POLITECNICO DI MILANO

Computer Science and Engineering

Requirements Analysis and Specifications Document

eMall - e-Mobility for All

Software Engineering 2 Project

Academic year 2022 - 2023

05 February 2023

Version 3.0

Authors:
Balestrieri Niccolò
Bertogalli Andrea
Tombini Nicolò

Professor:
Matteo Camilli

Version history

Date	Revision	Notes
24/11/2022	v.1.0	First release.
4/12/2022	v.1.0.1	References Chapter and general fixing.
5/12/2022	v.1.0.2	Updating scope.
7/12/2022	v.1.0.3	Updating scope and general revision.
13/12/2022	v.1.0.4	Fixed a typo.
17/12/2022	v.1.0.5	New domain assumption.
21/12/2022	v.1.1	Final release.
06/01/2023	v.2.0	Change protocol from OCPI to HTTP and final release.
05/02/2023	v.3.0	Fix of typos and updating of communication interfaces.

Contents

Chapter 1

Introduction

1.1 Purpose

This document represents the **R**equirements **A**nalysis and **S**pecification **D**ocument (RASD). Its purpose is to present a thorough and meticulous description of the e-Mobility for All (eMall) system in terms of functional and non-functional requirements highlighting real possible use cases, system limits and constraints and finally end users interaction. The following document it is also a strong baseline for project planning and for a future software evaluation.

The audience of this document includes end users that are mostly interested in an high level description of the features and developers who have to implement the requirements presented in this document.

1.1.1 Goals

Nowadays, the urban and sub-urban mobility is one of the main causes of pollution which afflicts our planet, so to reduce the carbon footprint the usage of electric vehicles is encouraged as much as possible. The main goal of eMall is to help the spreading of this kind of transports focusing on the following general points:

Goal	Description
G.1	Allow end users to find a most suitable charging station for them, according to various criteria.
G.2	Allow end users to make bookings and manage them.
G.3	Allow end users to start/monitor/stop the charging process.
G.4	Allow end users to automatically pay for a charge.
G.5	Allow the charging station administrators to monitor various information about their charging stations as the internal and external status.
G.6	Allow the charging station administrators to retrieve information about the price from the energy providers also choosing which supplier to buy from.
G.7	Allow the charging station administrators to decide how to use purchased energy and also decide its selling price.

1.2 Scope

Electric mobility for All (eMall) is a **user-friendly** application which is intended to facilitate the use of the system to electric vehicle owners and charging station administrators.
To reach these goals, the application provides an interface for end users that allows them **to search for a nearby charging station** that has a certain type of charging sockets (slow, fast, rapid), visualizing also its special offers. Consequently, end users **can charge their vehicles** in a specific station by reserving a time slot receiving a notification when the charge is completed. This feature allows to avoid traffic at the station and reduce the simultaneous presence of vehicles. The system also provides a full integrated payment system that enables end users to pay for the service.
The software also proposes a very smart feature: indeed, is able to proactively suggest the user to go and charge the vehicle, depending on the status of the battery, his schedule, any special offers available, and the availability of charging slots at the identified stations.
The platform includes also a charging station management system which is used by the charging station administrators **to manage the infrastructure (batteries, etc) of charging stations** and, moreover, it has the possibility to buy the energy from 3^{rd} parties in the smartest possible way and distribute it to end user vehicles. The management system will also **keep track of the internal and external status** of each charging station respecting the physical constraints of the infrastructure.

1.2.1 World phenomena

World Phenomena	Description
WP.1	The end user needs to charge his vehicle.
WP.2	The end user drives to the chosen charging station.
WP.3	The end user plugs in the cable in the designated charging socket.
WP.4	The end user unplugs the charging cable.
WP.5	The end user leaves the charging station.
WP.6	The charging station administrator owns the charging station.

1.2.2 Shared phenomena

Shared Phenomena	Description
SP.1	The system shows charging stations nearby, their cost and any special offer they have.
SP.2	The end user can book a time slot at the charging station.
SP.3	The end user can start the charging process.
SP.4	The end user can pay for the service, then he receives the invoice through an email.
SP.5	The system suggests the most suitable charging station for a certain end user.
SP.6	The charging station administrator acquires energy from energy providers dinamically chosen.
SP.7	The charging station administrator or the system decide whether or not to buy energy or use batteries.
SP.8	The system allows the charging station administrator to know about the status of his charging stations.
SP.9	The interaction between the end user and the charging socket happens through a QR-Code scan.

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

Term	Definition
End user	Identifies electric vehicles owners who use the service, can also be referred as user.
Charging station	Place of charging of electric vehicles.
Charging Station available	At the moment at least one socket is free.
Charging Station not available	At the moment all the sockets are occupied.
Charging socket	Single charging point.
Booking	Reserved time slot in a specific station for charging vehicles. It also can be referred to as Reservation.
Time slot	Slot of time in which customers can charge their vehicles.
QR-Code	A QR-code is a type of matrix barcode (or two-dimensional barcode).
System	Set of hardware and software tools that provide the desired service. It can be considered as eMall, Application and Platform.
Internal status	For internal status of a charging station: the amount of energy available in its batteries, the number of vehicles being charged and, for each charging vehicle, the amount of power absorbed and time left to the end of the charge.
External status	For external status of a charging station: the number of charging sockets available, their type such as slow/fast/rapid, their cost, and, if all sockets of a certain type are occupied, the estimated amount of time until the first socket of that type is freed.

