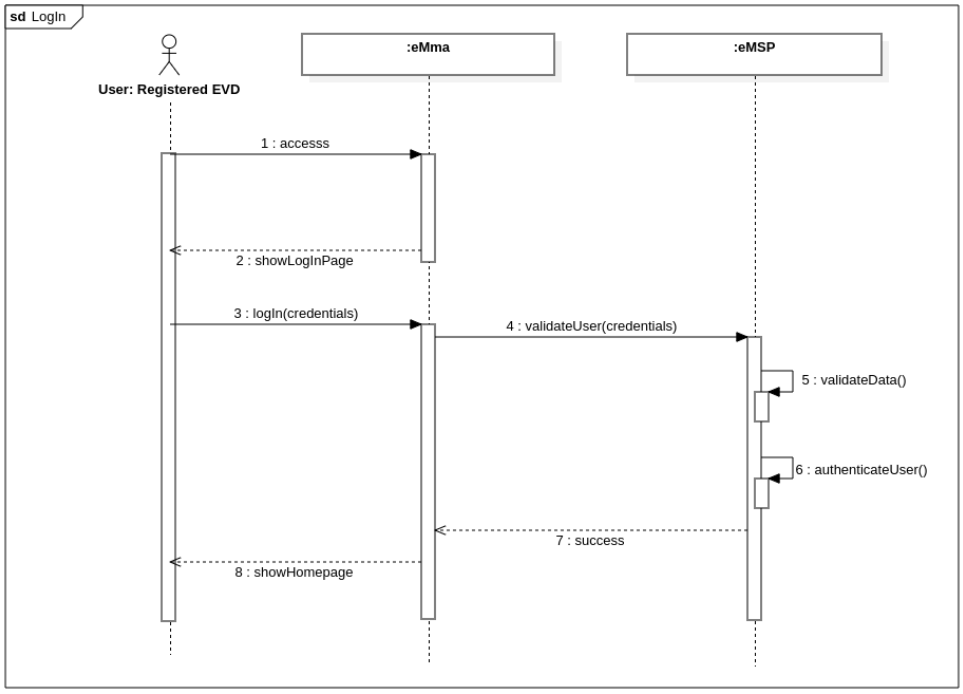

Figure 3.3: LogIn sequence diagram

Booking a charging point

Use case name	BookChargingPoint
Actor	EVD
Entry condition	The EVD is logged in the eMma and on the homepage
Event flow	1. The EVD performs «include»VisualizeStations 2. The EVD selects to book charge from a charging station 3. eMma shows a list of available charging points with the following information: • The type of charging (AC/DC) • The type of the socket (type 1, type 2, CCS, CHAdeMO, etc.) • The charging speed denoted in kW and km/h (km gained per one hour of charge) • The price for kWh • The price for unlocking the socket 4. EVD selects the charging point that suits his needs 5. eMma shows to the EVD a summary of the booking and asks the EVD for confirmation 6. EVD confirms the booking 7. The system processes the booking request 8. eMma notifies the EVD about the success of the booking operation
Exit condition	Booking is registered and the charging point socket changes status to reserved
Exceptions	If the booking confirmation fails, eMma shall notify the user about the occurrence of an error and should bring the view back to the list of available charging points
Special requirements	The system processes the booking request and sends a notification message in less than 5 seconds. Also, at each interaction the response of the eMma is perceived as immediate, taking less than 2 seconds to perform any operation

Table 3.3: BookingChargingPoint

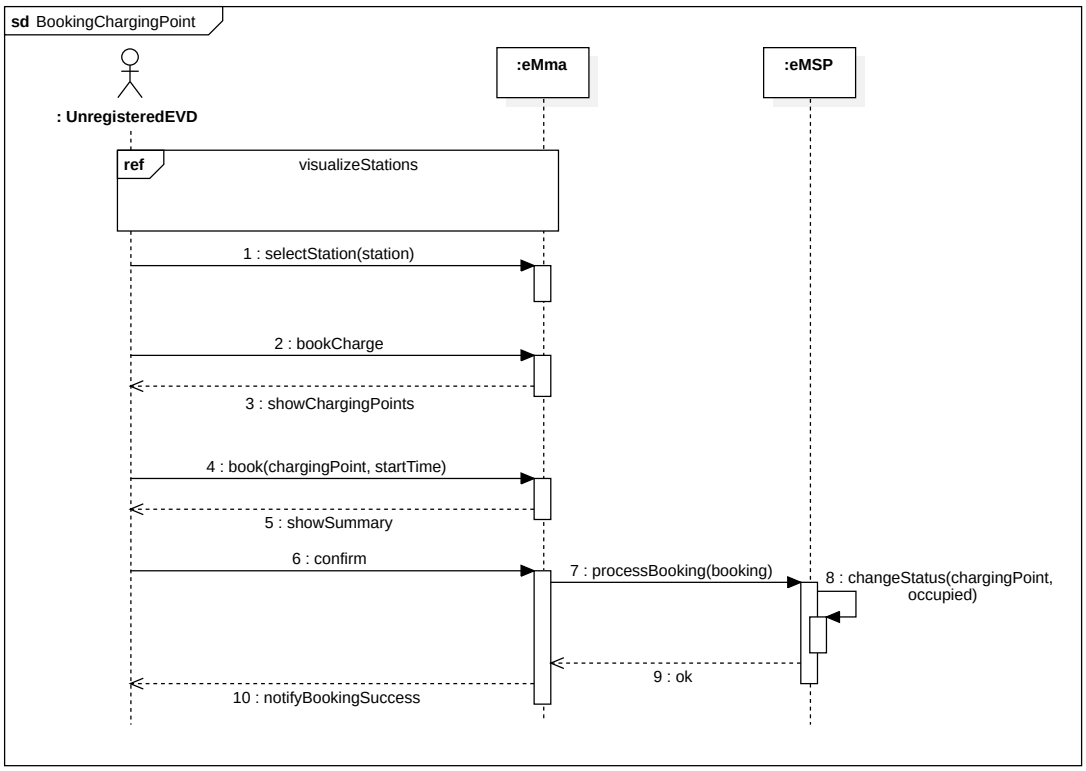

Figure 3.4: BookingCharginPoint sequence diagram

Cancel a charging booking

Use case name	CancelChargingBooking
Actor	EVD
Entry condition	The EVD is logged in the eMma and on the homepage

continue on the next page

Event flow	1. EVD selects imminent charging section 2. eMma shows an entry with details about the imminent charging booking 3. EVD selects the entry 4. eMma allows the EVD the possibility to cancel the charging booking 5. EVD selects to cancel the booking 6. eMma asks for confirmation 7. EVD confirms 8. eMma notifies the EVD about the canceling and sends information to the eMSP about the canceling
Exit condition	The booking is cancelled and the previously reserved charging point socket changes status to free in the booked time-frame
Exceptions	If the booking confirmation fails, eMma shall notify the user about the occurrence of an error and should bring the view back to the list of availables charging points
Special requirements	The EVD is notified about the booking cancellation in less than 2 seconds, for the operation to be perceived as immediate

Table 3.4: CancelChargingBooking

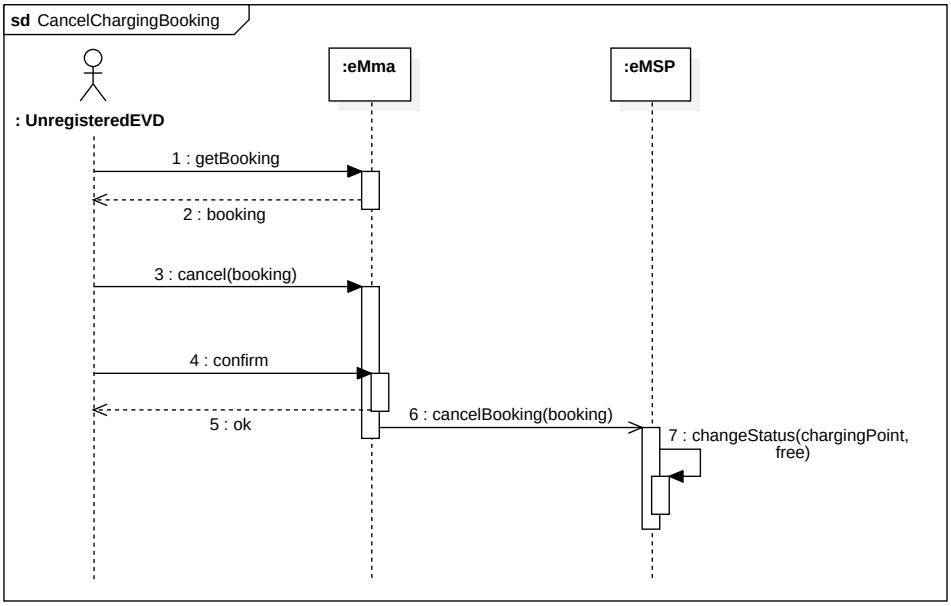

Figure 3.5: CancelChargingBooking sequence diagram

Visualize charging history

Use case name	VisualizeChargingHistory
Actor	Registered EVD
Entry condition	User is logged in eMma

continue on the next page

Event flow	1. EVD selects to view the charging history section 2. eMma shows a view containing details about the imminent charge booking with it's details and a list 3. EVD confirms to see the charging history 4. eMma shows a chronologically ordered list of all the previous charging sessions processed throught the system. Each element on the list contains details about: • the EV involved • the date and time • the location • how long the charging lasted • how many KWh were charged • type of socket used • cost per kWh of the charge • total cost of the charge 5. EVD can select the filter option, to have a different visualization, based on what he is curious about, for example grouping the visualization on the EV involved in the charge 6. eMall shows the list of the previous charging sessions, with the details, according to the selected filter
Exit condition	The EVD exits from the visualization history and the eMall reloads the homepage
Exceptions	a. If the user hasn't got any imminent charges booked, the eMma shows directly the charging history b. If the user hasn't done any charges yet with the system, then the eMma shows an empty page
Special requirements	The eMma shows the history in less than 5 seconds, for every chosen filtering option