1.3.2 Acronyms

Acronyms	Meaning
eMall	e-Mobility for All
eMSP	e-Mobility Service Provider
CPO	Charging Point Operator
CPMS	Charging Point Management System
DSO	Distribution System Operator
API	Application Programming Interface
GPS	Global Positioning System
PC	Personal Computer
HTTPS	Hyper Text Transfer Protocol Secure
HTTP	Hyper Text Transfer Protocol
WiFi	Wireless Fidelity
LTE	Long Term Evolution
3G	Third-Generation Wireless
4G	Fourth-Generation Wireless
5G	Fifth-Generation Wireless
SSL	Secure Socket Layer
SHA-256	Secure Hash Algorithm
SCA	Strong Customer Authentication
a.m.	Ante Meridiem
p.m.	Post Meridiem

1.3.3 Abbreviations

Abbreviations	Meaning
WP	World Phenomena
SP	Shared Phenomena
G	Goal
R	Requirement
NFR	Non Functional Requirement
D	Domain Assumption
w.r.t.	with reference to
e.g.	exempli gratia
i.e.	id est
etc.	etcetera

1.3.4 Important terminology

- The **CPO** is the charging station's administrator and owner (for example an e-mobility company).
- The **eMSP** is the system module which interfaces with the end users, it includes services and a mobile application.

- The **CPMS** is the system module which interfaces with the CPOs, and manages the charging stations.

The eMall platform provides both a CPMS and an eMSP modules.

1.4 Reference Documents

- *Course slides on WeeBeep.*
- *RASD assignament document.*
- *RASD review by Prof. M. Camilli.*

1.5 Document Structure

The structure of this RASD document is the following:

1. **Introduction**: In this section is presented the purpose of this document highlighting in particular the main goals, the audience which is referred to, the identification of the product and application domain describing world and shared phenomena and, lastly, the terms definitions.
2. **Overall Description**: This chapter describes the possible scenarios of the platform, the shared phenomena presented at the beginning of the document and assumptions on the domain of the application.
3. **Specific Requirements**: Includes all the requirements in a more specific way than the "Overall Description" section. Moreover, it is useful to show functional requirements in terms of use cases diagrams, sequence/activity diagrams.
4. **Formal Analysis Using Alloy**: Includes Alloy models which are used for the description of the application domain and his properties, referring to the operations which the system has to provide.
5. **Effort Spent**: This section shows the effort spent in terms of time for each team member and the whole team.
6. **References**: Includes all documents that were helpful in drafting the RASD.

Chapter 2

Overall Description

2.1 Product perspective

In the section is presented a list of real sceanarios and diagrams illustrating further details about shared phenomenas.

2.1.1 Scenarios

1st scenario: An end user needs to charge his vehicle

Bob, who owns an electric vehicle and who previously registered to the system, while on the road needs to charge it. For this reason he opens the eMall app which offers him the charging stations closest to him, available at that precise moment with the relative charge prices and any offer available. Subsequently, he books, in that station, the desired socket for a certain time receiving a QR-Code to be shown at the socket scanner at the moment of charging. At the end of the process, the system notifies Bob of the end and automatically subtracts the cost of the charge and, moreover, it sends the invoice at the user's email address.

2nd scenario: An end user needs to delete a booking

Bob is at home planning a trip, so he decides to look at the charging stations on his way by booking them in advance. During the booking, he realizes that he has mistakenly selected an unwanted station; for this reason, he deletes the booking using the appropriate section of the application.

3rd scenario: A CPO monitors a charging process

Bob has started the charging process in a station at the rapid socket 3. Bob wants a coffee, so he goes to a nearby cafe, meanwhile the CPO can monitor, through the CPMS, the charging process of Bob's brand new car to check that everything is fine.

4th scenario: A CPO installs a new charging station

A CPO has just installed a new charging station that must be added to the system. So, through the CPMS the CPO can insert the new charging station providing specific information, i.e. number and type of each charging socket, coordinates, etc.

5th scenario: A CPO makes an energy purchase

A CPO wants to purchase energy from the most convenient DSO, choosing between different options presented by the system in order to provide end users with more and more appealing offers. Furthermore, since the price of energy today is lower than usual, the CPO decides to store the energy inside the batteries to maximize the profits as much as possible instead of acquiring energy directly from DSO.

2.1.2 Class diagram

Class diagrams are useful for illustrating relationship between classes and interfaces.

- **User**: in eMall system, user is the general term to indicate **CPO** or **End user**.
- **End user**: represents an end user of eMSP software.
- **CPO**: represents a user of CPMS software.
- **DSO**: represents the energy provider.
- **Booking**: represents a reservation with some information.
- **Charging Station**: represents the charging station infrastructure. It is composed by at least one charging socket.
- **Charging Socket**: represents the charging point.
- **Payment Method**: represents a method with which the end user pays for the service.
- **Socket Status**: it's an enumeration that represents the status of charging sockets (*Busy, Free*).
- **Socket Type**: another enumeration that represents the type of charging sockets (*Slow, Fast, Rapid*).
- **Booking Status**: enumeration that represents the status of booking (*Expired, Valid*).

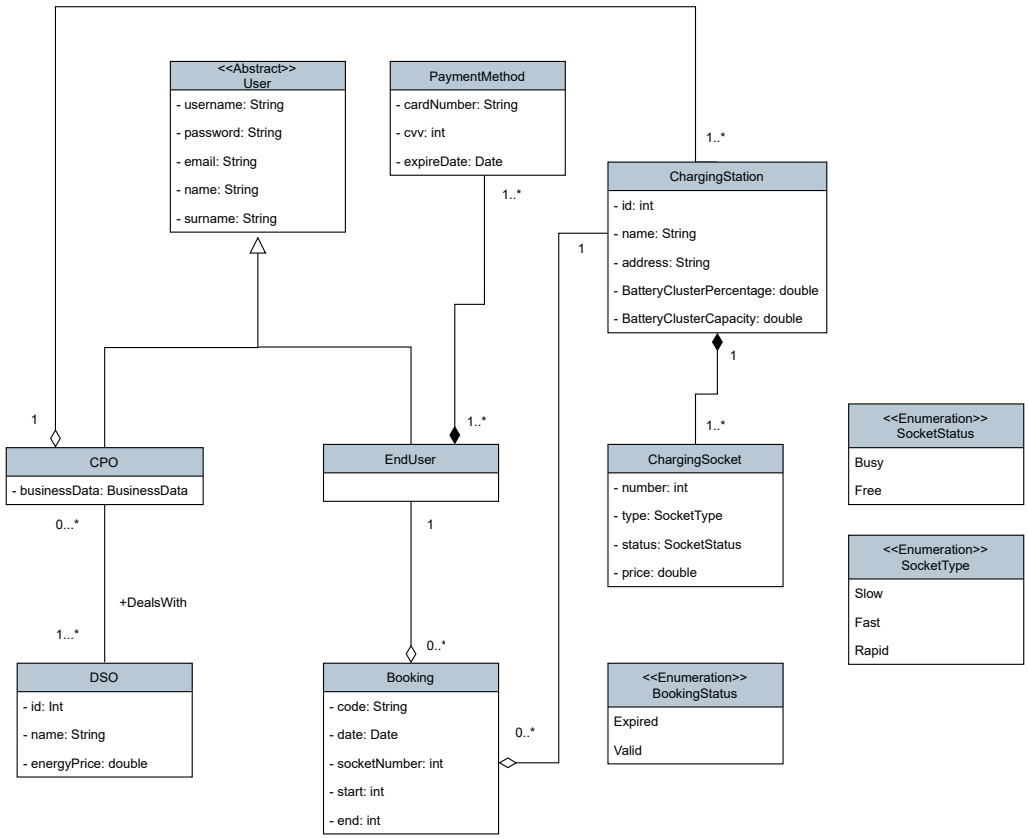


Figure 2.1: Class diagram

2.1.3 State diagram

State diagrams model the behavior of a single object and are used for objects with significant dynamic behavior; they also specify the sequence of states that an object goes through during its lifetime in response to stimuli from the environment. In particular, they show the life history of a given class, the events that cause a transition from one state to another and the actions that result from a state change.
In this section we will represent the main state diagrams of the whole system.

Authentication

In figure 2.2 the authentication process carried out by both end users and the CPO is highlighted. First of all, the user (CPO/end user) has the possibility to register: if registration fails the system will repeat the operation, while if successful the user can log in.
If the end user or the CPO previously had an account, then they can log in directly: if the username and password are correct, the system shows the dashboard relating to the role that the person attempting to access has, otherwise it asks to re-enter the credentials.
Figure 2.3 shows how this procedure is used before the booking process, obviously by end users.

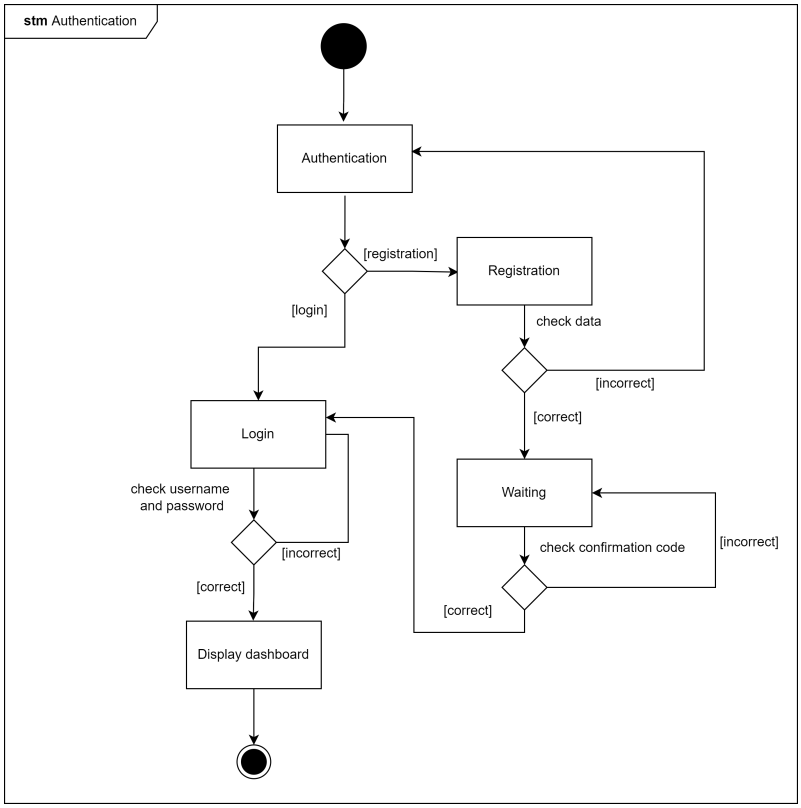


Figure 2.2: Authentication state diagram

Booking

In the following figure is shown the process of reserving the charging socket by the end users. After logging in successfully, the reservation is created by choosing the preferred charging station, charging socket and timeframe: the system checks the validity of the choice and repeats the process if incorrect. If correct, a valid QR-Code is generated for the subsequent recognition of the user at the correct station (as explained in detail in figure 2.4). Obviously, after the reservation has been made successfully, the system also provides the possibility to delete it.