Table 3.5: VisualizeChargingHistory

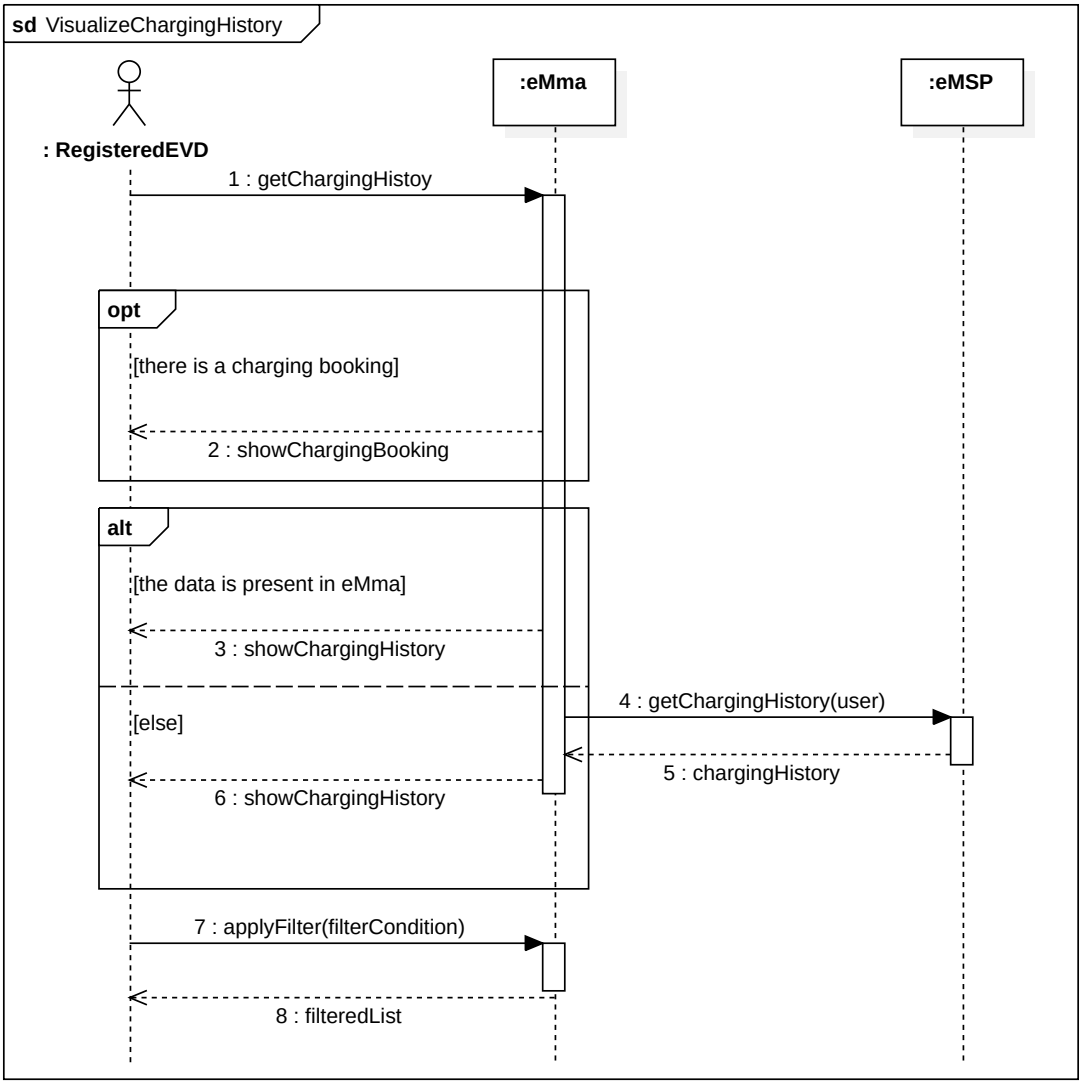

Figure 3.6: VisualizeChargingHistory sequence diagram

Start a charging session

Use case name	ChargeNow
Actor	EVD
Entry condition	The EVD is on the homepage of the eMma and the status of the charging point is free

continue on the next page

Event flow	1. The EVD selects the section used for the immediate charging operation 2. The EVD inserts the eMci code in the eMma and confirms the operation 3. The eMall checks the correctness of the code and unlocks the charging point 4. The system changes the status of the charging point from free to occupied 5. The eMma sends a success notification message to the user, to inform him that the charging point is ready and is waiting for the connector to be plugged in
Exit condition	The EVD plugs in the connector and the system actually starts the charging process
Exceptions	a. If the EVD doesn't insert the correct code, the eMall doesn't unlock the charging point and the eMma returns a warning message, allowing the user to reinsert the code b. If the user doesn't insert the plug in less than 5 minutes, the operation is deleted, and the charging point status return to free
Special requirements	After inserting the code, the eMall does all the necessary checks and changes the status of the charging point in less than 2 seconds, so the service can be perceived as fast and responsive. Also, the system has to start the charging of the EVD in less than 2 seconds, for the same reason

Table 3.6: ChargeNow

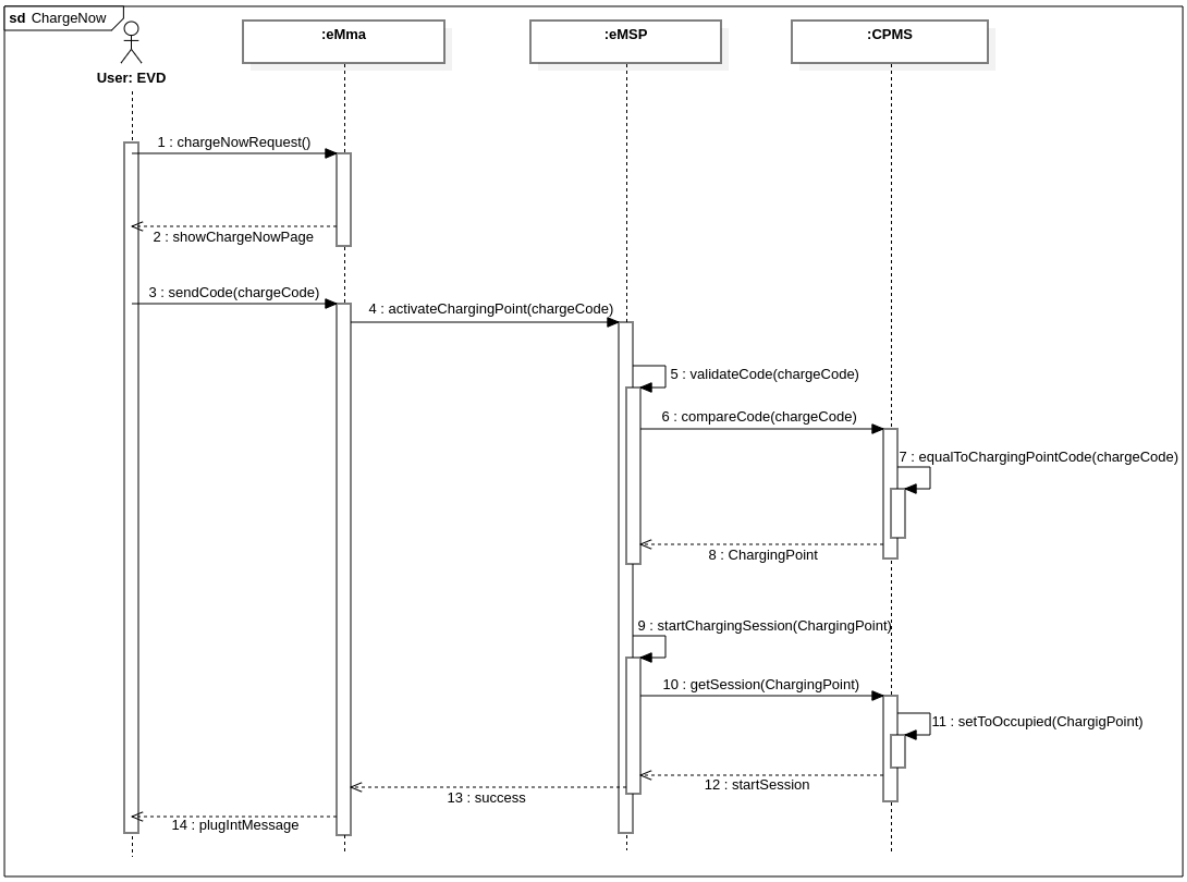

Figure 3.7: ChargeNow sequence diagram

Update profile details adding a new vehicle

Use case name	ModifyProfileWithNewVehicle
Actor	Registered EVD
Entry condition	The EVD is logged in the eMma and on the homepage

continue on the next page

Event flow	1. The EVD enters on his profile page 2. The system shows the profile page with personal information and EV's details 3. The EVD selects the update option 4. The eMma shows a page with different buttons from which to choose the update action 5. The EVD chooses as update action, the one that represents the insertion of a new vehicle 6. The mobile app shows a form with different fields to fill up 7. The EVD inserts the data about the new EV: inserts the type of EV, the supported inlet type of the EV and the presence of the rectifier, the capacity of the battery, the supported power levels and the supported current levels 8. The EVD confirms the operation, submitting the form 9. The eMall saves the data and associates them to the EVD's profile 10. The system reloads the profile page with the new vehicle information
Exit condition	The new EV is associated to the user information already saved in the system, and the eMma reloads the profile page with the new EV's details
Exceptions	a. If the EVD doesn't insert all the mandatory information, after clicking the 'Ok' button the system gives an explicit error message with the missing data, and the user can continue to complete the form b. If at any time the user wants to exit the form, the application allows it, but all the data inserted so far are lost
Special requirements	After the EVD's confirmation, the system has to save the data and reload the profile page in less than 10 seconds