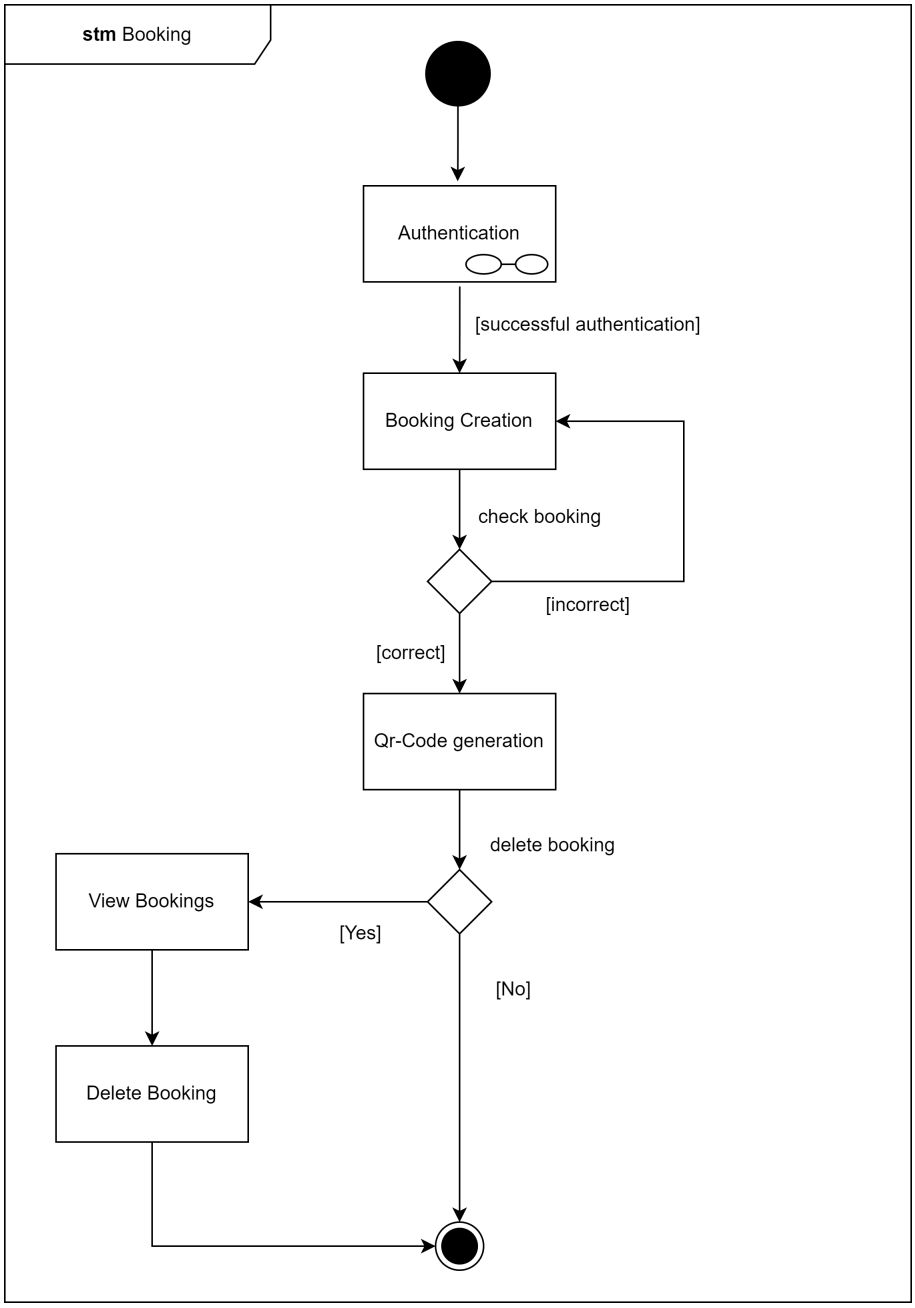


Figure 2.3: Booking state diagram

QR-Code Recognition

Figure 2.4 shows how end users reservations are recognized by a charging socket of a specific charging station. First, the QR-Code is scanned from the smartphone and the system checks its existence; if it exists, its validity is checked in terms of expiration. If it has not expired, end users are allowed to charge their vehicle.

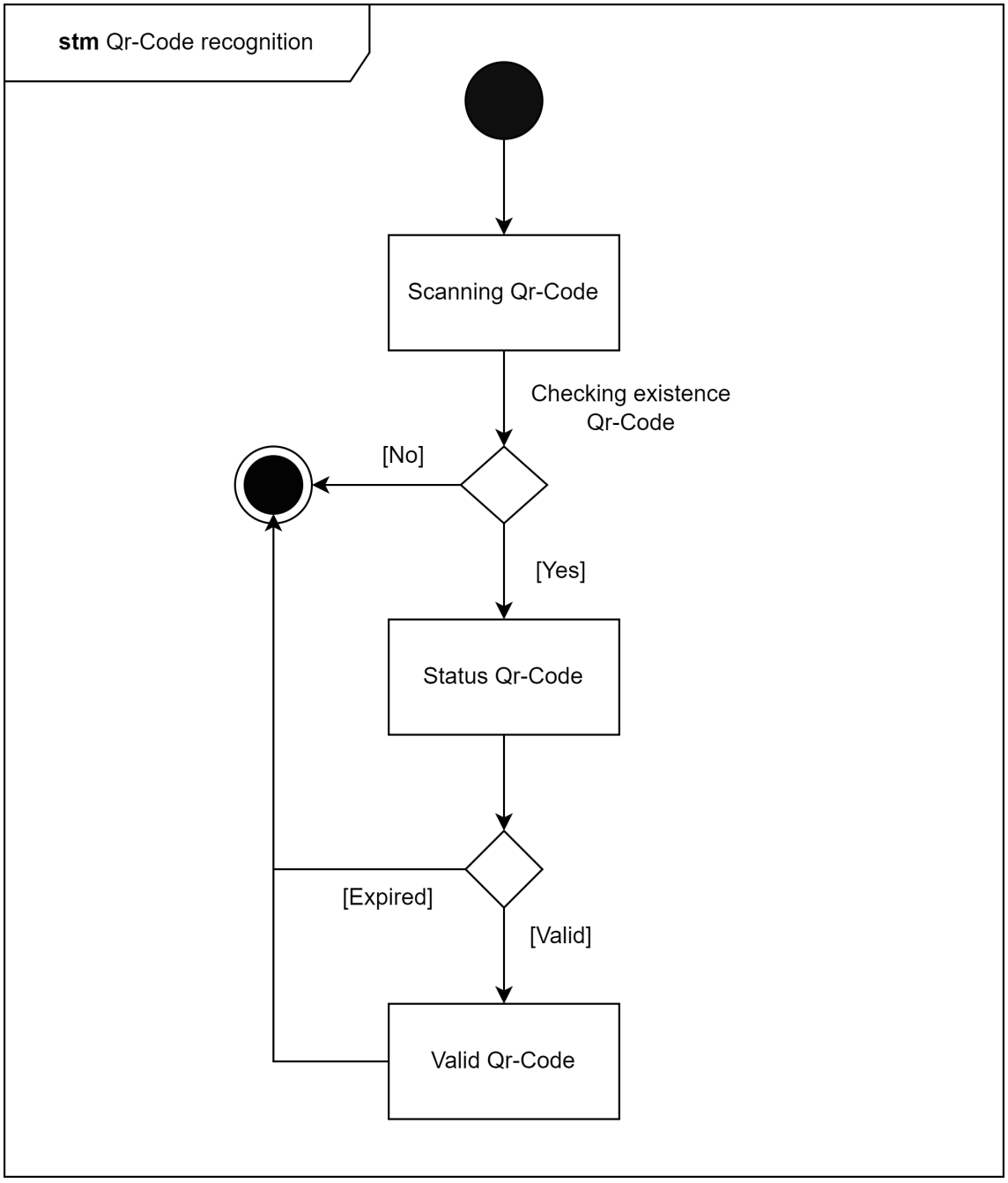


Figure 2.4: Qr-Code Recognition state diagram

2.2 Product functions

In this section is presented a detailed description for each functionality that the platform provides, w.r.t the goals described in the first section 1.1.1.

2.2.1 System authentication function

The system allows both the end users and CPO to register an account and consequently login with that, but the two operations differ on the data which are provided during the registration. Obviously the privileges that a CPO and an end user have are completely different.

End users authentication

End users have the possibility to log into the application using **username and password**. For the registration a new end user needs to provide username, password, an **email** and personal data. The email provided will be confirmed through a **OTP** received on the latter.

CPOs authentication

CPOs, like end users, have the possibility to log in into the application with **username and password**. The operation that is different from end users is the registration, in fact, a CPO in the registration phase must provide personal data, an **email** but also **business data** related to the CPO's company. When a CPO is creating a new account he also needs to provide information related to his charging station. The email provided will be confirmed through a **OTP** received on the latter.

2.2.2 Charging station booking function

A reservation specifies three main points of information, that are:

- The designated charging station.
- A charging socket installed in the station.
- A time interval in which the end user will charge his vehicle.

Obviously the reservations can't overlap each other, an end user in fact can't book a charge in two different stations at the same time, but he can book multiple charges in different days for examples. The eMSP also provides the possibility to **delete** a reservation if needed.

Charging station search function

The eMSP suggests to the end user the best options he has when he needs to charge his vehicle according to his calendar, the remaining amount of energy, the distance, the energy price, special offers available and more. Once the end user selects the most suitable station, the application, that deals with the end user GPS, sets up the navigation to the charging point.

Booking confirmation function

When a customer makes a reservation a **QR-Code** is generated. The code is important because it represents in a **unique way** the reservation and is mandatory to verify the identity of the customer at the charging point, avoiding scams and to assure the correct usage of the charging sockets.

Booking validation function

When an end user reaches the designated charging station and consequently the assigned socket, he needs to confirm to the system that he actually made the reservation. To do so the end user uses the proper **QR-Code scanner** on the socket, which deals with the validation process and allows the charging operation. A reservation is valid until the upper bound of the time slot **is past**: in this case the user isn't allowed to charge his vehicle and needs to book again.

Booking payment function

Once the charging process is completed, the system will automatically subtract the correct amount of money from the **payment method** specified by the user on his account. After the payment the user will receive an **invoice** on his personal (confirmed) email.

2.2.3 Charging station management function

The CPMS supplies functionalities that are used by various CPOs to manage their stations: the functions vary from the charging socket interaction to the dialogue with the DSO.

Energy acquisition function

The CPMS provides to the CPO the possibility to **dinamically buy energy** from third parties distributors (DSO). For "dinamically buy" is intended that the purchase is made considering different aspects like the comparison between energy prices of multiple DSOs.

Energy distribution function

Through the CPMS's dashboard, the CPO can decide the energy selling price and can also set special offers for his end users. The selling price and, consequently, the offers depend on the purchase price and on the energy source. The possible energy sources are storage batteries or the direct links with the DSO. The choice of the energy supply, for charging vehicles, can be automatically made by the CPMS or manually by the CPO with the aim of saving money w.r.t. the scenario described in 2.1.1 and to reduce the energy impact on electric network, paying particular attention to avoid absorption peaks.

Station monitoring function

Using the CPMS, the CPO can monitor both internal and external status. For internal status are intended the following features:

- Level of energy in its batteries.

- Number of vehicles being charged.
- Amount of power absorbed for each charging vehicle.
- Time left to the end of the charge.

While for external status are intended the following points:

- Number of charging sockets available.
- The socket type (slow/fast/rapid) .
- The socket cost.
- The estimated amount of time until the first socket of a type is freed.

Charging termination function

The system automatically notifies the user when the charging operation of his vehicle ends.

New infrastructure function

The CPO is able to add a new charging station and a new charging socket to the system, providing various information like: the position, the organization etc.

2.3 User characteristics

The eMall system has two user categories:

- **End users**: they are the main users of the system who use the eMSP, and they can be of two types:
  - driving license holders
  - non-holders

  The latter may want to use the system to help or advise relatives and/or friends or, more simply, for pure personal interest.
- **CPOs**: The CPOs are administrators and owners of various charging stations. They also have to provide, when necessary, all the commercial information.
  They are able to manage the station in all its aspects, also purchasing energy from third parties (DSO). Although the application is simple to use, it is assumed that they have taken the online course for CPO.