Table 3.7: ModifyProfileWithNewVehicle

Update profile details deleting vehicle

Use case name	ModifyProfileDeletingVehicle
Actor	Registered EVD
Entry condition	The EVD is logged in the eMma and on the homepage
Event flow	1. The EVD enters on his profile page 2. The system shows the profile page with personal information and EV's details 3. The EVD selects the update option 4. The eMma shows a page with different buttons from which to choose the update action 5. The EVD selects the action that represents the operation of deleting a vehicle 6. The mobile app shows a form with the list of the registered vehicles 7. The EVD selects the EV he wants to delete 8. The EVD confirms the operation submitting his choice 9. The eMall retrieves the data of the user and deletes the selected vehicle 10. The system reloads the profile page, in which the deleted vehicle is no more present
Exit condition	The EV is deleted from the data associated to the EVD and the profile page is reloaded
Exceptions	a. If the EVD doesn't select an EV to delete from the list, after clicking the 'Ok' button the system gives an error message, and the user has to select a vehicle or cancel the operation b. If the user wants to exit without selecting an EV, the application allows it, and no vehicle will be deleted from the profile
Special requirements	After the EVD's confirmation, the system has to delete the vehicle from the EVD's data and reload the profile page in less than 3 seconds

Table 3.8: ModifyProfileDeletingVehicle

Update profile modifying personal data

Use case name	ModifyProfilePersonalData
Actor	Registered EVD
Entry condition	The EVD is logged in the eMma and on the homepage
Event flow	1. he EVD enters on his profile page 2. The system shows the profile page with personal information and EV's details 3. The EVD selects the update option 4. The eMma shows a page with different buttons from which to choose the update action 5. The EVD selects the option the represent the operation of updating the profile personal data 6. The mobile app shows a form, already filled up with the personal data 7. The EVD can modify the elements of the form, for example the email, the name, the payment details 8. The EVD, after modifying some data, confirms the operation and submits the form 9. The eMall checks (internally and externally using APIs) and updates the data that have been changed, saving the correct information associated to the EVD's profile 10. The system reloads the profile page with the new personal data
Exit condition	The system saves the new data, discarding the old ones, and the eMma reloads the profile page with the updated personal details
Exceptions	a. If the EVD doesn't change any data of the form, after clicking the 'Ok' button the system reloads the profile page showing all the data present before the operation b. If at any time the user wants to exit the form, the application allows it, but all the data inserted so far are lost, keeping the last personal details c. If the EVD doesn't confirm the operation from the eMma or from the external applications, the updating request is deleted after 10 minutes, and the operation has to be restarted

continue on the next page

Special requirements	After clicking the 'Confirm' button the system has to save the updated data and reload the profile page in less than 10 seconds

Table 3.9: ModifyProfilePersonalData

We only show a sequence diagram for all the use cases regarding the modification of the profile, because even if the data which are modified are different and some actions and checks can change, overall the event flow is the same in all of them. So, the following diagram treats only the action in a general way, representing with it all the possible operations.

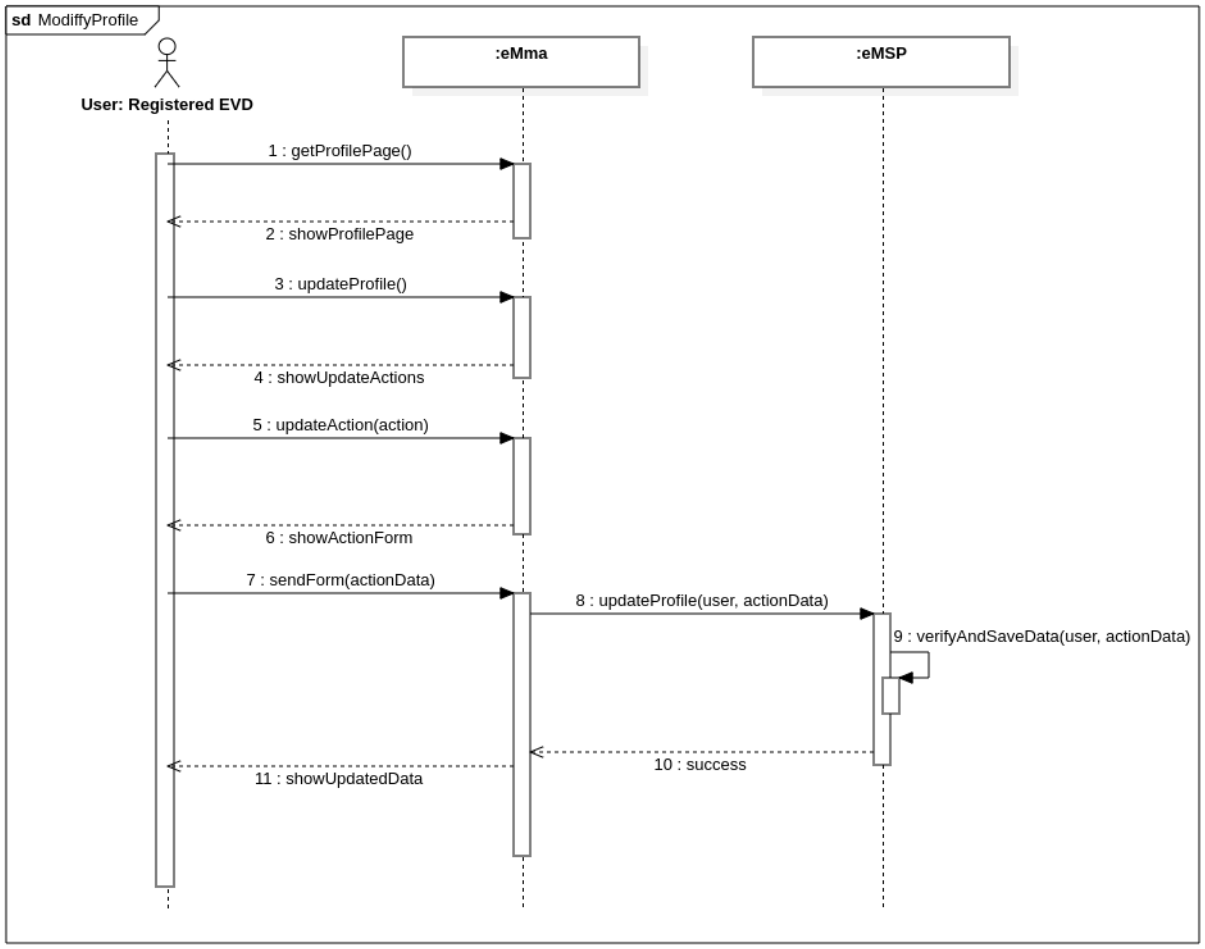

Figure 3.8: ModifyProfile sequence diagram: the use cases ModifyProfileNewVehicle, ModifyProfileDeleteVehicle and ModifyProfilePersonalData are grouped in this diagram

Visualize charging stations on the map

Use case name	VisualizeStations
Actor	Registered EVD or unregistered EVD
Entry condition	The EVD is on the eMma homepage
Event flow	1. On the homepage the eMma shows the map of the territory, based on the location shared by the mobile phone of the EVD 2. The EVD clicks on the map 3. The map shows the charging stations nearby the position on which the EVD clicked 4. The EVD clicks on one of the charging stations presented on the map 5. The eMma shows a page with the information related to the selected charging station: • The name of the charging station • the rating of the charging station • indication about the available sockets types and their number • contact details • address of the charging station • any directions on how to handle the charging process • rating and reviews • the presence of any special offer
Exit condition	The EVD exits the homepage or closes the application
Exceptions	a. If the EVD clicks on the map on an area without charging stations, no charging station will appear on the map b. If at any time the user wants to exit the eMma, the application allows it, and at the following access the action restarts from the homepage
Special requirements	After the EVD clicks on the map, the charging stations are shown in less than 2 seconds. As well, when the EVD clicks on the charging stations the related data are shown in less than 2 seconds, so the application can be perceived as fast and responsive

Table 3.10: VisualizeStations

Terminate charging process

Use case name	TerminateCharging
Actor	eMma
Entry condition	Charging is almost completed
Event flow	1.
Exit condition	The EVD exits the homepage or closes the application
Exceptions	a.
Special requirements	