2.4 Assumptions, dependencies and constraints

2.4.1 Assumptions and Dependencies

The assumptions, in the domain of interest of eMall, are illustrated below:

Assumptions	Description
D.1	All end users must have a smartphone with an available Internet connection.
D.2	All smartphones used must have a functioning and actionable GPS.
D.3	All end users must have at least one electric vehicle to charge.
D.4	All end users must give consent to the system to access their calendar.
D.5	All end users must give consent to the system to access their navigation system and the status of battery of their vehicles.
D.6	All charging stations have a QR-Code scanner that allows to recognize a specific end user in a specif time slot.
D.7	All end users must have a working charging cable and any adapters useful for charging process.
D.8	All end users must be able to link cable from stations to vehicles and to charge vehicles themselves.
D.9	All CPOs must have Internet connection to use CPMS software.
D.10	All CPOs must have the adeguate IT infrastructure.
D.11	All CPOs should have a good knowledge of the e-mobility domain.
D.12	Charging stations have unique codes.
D.13	All CPOs have at least one charging station.
D.14	Qr-Codes are associated with a specific reservation.
D.15	All end users must have an email address to sign-up.
D.16	All CPOs must have an email address and all commercial data to sign-up.
D.17	A timeslot begins and ends only at the hour on the dot.
D.18	Only end users start the charging process.
D.19	All charging stations are composed by at least one charging socket.
D.20	All charging sockets can charge one vehicle at a time.
D.21	A DSO has only one offer for the energy.
D.22	All charging sockets are "charging columns" with only one cable attachment.
D.23	eMall doesn’t handle the possible bankruptcy of a CPO or the closure of a charging station.
D.24	If the user does not have enough money to pay, if his bank does not allow debt, the system will attempt to subtract the money periodically.
D.25	All CPOs don’t have a payment method, as the billing of the energy purchased by the DSOs is done externally to the eMall system.
D.26	All end users have debit cards as a method of payment and cannot make bank transfers.
D.27	All charging stations have a battery cluster to store the energy purchased by the DSO.

D.28	The name, the surname and the e-mail of the CPO are related to the representative for the company which is in charge for the station monitor operations.
D.29	For simplicity's sake, CPO business data is a company code of length 10, that is unique in the whole system.
D.30	Only one CPO per company exists.

2.4.2 Constraints

In this section are illustrated limits and system features.

Privacy policies

The eMall system collects and uses the end user's and CPO's data to better manage the service provided. A user who wants to use the system must accept consent to the processing of data, which will not in any case be disclosed to third parties or used for marketing purposes.

Hardware limitations

Although eMall is a simple system to use, unfortunately it is not available on all devices.
The requirements that the devices exploiting the system must meet are listed below:

- LTE/3G/4G/5G connection
- iPhone with iOS version at least 13.0
- Android with Android 6 Marshmallow
- GPS available
- Web Browser (Opera, Safari, Chrome, Firefox), for the CPO's app.

Software limitations

The system used is strictly related to the use of APIs regarding positioning on the map: they are necessary for the correct functioning of the platform.
Moreover, if the system does not have access to the user's calendar, battery status or navigation system, then eMall will not operate at its full capacity.

Chapter 3

Specific Requirements

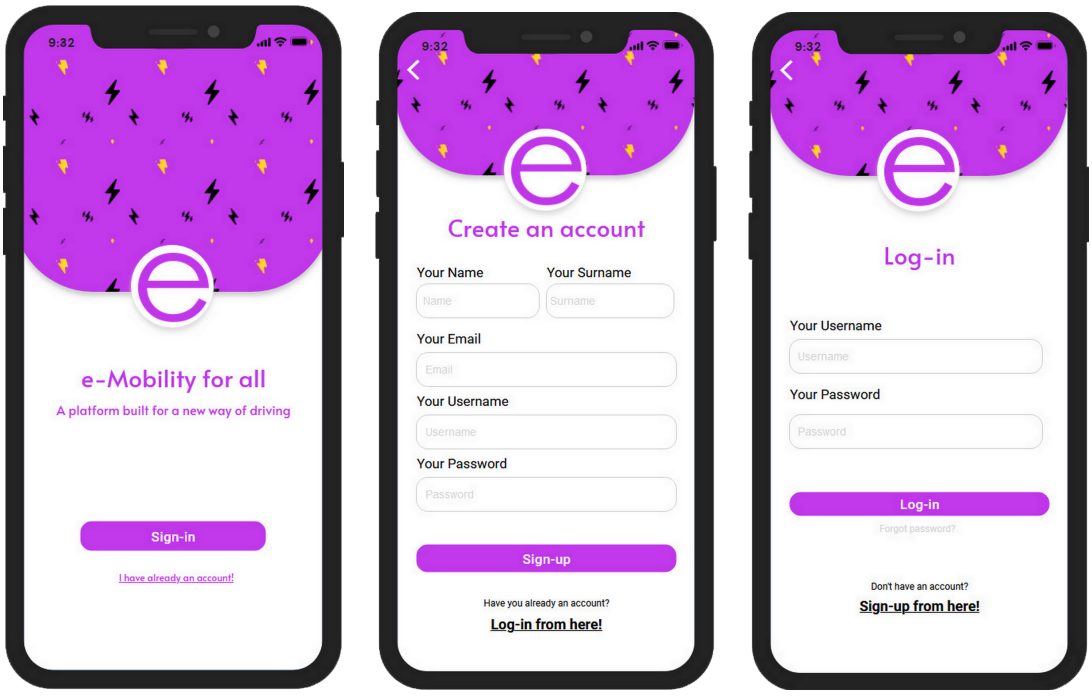
3.1 External Interface Requirements

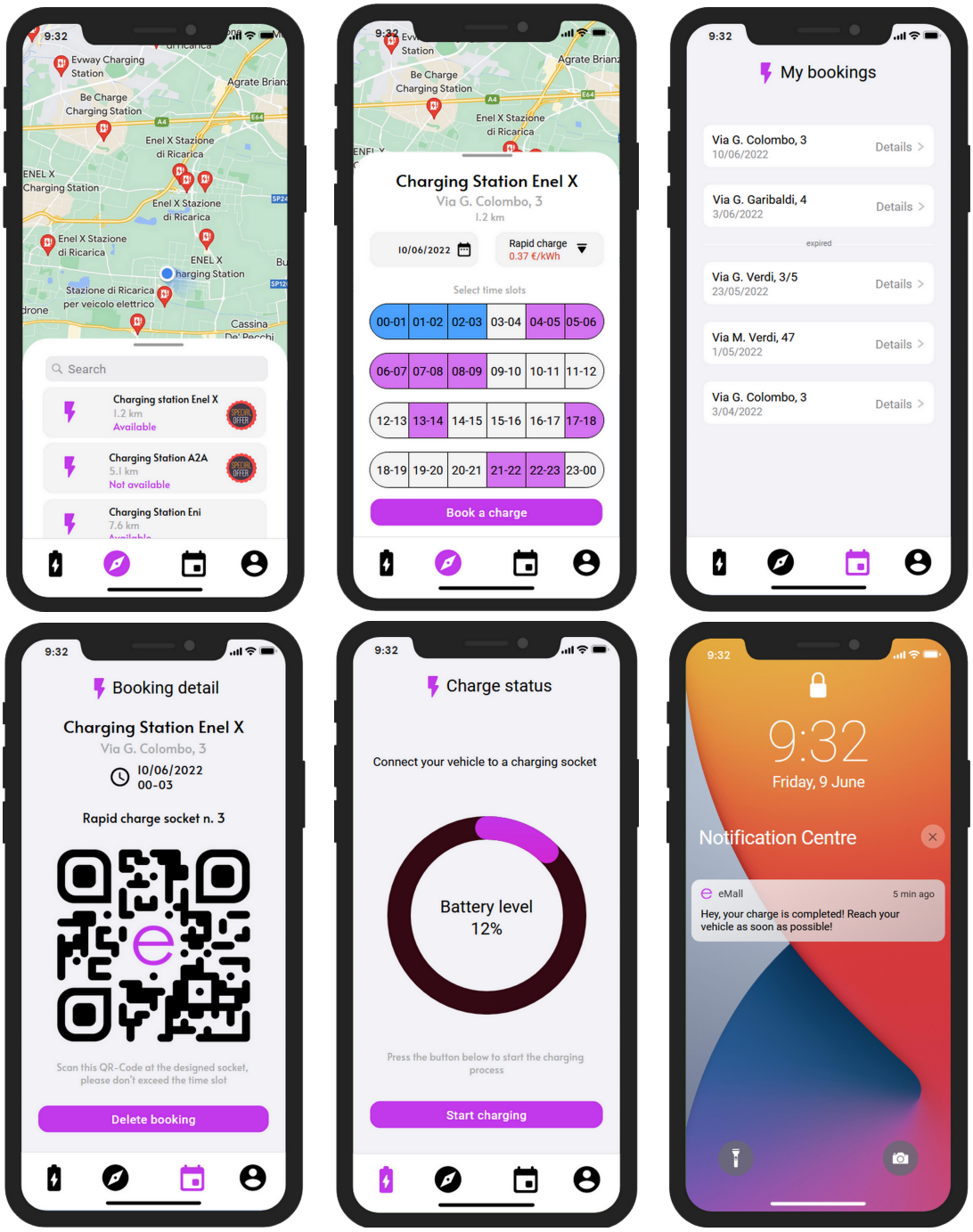
In this section will be presented all the specific functional requirements of the eMall system. They outline how the system will interface with other components.

3.1.1 User Interfaces

This section describes the logical characteristics of each interface between the software and the users. It defines the software components for which a user interface is needed.