Table 3.11: TerminateCharging

3.2.3. CPO's use cases

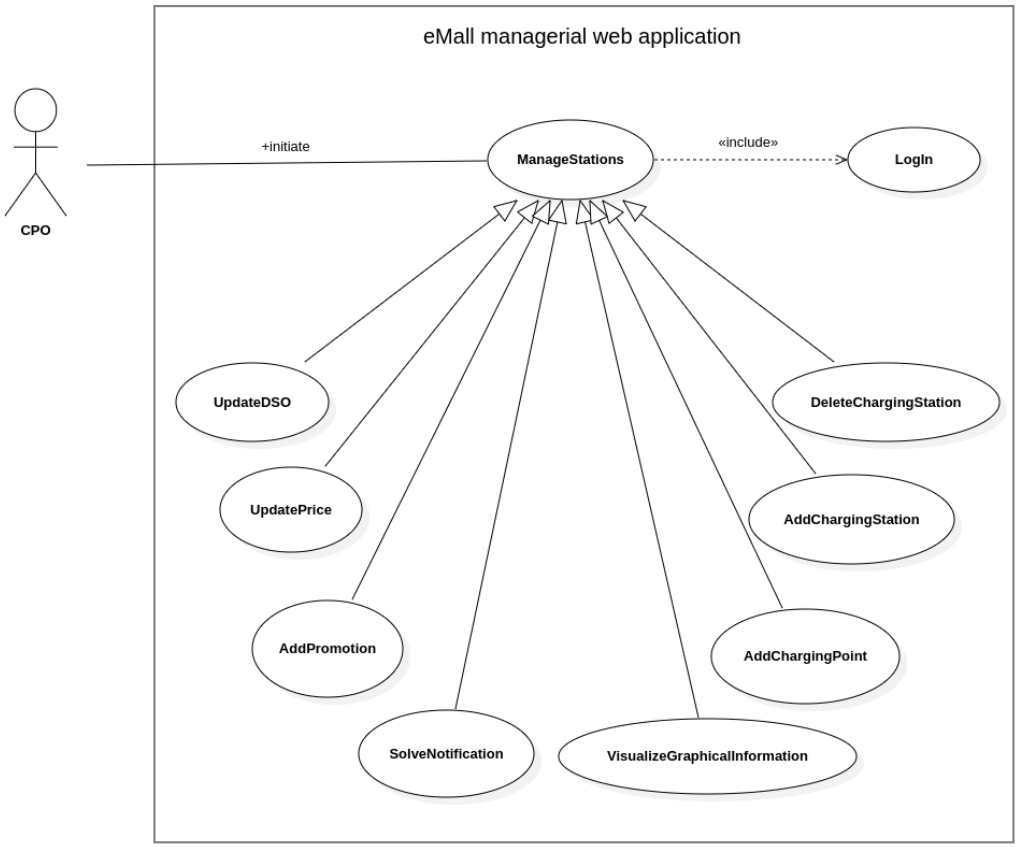

Figure 3.9: Use cases diagram of the CPO

Manage the charging stations

Use case name	ManageStations
Actor	CPO
Entry condition	The CPO is logged in the web application of the eMall and on the homepage

continue on the next page

Event flow	1. On the homepage the eMall shows to the CPO the charging stations associated to the company registration and any notification on the stations 2. The CPO clicks on a charging station or on a notification 3. The system shows a form with the details of the charging station and if there are notifications regarding the station the interested parts of the form are highlighted in red 4. The CPO can click on any part of the form and modify the data of the station 5. The CPO clicks the 'Confirm' button 6. The system checks and saves the new data related to the charging station 7. The system sends a notification message informing of the success of the operation 8. The system loads a page showing the charging station with the new associated information
Exit condition	The CPO closes the page loaded by the system with the updated data of the charging station, returning to the homepage
Exceptions	a. If the CPO doesn't modify anything on the form before submitting it, the eMall sends a message which informs that the data have not been modified, and returns to the homepage b. If at any time the CPO wants to exit, the application allows it, and no changes will be applied if the procedure wasn't completed
Special requirements	After the CPO clicks on the 'Confirm' button, the charging station with the related details is updated and shown on a new page in less than 2 seconds, in order for the application to be perceived as fast and responsive

Table 3.12: ManageStations

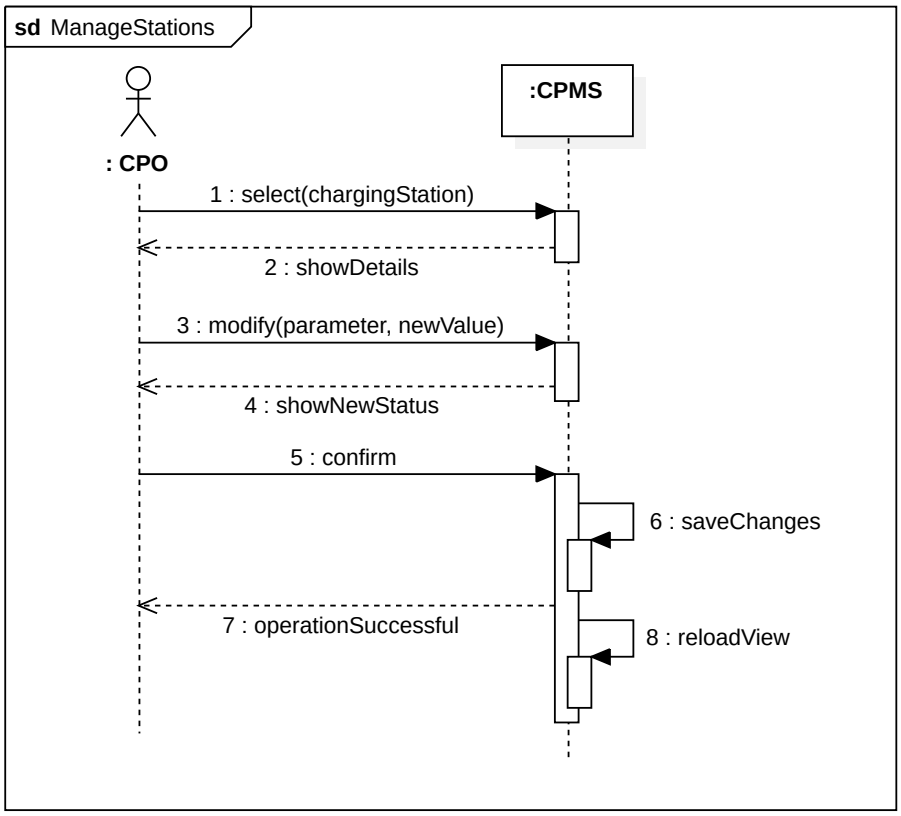

Figure 3.10: ManageStations sequence diagram

The use case about the CPO managing the charging stations, is a generic use case, that can be further analyzed considering the actions that the CPO actually performs on the system, to manage the stations. In the following use cases, we specialize some of these interactions, showing available functionalities of the web application of the eMall. The sequence diagrams of the next use cases are similar to the one reported for the general interaction of the CPO with the system, so they will not be added to this document.

Update the DSO of a charging station

Use case name	UpdateDSO
Actor	CPO
Entry condition	The CPO is logged in the web application of the eMall and on the homepage

continue on the next page

Event flow	1. On the homepage the eMall shows to the CPO the charging stations associated to the company registration and any notification on the stations 2. The CPO clicks on a charging station 3. The system shows a form with the details of the charging station 4. The CPO clicks on the 'DSO' cell present in the form 5. The system shows a new sub-page with the available DSOs and the respective information: for each DSO the page shows the energy resources, their capacity and their prices 6. The CPO selects the DSO and the energy source he wants to use for the charging station 7. The CPO clicks on the 'Ok' button 8. The web application returns to the form of the charging station with the new selected DSO information replacing the previous one 9. The CPO clicks the 'Confirm' button 10. The system checks and saves the new data related to the charging station 11. The system sends a notification message informing of the success of the operation 12. The system loads a page showing the charging station with the new associated information
Exit condition	The CPO closes the page loaded by the system with the updated data of the charging station, returning to the homepage
Exceptions	a. If the CPO doesn't modify anything on the form before submitting it, the eMall sends a message which informs that the data have not been modified, and returns to the homepage b. If at any time the CPO wants to exit, the application allows it, and no changes will be applied if the procedure wasn't completed
Special requirements	After the CPO clicks on the 'Confirm' button, the charging station with the related details is updated and shown on a new page in less than 2 seconds, in order for the application to be perceived as fast and responsive

continue on the next page

Table 3.13: UpdateDSO

Update the price of a charging station

Use case name	UpdatePrice
Actor	CPO
Entry condition	The CPO is logged in the web application of the eMall and on the homepage
Event flow	1. On the homepage the eMall shows to the CPO the charging stations associated to the company registration and any notification on the stations 2. The CPO clicks on a charging station 3. The system shows a form with the details of the charging station 4. The CPO changes the price of the charging station from the form 5. The CPO clicks the 'Confirm' button 6. The system checks and saves the new data related to the charging station 7. The system sends a notification message informing of the success of the operation 8. The system loads a page showing the charging station with the new associated information
Exit condition	The CPO closes the page loaded by the system with the updated data of the charging station, returning to the homepage

continue on the next page

Exceptions	a. If the CPO doesn't modify anything on the form before submitting it, the eMall sends a message which informs that the data have not been modified, and returns to the homepage b. If the price doesn't respect a level fixed by the company policy, the system sends an error message, informing the CPO that the price is to high or too low, and the CPO has to modify again the form, otherwise no change will be applied c. If at any time the CPO wants to exit, the application allows it, and no changes will be applied if the procedure wasn't completed
Special requirements	After the CPO clicks on the 'Confirm' button, the charging station with the related details is updated and shown on a new page in less than 2 seconds, in order for the application to be perceived as fast and responsive

Table 3.14: UpdatePrice

Add a promotion for the charging station

Use case name	AddPromotion
Actor	CPO
Entry condition	The CPO is logged in the web application of the eMall and on the homepage

continue on the next page

Event flow	1. On the homepage the eMall shows to the CPO the charging stations associated to the company registration and any notification on the stations 2. The CPO clicks on a charging station 3. The system shows a form with the details of the charging station 4. The CPO sets a promotion for the charging station, selecting it from the ones available, for example from a combo box 5. The CPO clicks the 'Confirm' button 6. The system checks and saves the data related to the new promotion set for the charging station 7. The system sends a notification message informing of the success of the operation 8. The system loads a page showing the charging station with the new associated information
Exit condition	The CPO closes the page loaded by the system with the updated data of the charging station, returning to the homepage
Exceptions	a. If the CPO doesn't modify anything on the form before submitting it, the eMall sends a message which informs that the data have not been modified, and returns to the homepage b. If the promotion doesn't respect the parameters fixed by the company policy, the system sends an error message, informing the CPO that the promotion is not acceptable, and the CPO has to modify again the form, otherwise no change will be applied c. If at any time the CPO wants to exit, the application allows it, and no changes will be applied if the procedure wasn't completed
Special requirements	After the CPO clicks on the 'Confirm' button, the charging station with the related details is updated and shown on a new page in less than 2 seconds, in order for the application to be perceived as fast and responsive

Table 3.15: AddPromotion

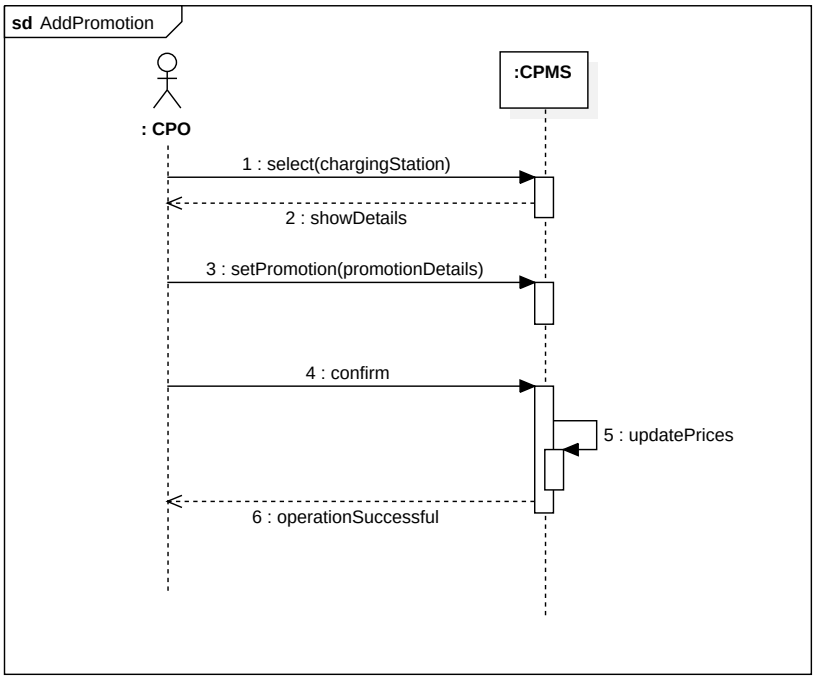

Figure 3.11: AddPromotion sequence diagram

Delete a charging station

Use case name	DeleteChargingStation
Actor	CPO
Entry condition	The CPO is logged in the web application of the eMall and on the homepage

continue on the next page

Event flow	1. On the homepage the eMall shows to the CPO the charging stations associated to the company registration and any notification on the stations 2. The CPO clicks on a charging station 3. The system shows a form with the details of the charging station 4. The CPO clicks the 'Delete charging station' button 5. The system sends a warning message and asks for confirmation, because this is a delicate operation 6. The CPO clicks the 'Confirm' button 7. The system deletes the charging station from the information related to the company 8. The system sends a notification message informing of the success of the operation 9. The system reloads the homepage, without the deleted charging station
Exit condition	The CPO closes the page loaded by the system with the updated data of the charging station, returning to the homepage
Exceptions	a. If the CPO doesn't confirm the cancellation of the charging station, after 5 minutes, the system reloads the homepage, without applying any change b. If at any time the CPO wants to exit, the application allows it, and no changes will be applied if the procedure wasn't completed
Special requirements	After the CPO clicks on the 'Confirm' button, the charging station will be deleted from the information related to the company and the homepage will be shown in less than 2 seconds, in order for the application to be perceived as fast and responsive

Table 3.16: DeleteChargingStation

Add a charging station

Use case name	AddChargingStation
Actor	CPO
Entry condition	The CPO is logged in the web application of the eMall and on the homepage
Event flow	1. The CPO selects the button 'Add charging station' 2. The system shows a form to fill up with the data related to the new charging station: the code of the station, the position, the charging points, the available sockets, the prices, the batteries of the station and other details for each charging point 3. The CPO completes the form and clicks the 'Confirm' button 4. The system checks and saves the new data related to the charging station 5. The system sends a notification message informing of the success of the operation 6. The system loads a page showing the new charging station with the associated information
Exit condition	The CPO closes the page loaded by the system with the updated data of the charging station, returning to the homepage
Exceptions	a. If the CPO doesn't fill up all the mandatory data of the form before submitting it, the eMall sends a message, which informs what other data are required, and returns to the form that the CPO can continue to complete b. If at any time the CPO wants to exit, the application allows it, and no changes will be applied if the procedure wasn't completed
Special requirements	After the CPO clicks on the 'Confirm' button, the new charging station with the related details is added to the data related to the company, and shown on a new page in less than 2 seconds, in order for the application to be perceived as fast and responsive

Table 3.17: AddChargingStation

Add a charging point

Use case name	AddChargingPoint
Actor	CPO
Entry condition	The CPO is logged in the web application of the eMall and on the homepage
Event flow	1. On the homepage the eMall shows to the CPO the charging stations associated to the company registration and any notification on the stations 2. The CPO clicks on a charging station 3. The system shows a form with the details of the charging station 4. The CPO clicks the 'Add charging point' button 5. The system shows a new form to fill up with the data related to the new charging point: the code of the charging point, the available sockets, the maximum and minimum output capacity and other details 6. The CPO completes the form and clicks the 'Confirm' button 7. The system checks and saves the new charging point related to the charging station 8. The system sends a notification message informing of the success of the operation 9. The system loads a page showing the charging stations with the new associated charging point
Exit condition	The CPO closes the page loaded by the system with the updated data of the charging station, returning to the homepage
Exceptions	a. If the CPO doesn't fill up all the mandatory data of the form before submitting it, the eMall sends a message, which informs what other data are required, and returns to the form, that the CPO can continue to complete b. If at any time the CPO wants to exit, the application allows it, and no changes will be applied if the procedure wasn't completed

continue on the next page

Special requirements	After the CPO clicks on the 'Confirm' button, the new charging point of the station, with the related details, is added to the data related to the company, and shown on a new page in less than 2 seconds, in order for the application to be perceived as fast and responsive

Table 3.18: AddChargingPoint

Visualize graphical information about the charging stations

Use case name	VisualizeGraphicalInformation
Actor	CPO
Entry condition	The CPO is logged in the web application of the eMall and on the homepage
Event flow	1. On the homepage the eMall shows to the CPO the charging stations associated to the company registration and any notification on the stations 2. To see more specific information the CPO click on 'Visualization' button 3. The system shows a list with all the elements that can be visualized in form of a graph regarding the managed charging stations 4. The CPO selects the information that is interested in, for example the peak hours and energy requests 5. The CPO clicks the 'Confirm' button 6. The system processes the chosen data regarding all the stations 7. The system loads a new page in which shows some graphical representations of the data
Exit condition	The CPO closes the page loaded by the system with the graphical visualizations of the data of the charging stations, returning to the homepage

continue on the next page

Exceptions	a. If the CPO doesn't select anything on the list before submitting it, the eMall sends a message which informs that nothing can be visualized, and returns to the homepage b. If at any time the CPO wants to exit, the application allows it, and no changes will be applied if the procedure wasn't completed
Special requirements	After the CPO clicks on the 'Confirm' button, the graphical visualizations are shown on a new page in less than 2 minutes, in order for the system to have enough time to process a lot of the present data, and show an approximated solution in an acceptable time

Table 3.19: VisualizeGraphicalInformation

Solve the notification regarding a charging station

Use case name	SolveNotification
Actor	CPO
Entry condition	The CPO is logged in the web application of the eMall and on the homepage

continue on the next page

Event flow	1. On the homepage the eMall shows to the CPO the charging stations associated to the company registration and any notification on the stations 2. The CPO clicks on a notification 3. The system shows a form with the details of the charging station, with the data related to the notification highlighted in red 4. The CPO clicks on the battery associated to the charging station, that is highlighted in red 5. When clicking on a red element, the system shows a notification message, informing the CPO about the problem or the changes undergone to that element of the station 6. The CPO changes the details of the battery, selecting a DSO from which to acquire energy, because in this case the notification was about the battery being empty 7. The CPO clicks the 'Confirm' button 8. The system checks and saves the new data related to the charging station 9. The system sends a notification message informing of the success of the operation and deletes the notification message present before the operation 10. The system loads a page showing the charging station with the new associated information
Exit condition	The CPO closes the page loaded by the system with the updated data of the charging station, returning to the homepage

continue on the next page

Exceptions	a. If the CPO doesn't modify anything on the form before submitting it, the eMall sends a message which informs that the data have not been modified, and returns to the homepage without deleting the notification message from the system b. If the CPO changes the form, but not the details regarding the notification, the eMall will apply the operation, after the submitting of the form, but will maintain the notification message in the system c. If at any time the CPO wants to exit, the application allows it, and no changes will be applied if the procedure wasn't completed
Special requirements	After the CPO clicks on the 'Confirm' button, the charging station with the related details is updated and shown on a new page in less than 2 seconds, in order for the application to be perceived as fast and responsive

Table 3.20: SolveNotification

3.2.4. Requirements

Requirement	Description
R1	The system shall allow an unregistered EVD to register an account
R2	The system shall allow a registered EVD to insert data about his EVs
R3	The system shall allow a registered EVD to update EV's details
R4	The system shall allow a registered EVD to add new EVs
R5	The system shall allow a registered EVD to delete an EV
R6	The system shall allow a registered EVD to insert personal data (name, surname, email, payment details)
R7	The system shall allow a registered EVD to update personal data
R8	The system shall allow an EVD to view the charging stations on the map
R9	The system shall allow an EVD to view relevant data about the charging stations
R10	The system shall allow an EVD to view relevant data about a specific charging point
R11	The system shall allow an EVD to view the prices of the charging points of the stations
R12	The system shall allow an EVD to view the special offers of the charging points of the stations
R13	The system shall allow a registered EVD to review a charging station
R14	The system shall allow an EVD to share his location through the GPS module of his mobile phone
R15	The system shall allow an EVD to choose the area in which to visualize the charging stations on the map, if different from the actual GPS location
R16	The system by default must show on the homepage the charging stations on the map territory, centered on the EVD's location
R17	The system shall allow a registered EVD to book a charge from an available charging point in a charging station
R18	The system shall allow a registered EVD to choose the time-frame (date and time) in which to book a charge
R19	The system shall allow a registered EVD to cancel a charging point booking
R20	The system shall allow a registered EVD to visualize his profile data

R21	The system shall allow a registered EVD to view the charging history
R22	The system shall allow a registered EVD to choose the criteria of the visualization of the charging history
R23	The system shall allow an EVD to start a charging session
R24	The system shall allow a registered EVD to insert the charging point code, in order to start a charging session
R25	The system shall allow an EVD to pay for the charging service
R26	The system must allow the registered EVD to log in
R27	The system must allow the CPO to log in with the company credentials
R28	The system must allow the EVD to access the mobile application, eMma, with or without an account
R29	The system must store the charging history: previous charges and bookings related to the specific EV
R30	The system must collect electric energy data from the DSOs
R31	The system must collect the data about the charging stations from the CPOs
R32	The system must notify the EVD with a specific message, that clarifies the problem, if an error occurs (login error, update error, payment error, ecc.)
R33	The system must notify the EVD with a success message if the operation terminated without errors (successful registration, successful profile modification, ecc.)
R34	The system must ask for EVD's consent to use the location information given by the GPS module
R35	The system must ask the EVD, during registration, to agree to the terms of service
R36	The system must check for the correctness of the inserted data (login details, charging point code, ecc.)
R37	The system must show the available charging points to book
R38	The system must show for each available charging point the relevant information (type of charging, type of socket, charging speed, price, special offers, etc.)
R39	The system must show a summary of the successful booking operation
R40	The system must change the status of the charging point during charging
R41	The system must unlock the charging point if the code is correct

R42	The system shall allow the CPO to view the charging stations
R43	The system shall allow the CPO to view any notification regarding the charging stations
R44	The system shall allow the CPO to update a charging station
R45	The system shall allow the CPO to delete a charging station
R46	The system shall allow the CPO to add a charging station
R47	The system shall allow the CPO to delete a charging point from a charging station
R48	The system shall allow the CPO to add a charging point to a charging station
R49	The system shall allow the CPO to set the price of the charging point
R50	The system shall allow the CPO to set a special offer for the charging point
R51	The system must check the correctness of the data inserted by the CPO
R52	The system must store the data of the charging stations
R53	The system must notify the CPO with a specific message if an error occurs during an operation
R54	The system must notify the CPO with a success message if the operation terminates without errors
R55	The system shall allow the CPO to select a charging station
R56	The system shall allow the CPO to view the DSO's updated prices for the energy sources
R57	The system shall allow the CPO to view the DSO's special offers for the energy sources
R58	The system shall allow the CPO to change the DSO of a charging station
R59	The system shall allow the CPO to choose the DSO's energy source for the charging station
R60	The system shall allow the CPO to update the price of a charging point
R61	The system shall allow the CPO to set a special offer for the charging station
R62	The system shall allow the CPO to select some criteria to graphically visualize aspects of the charging stations
R63	The system must show a graphical representation of some aspects of the charging stations

R64	The system shall be interoperable with systems made by other companies
R65	The system shall allow the registered EVD to view his upcoming bookings
R66	The system shall allow the CPO to select a notification to solve it

Table 3.21: Requirements

3.2.5. Mapping on requirements

Use case	**Requirement**
RegisterEVD	R1, R2, R6, R14, R26, R28, R32, R33, R34, R35, R36
LogIn	R1, R26, R28, R32, R36
BookChargingPoint	R8, R9, R10, R11, R12, R14, R15, R16, R17, R18, R26, R28, R31, R32, R33, R36, R37, R38, R39
CancelChargingBooking	R19, R26, R28, R33, R65
VisualizeChargingHistory	R21, R22, R26, R28, R29, R65
ChargeNow	R23, R26, R28, R32, R33, R36, R40, R41, R64
ModifyProfileWithNewVehicle	R2, R4, R20, R26, R28, R32, R33
ModifyProfileDeletingVehicle	R5, R20, R26, R28, R32, R33
ModifyProfilePersonalData	R7, R20, R26, R28, R32, R33
VisualizeStations	R8, R9, R10, R11, R12, R14, R15, R16, R26, R28
TerminateCharging	
GetDataFromDSO	
UpdateDSO	R27, R30, R31, R42, R43, R44, R51, R52, R53, R54, R55, R56, R57, R58, R59, R64
UpdatePrice	R27, R30, R31, R42, R43, R44, R49, R51, R52, R53, R54, R55, R56, R57, R60, R64
AddPromotion	R27, R30, R31, R42, R43, R44, R50, R51, R52, R53, R54, R55, R56, R57, R61, R64
SolveNotification	R27, R30, R31, R42, R43, R51, R52, R53, R54, R64, R66
VisualizeGraphicalInformation	R27, R30, R31, R62, R63, R64

AddChargingPoint	R27, R30, R31, R42, R43, R44, R48, R49, R50, R51, R52, R53, R54, R55, R64
AddChargingStation	R27, R30, R31, R42, R43, R46, R48, R49, R50, R51, R52, R53, R54, R56, R57, R59, R61, R64
DeleteChargingStation	R27, R30, R31, R42, R43, R45, R53, R54, R55, R64

Table 3.22: Mapping on requirements

3.2.6. Mapping on goals

Goal	**Domain assumptions**	**Requirements**
G1	D1, D2, D3, D4, D6	R8, R9, R10, R11, R12, R14, R15, R16, R26, R28
G2	D1, D2, D3, D4, D6	R8, R9, R10, R11, R12, R14, R15, R16, R26, R28
G3	D1, D2, D3, D4, D6	R8, R9, R10, R11, R12, R14, R15, R16, R26, R28
G4	D1, D2, D3, D4, D6, D8	R1, R2, R6, R8, R9, R10, R11, R12, R14, R15, R16, R17, R18, R26, R28, R31, R32, R33, R34, R35, R36, R37, R38, R39
G5	D1, D2, D3, D4, D6	R23, R26, R28, R32, R33, R36, R40, R41, R64
G6	D1, D2, D3, D4, D7	
G7	D5, D6, D9	R27, R30, R31, R42, R43, R44, R51, R52, R53, R54, R55, R56, R57, R58, R59, R64
G8	D5, D6, D9	R27, R30, R31, R42, R43, R44, R49, R51, R52, R53, R54, R55, R56, R57, R60, R64
G9	D5, D6, D9	R27, R30, R31, R42, R43, R44, R50, R51, R52, R53, R54, R55, R56, R57, R61, R64
G10	D5, D6, D9	
G11	D5, D6, D9	

Table 3.23: Mapping on goals

3.3. Performance Requirements

The scope of this section is to specify both the static and the dynamic numerical requirements placed on the software or on human interaction with the software as a whole.

Static numerical requirements:
By account of the European Automotive Manufacturer's Association (ACEA) in 2021 the car fleet of the whole EU amounted to 242 million of vehicles and among those 1.1% are electrically-chargeable vehicles, in other words in EU roads there are roughly 2.7 million vehicles with a plug. The number of new registered vehicles amounted to 9.7 million, of which 18%, roughly 1.7 million, (it was 10% in 2020) were EVs or plug-in hybrids. This shows that the adoption of electrically-chargeable vehicles is increasing and so the need to use systems to manage the charging process. With these numbers in mind we can be cautious and consider that roughly a number of 15 million electrically-chargeable vehicles will be circulating in the EU roads in the next year, so we will base our analysis on this number. Obviously the results can be factored by the market share the company believes it will cover.

We can assume that to each EV corresponds a user and that 1KB of data each is sufficient to record the personal data of the user and the information regarding the parameters of the EVD:

$$userData = 15 * 10^6 * 1KB = 15GB$$

$$evData = 15 * 10^6 * 1KB = 15GB$$

There is also the information regarding the charging stations and all their parameters, like charging points, prices, location, rating, reviews etc. In consideration of the low number of charging stations, that we assume to be no more than 100,000, and the generous amount of data needed to describe them we can allocate 1MB for charging stations which gives us:

$$chargingStationsData = 10 * 5 * 1MB = 100GB$$

Furthermore we have to keep track of the charging history, in particular each element of the list must contain:

- EV identifier
- charging station identifier
- timestamp of charging start
- timestamp of charging end

- price per kWh paid
- kWh recharged

1KB is large enough to encode this information. Limiting the length of the charging history to 100 elements we get a maximum storage need of:

$$maxHistoryDataPerUser = 100 * 1KB = 100KB$$

$$maxHistory = 15 * 10^6 * maxHistoryDataPerUser = 1.5TB$$

Which is perfectly manageable by modern system. But we also point out that this information can also be stored not in a central server, but in a distributed manner in the mobile devices of the users.

For the CPMS we have to keep track of all the information generated by the charging points for each charging station that it manages. We can assume that 1KB of data is enough to store all the information of charger and that a charging station has on average 10 chargers, so:

$$chargersDataAvg = 10 * 1KB = 10KB$$

We also have to keep track of the CPO authentication details but the memory footprint of this data is negligible so it will be not considered.

Next we have to keep track also of the data that we get from the DSOs. The electricity grid market is characterized by high cost of this investment, which means that there are only a few organization that operate in this market, and we assume 100 being generous. Assuming also 1MB of data for each DSO we get:

$$dsoData = 100 * 1MB = 100MB$$

Dynamical performance requirements:
Considering that the operation of charging an EV is not an operation that happens that often, we assume that at a certain point of time only 2-5% of the user are actively using the eMSP at the same moment. Since the speed of processing a request is not vital to the functionality of the system, we do not impose any hard constraint on the time a request should be processed, only that it shall be done on the order of 5-20 seconds to give the user the impression of good usage experience.

Regarding eMma, we'd like to point out that it's the component that will influence the most the perceived usage experience, so for all the operations that do not involve request to the eMSP server eMma shall process the request in less than 2 seconds.

eMci, like eMma, shall process the transaction in less than 2 seconds for all those operations that do not require communication with the rest of the eMSP.

The CPMS dynamical performance requirements regarding the changing of the cost of charging, the usage of energy stored in batteries, and concluding purchase agreements with a new DSO may be fundamental for the economy of the business, so the operations regarding this aspect of managing the charging station shall be processed as soon as possible, in sub-second time after the CPO confirms the transaction. Furthermore the time required by the CPMS to unlock and start the charging process from a charger shall not be greater than 10s.

3.4. Design Constraints

In the section we specify constraints on the system design imposed by external standards, regulatory requirements, or project limitations.

3.4.1. Standard compliance

The eMall system has to store all different kind of personal data provided by the users, therefore this data must be handled in accordance to the privacy regulation of the countries in which the organization will operate.

3.4.2. Hardware limitations

eMci is a subpart of the system that will run on embedded system, so the limited computing resources of this kind of devices must be kept in mind in the design the architecture of the system. Furthermore, the screen component on this devices tend to be of small size and resolution, meaning that the GUI for this component must be simple. The same can be said for the eMma, which will run on mobile devices (smartphones or tablets) and an adequate user experience shall be granted to all users, even those with low-end devices. For the part of the eMSP that will handle the transactions coming form eMma and eMci and for the CPMS part of the system no particular hardware constraints are to be considered since they'll run as server process in a general purpose computer.

3.4.3. Any other constraints

No any other specific constraint is required.

3.5. Software System Attributes

3.5.1. Reliability

It goes without saying that the system shall achieve an adequate level of reliability. In particular some functionalities of the system shall present stronger reliability than others. In particular the key functionalities of the system like login, booking, charging and payment must exhibit a fairly reliable behaviour. The software should thoroughly tested before delivery to remove defect that would cause the system to crash. The system shall be implemented as loosely coupled as possible and the component that implement the key functionalities described above should be independent from the other part of the system. In this manner failures of secondary components shall not affect the correct working of the main ones. From the hardware point of view, to increase reliability, redundant and parallel architecture should be implemented when possible. In this way, a fail-over mechanism can be implemented if one of the machines fails.

3.5.2. Availability

Even though periods of downtime can cause disruption to clients, these are not critical, so interval of time where the service is not available may be tolerated. Still, a not working service may damage the image of the company and the business itself, thus availability is an aspect that should not be disregarded. The system shall, then, be build with replication and fault tolerance in mind. The initial availability target that the designer should focus on achieving is two nines, meaning that the system should be downtime roughly for no more than 4 days in a year. Given that the devices in which the system will be running are fairly common and inexpensive, the system shall be build with extensions in mind by means of replication and fault tolerance. Following this approach, the availability score can be adjusted based on customer needs.

3.6. Security

eMall stores a lot of user sensitive data and these must be protected in the event of malicious attacks from external agents. To guarantee the protection of the data provided by the users, common security practices shall followed. In particular, sensitive data shall

not be stored in plain-text but a it shall be encrypted using a secure encryption algorithm like SHA2. Furthermore, a lot of communication occurs through the internet by different components of our system. we can't exclude the possibility of potential sniffing attacks on the communication network, so all communication shall be conducted with adequately secure communication protocols like `https` or `TLS`.

3.6.1. Maintainability

The charging facilities and management is a relative new and dynamically changing environment. This means that the system shall be build to facilitate maintenance and extensions. To achieve this goal the designer and the development team should adhere to the commonly known patterns and principles that guide to a modular, lowly coupled and high coherent system. It goes without saying that, the whole system needs to be thoroughly documented, both by comments on the code and by providing a specification document.

3.6.2. Portability

As we previously discussed, we mentioned that that the eMma component of the system shall be run on mobile devices. currently there are two main platforms that dominate the market in this sector, but new competitors may arise in the future, so the option building this part of the system with tools that will guarantee it's portability to new platforms must be considered if cost-effective. While for the development of the eMci component, using tools that guarantee the portability will be mandatory due to high variety of embedded platforms present in the market. The last note is about the eMSP server part and CPMS modules. Since these are thought to be run on general purpose computers and our system hasn't any distribution requirement portability isn't really need for these sub-systems.

4 | Formal analysis using Alloy

In this final section we present a formal specification of some of the requirements explained before.

```
open util/ordering[DateTime]

sig DateTime{}

// some actors in the system
abstract sig User {}

//We don't consider dangling elements
sig Email{}{
    this in EVD.email
}
sig Password{}{
    this in EVD.password
}
sig Location{}{
    this in ChargingStation.location
}
sig History{
}{
    this in EVD.chargingHistory
}

sig EVD {
    evs: some EV,
    email: one Email,
    password: one Password,
    chargingHistory: one History
```

```
}

sig DSO {}{
    this in ChargingStation.dso
}

abstract sig Socket {}
one sig Type1, Type2, Chademo extends Socket{}

sig EV {
    socket: one Socket,
}{
    this in EVD.evs
}

sig ChargingPoint {
    socket: some Socket,
    connectedEV: lone EV
}{
    EV.socket in socket
    this in ChargingStation.chargingPoints
}

sig ChargingStation {
    chargingPoints: some ChargingPoint,
    location: one Location,
    dso: one DSO
}{
    this in CPO.chargingStations
}

sig CPO extends User{
    chargingStations: some ChargingStation
}

sig Booking {
    ev: EV,
```

```
    cs: ChargingStation,
    cp: ChargingPoint,
    start: DateTime,
    end: DateTime
}{
    // Only Registerd User can book
    ev in EVD.evs &&
    cp in cs.chargingPoints
    lt[start, end]
}

/*********** FACTS  ************/

fact eachChargingToOnlyOneChargingStation{
    all disj x, y : ChargingStation |
        #(x.chargingPoints & y.chargingPoints) = 0
}

fact eachCStoOnlyOneCPO{
    all disj x, y : CPO |
        #(x.chargingStations & y.chargingStations) = 0
}

fact eachEVtoOnlyOneEVD {
    all disj x, y : EVD |
        #(x.evs & y.evs) = 0
}

fact eachEVConnectedToOneChargingPoint{
    all disj x, y: ChargingPoint |
        #(x.connectedEV & y.connectedEV) = 0
}

// impose that there must not exist multiple bookings for the
   ↪ same vehicles at the same time
fact noEVOverBooking{
    no disj b1, b2: Booking
```

```
    | b1.ev = b2.ev &&
        (gte[b1.start, b2.start] && lte[b1.start, b2.end] ||
         gte[b1.end, b2.start] && lte[b1.end, b2.end])
}

// impose that there must not exist multiple bookings for the
   ↪  same charging point at the same time
fact noOverBooking{
    no disj b1, b2: Booking
    | b1.cp = b2.cp &&
        (gte[b1.start, b2.start] && lte[b1.start, b2.end] ||
         gte[b1.end, b2.start] && lte[b1.end, b2.end])
}

//Only an EV can be in charge for EVD
fact onlyAnEVisChargingForEVD{
    all evd: EVD, disj ev1, ev2 :EV |
    ev1 in evd.evs and ev2 in evd.evs and ev1 in
   ↪ ChargingPoint.connectedEV
    implies ev2 not in ChargingPoint.connectedEV
}

//Unique email to EVD
fact uniqueEmail{
 no disj driver1, driver2: EVD |
     driver1.email = driver2.email
}

//Unique history to EVD
fact uniqueHistory{
 no disj driver1, driver2: EVD |
     driver1.chargingHistory = driver2.chargingHistory
}

//Unique location to ChargingStation
fact uniqueLocation{
 no disj cs1, cs2: ChargingStation |
```

```
    cs1.location = cs2.location
}

/**** ASSERTIONS ****/
// The same charging point can't be in two different charging
   ↪ stations
assert noCPinTwoChargingStations{
    no cp: ChargingPoint, disj cs1, cs2: ChargingStation |
        cp in cs1.chargingPoints && cp in cs2.chargingPoints
}

assert NoCSInTwoCPO{
 no cs: ChargingStation, disj CPO1, CPO2: CPO |
    cs in CPO1.chargingStations && cs in CPO2.
   ↪ chargingStations
}

assert notExistsCPnotInCS{
    no cp: ChargingPoint
    | cp not in ChargingStation.chargingPoints
}

assert existsEVnotInEVD{
    some ev: EV
    | ev not in EVD.evs
}

assert noBookingForUnEVD{
    Booking.ev in EVD.evs
}

assert noEVOverBooking{
    no disj b1, b2: Booking |
    b1.ev = b2.ev && gte[b1.start, b2.start] && lte[b1.start,
   ↪ b2.end]
}
```

```
assert bookingStartLessThanBookingEnd{
    all b: Booking | lt[b.start, b.end]
}

check noCPinTwoChargingStations

check NoCSInTwoCPO

/*********PREDICATES*********/
pred findBookings{
    some disj b1, b2: Booking | lt[b2.end, b1.start]
}

pred addEVToEVD[evd', evd: EVD, NewEv: EV]{
    evd'.evs = evd.evs + NewEv
}
run addEVToEVD

pred addCSToCPO[cpo', cpo: CPO, NewCS: ChargingStation]{
    cpo'.chargingStations = cpo.chargingStations + NewCS
}

run addCSToCPO

pred deleteCSFromCPO[cpo', cpo: CPO, cs: ChargingStation]{
    cpo'.chargingStations = cpo.chargingStations - cs
}

run deleteCSFromCPO

pred addCPToCS
 [cs', cs: ChargingStation, cp: ChargingPoint, cpo: CPO]{
    cs in cpo.chargingStations
    and
    cs'.chargingPoints = cs.chargingPoints + cp
    implies cpo.chargingStations = cs'
}
```

```
run addCPToCS for 20 but exactly 5 ChargingPoint

run findBookings for 10

//To show the CPO interaction with the system
pred CPOworld{
    #CPO >= 2
    #ChargingStation >= 2
    #ChargingPoint >= 5
}

run CPOworld for 20

//To show the EVD interaction with the system
pred EVDworld{
    #EVD >= 2
    #EV >= 4
    #ChargingPoint >= 3
    #ChargingStation >= 2
}

run EVDworld for 10

run {}
```

4.0.1. Resulting worlds

The world mainly from the CPO point of view From the CPO point of view we show the following world generated with the Alloy analyzer, noticing some important requirements of the eMall:

- The CPO manages one or more charging stations
- The charging stations have one ore more charging points
- The charging points can have different socket types (in the following representation Alloy generated a case in which each charging point has only a socket of one type,

but in general the charging point can have many sockets of different types, but we assume that only one is used at the time)

- The charging stations acquire energy from different DSOs chosen by the CPO, and different charging stations, even of different CPOs, can acquire energy from the same DSO

We can see a simple representation of these elements in the CPOworld:

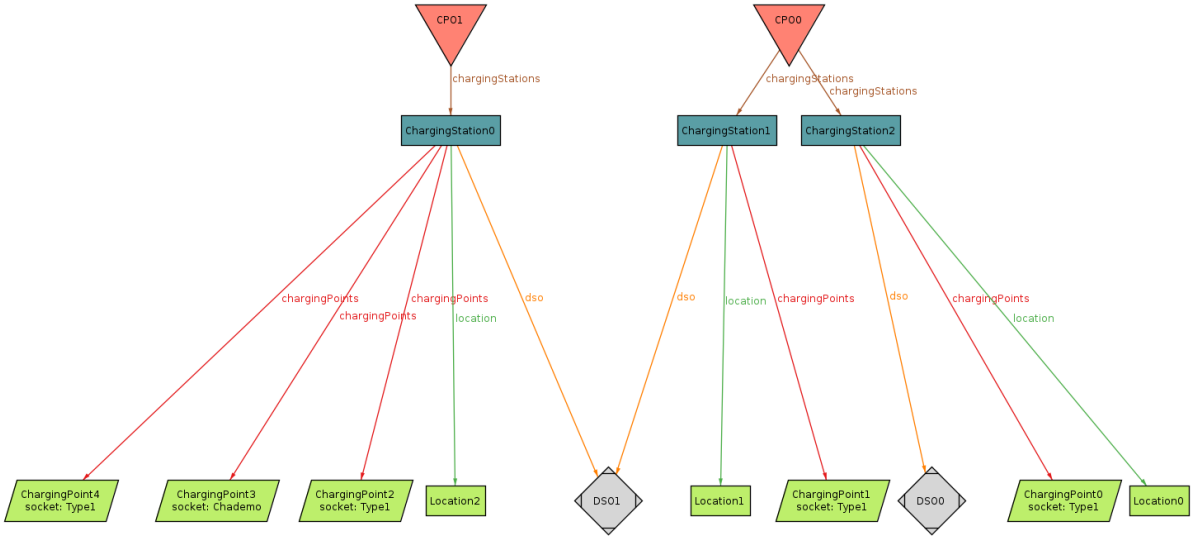

Figure 4.1: CPOworld

The world mainly from the EVD point of view To represent the EVD point of view we show another world generated by the Alloy analyzer, in order to verify some requirements:

- The

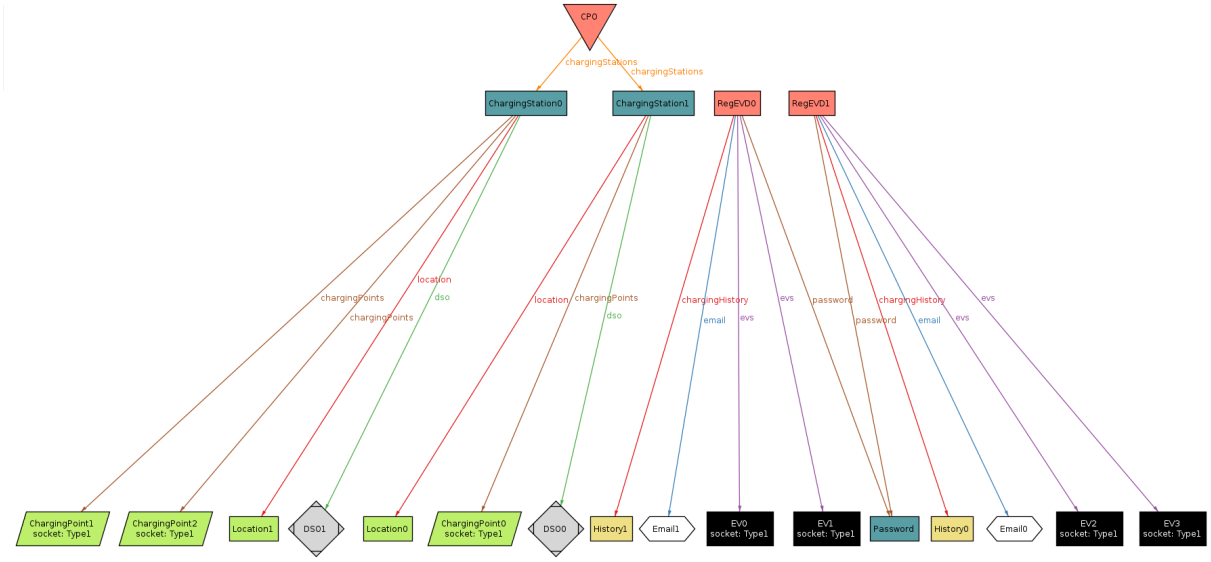

Figure 4.2: EVDworld

5 | Effort spent

Activity	Time spent
Organization	7h
Understanding the problem	10h
Introduction to the problem	10h
Scenarios and overall description	10h
Functional and non-functional requirements	22h
Formal analysis using Alloy	8h
Total time spent	h

Table 5.1: The time Bianca Savoiu has spent working on this project

Activity	Time spent
Organization	5h
Understanding the problem	10h
Introduction to the problem	10h
Scenarios and overall description	8h
Functional and non-functional requirements	12h
Formal analysis using Alloy	h
Total time spent	h

Table 5.2: The time Fabio Lusha has spent working on this project

6 | References

Bibliography

[1] C. Larman. *Applying UML and Patterns.* Prentice Hall PTR, 2005.

List of Figures

2.1 Domain model 14
2.2 State diagram of the EVD that starts a charging session 15
2.3 State diagram of the EVD that books a charging point 16
2.4 State diagram of the CPO that manages charging stations 16

3.1 Use cases diagram of the registered and unregistered EVD 29
3.2 RegisterEVD sequence diagram 31
3.3 LogIn sequence diagram 32
3.4 BookingCharginPoint sequence diagram 34
3.5 CancelChargingBooking sequence diagram 36
3.6 VisualizeChargingHistory sequence diagram 38
3.7 ChargeNow sequence diagram 40
3.8 ModifyProfile sequence diagram: the use cases ModifyProfileNewVehicle, ModifyProfileDeleteVehicle and ModifyProfilePersonalData are grouped in this diagram 44
3.9 Use cases diagram of the CPO 47
3.10 ManageStations sequence diagram 49
3.11 AddPromotion sequence diagram 54

4.1 CPOworld 80
4.2 EVDworld 81

List of Tables

1.1 Goals . . . 2
1.2 World Phenomena . . . 4
1.3 Shared Phenomena . . . 8

2.1 Assumptions . . . 24

3.1 RegisterEVD . . . 31
3.2 LogIn . . . 32
3.3 BookingChargingPoint . . . 33
3.4 CancelChargingBooking . . . 35
3.5 VisualizeChargingHistory . . . 37
3.6 ChargeNow . . . 39
3.7 ModifyProfileWithNewVehicle . . . 41
3.8 ModifyProfileDeletingVehicle . . . 42
3.9 ModifyProfilePersonalData . . . 44
3.10 VisualizeStations . . . 46
3.11 TerminateCharging . . . 46
3.12 ManageStations . . . 48
3.13 UpdateDSO . . . 51
3.14 UpdatePrice . . . 52
3.15 AddPromotion . . . 53
3.16 DeleteChargingStation . . . 55
3.17 AddChargingStation . . . 56
3.18 AddChargingPoint . . . 58
3.19 VisualizeGraphicalInformation . . . 59
3.20 SolveNotification . . . 61
3.21 Requirements . . . 65
3.22 Mapping on requirements . . . 66
3.23 Mapping on goals . . . 66

5.1 The time Bianca Savoiu has spent working on this project . . . 83

5.2 The time Fabio Lusha has spent working on this project 83