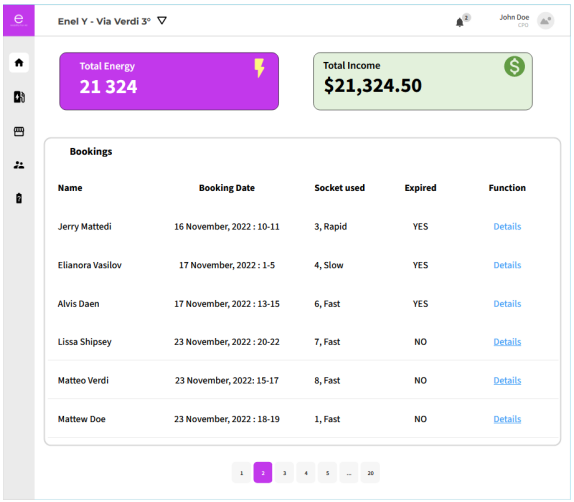
End User interfaces

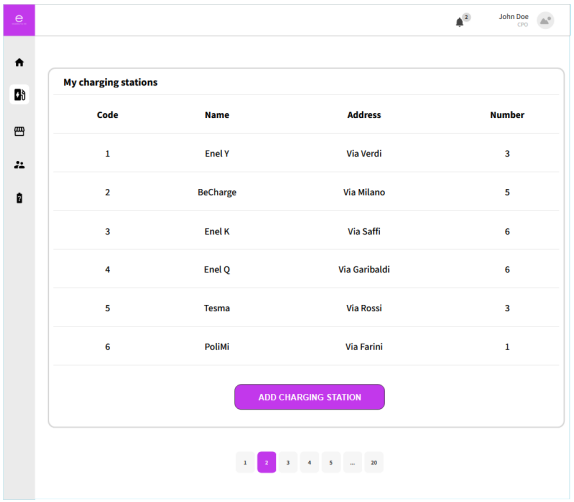


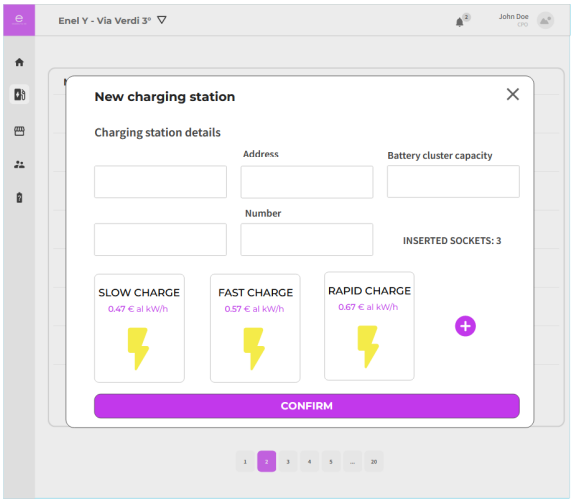

9:32
Evway Charging Station
Be Charge Charging Station
Enel X Stazione di Ricarica
ENEL X Charging Station
Stazione di Ricarica per veicolo elettrico
Search
Charging station Enel X
1.2 km
Available
Charging Station A2A
5.1 km
Not available
Charging Station Eni
7.6 km
Charging Station Enel X
Via G. Colombo, 3
1.2 km
10/06/2022
Rapid charge
0.37 €/kWh
Select time slots
00-01 01-02 02-03 03-04 04-05 05-06
06-07 07-08 08-09 09-10 10-11 11-12
12-13 13-14 14-15 15-16 16-17 17-18
18-19 19-20 20-21 21-22 22-23 23-00
Book a charge
My bookings
Via G. Colombo, 3
10/06/2022
Details
Via G. Garibaldi, 4
3/06/2022
expired
Via G. Verdi, 3/5
23/05/2022
Via M. Verdi, 47
1/05/2022
Via G. Colombo, 3
3/04/2022
Booking detail
Charging Station Enel X
Via G. Colombo, 3
10/06/2022
00-03
Rapid charge socket n. 3
Scan this QR-Code at the designed socket, please don't exceed the time slot
Delete booking
Charge status
Connect your vehicle to a charging socket
Battery level
12%
Press the button below to start the charging process
Start charging
9:32
Friday, 9 June
Notification Centre
eMall
5 min ago
Hey, your charge is completed! Reach your vehicle as soon as possible!

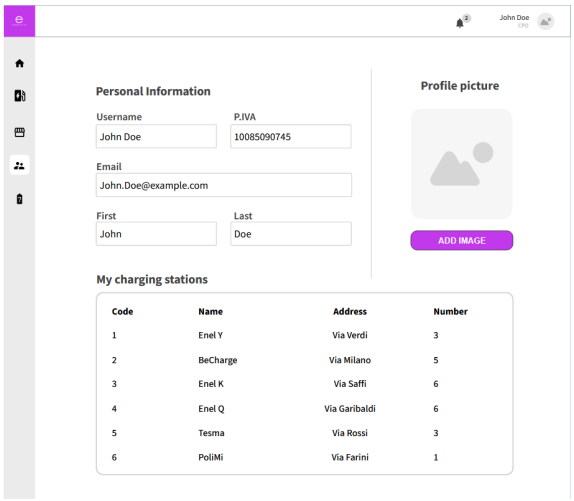

9:32
Charge status
Charging Station Enel X
Via G. Colombo, 3
Rapid charge socket n. 3
Charging
65%
Elapsed time: 1h 14m 34s
Stop charging
9:32
JDoe00
Personal data
John Doe
John.Doe@example.com
Payment methods
Credit Card
J. Doe
Access privileges
GPS
Calendar
iBattery

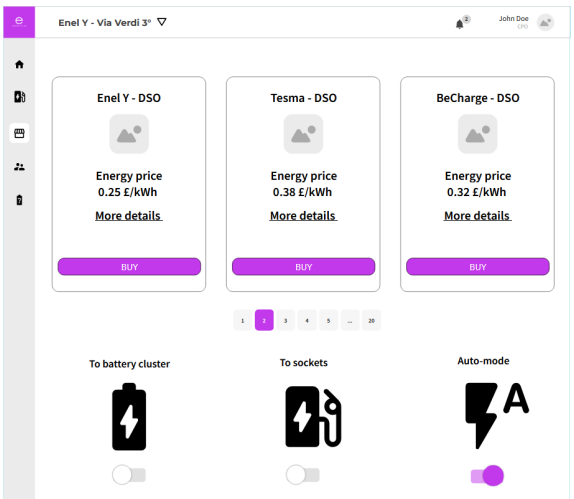
CPO interfaces

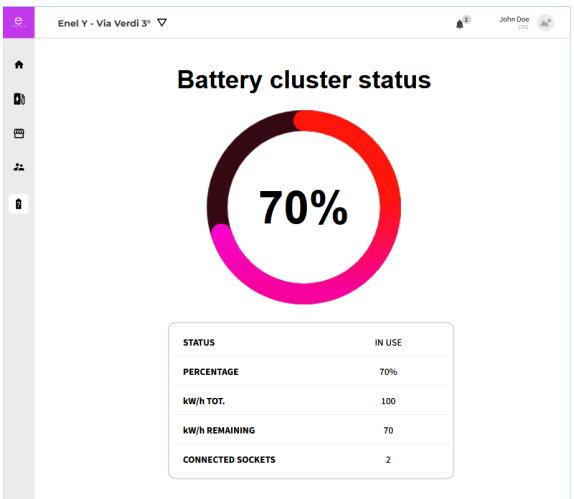












3.1.2 Hardware Interfaces

This section describes the logical and physical characteristics of each interface between the software and the hardware components of the system.
In order to allow the entire system to function properly, some devices are required:

- **smartphones** for end users equipped with GPS.
- **smartphones** or **PC** for CPOs to access their control panel via **web browser**.
- **QR-Code** scanner installed on the charging socket to allow end users' booking recognition.

3.1.3 Software Interfaces

This section describes the connections between the system and other specific software components.
The eMall system communicates with geolocation APIs to view the charging stations according to the criteria shown above, with bank APIs to substract automatically the amount of a charge. Moreover, the system should have an access to end user's vehicle APIs and end user's calendar APIs to proactively suggest a specific charging stations.

3.1.4 Communication Interfaces

This section describes the requirements associated with any communication function required by this system.
All the communications of the eMall infrastructure are made via the HTTP application layer protocol: obviously, all the devices using the platform must be connected via WiFi or mobile network (LTE/3G/4G/5G). Due to the fact that some of the interactions are real time, for example the communication between the charging sockets and the CPMS, is necessairly to use WebSocket protocol, which enables real time communication over HTTP.

3.2 Functional Requirements

3.2.1 Use Cases Diagrams

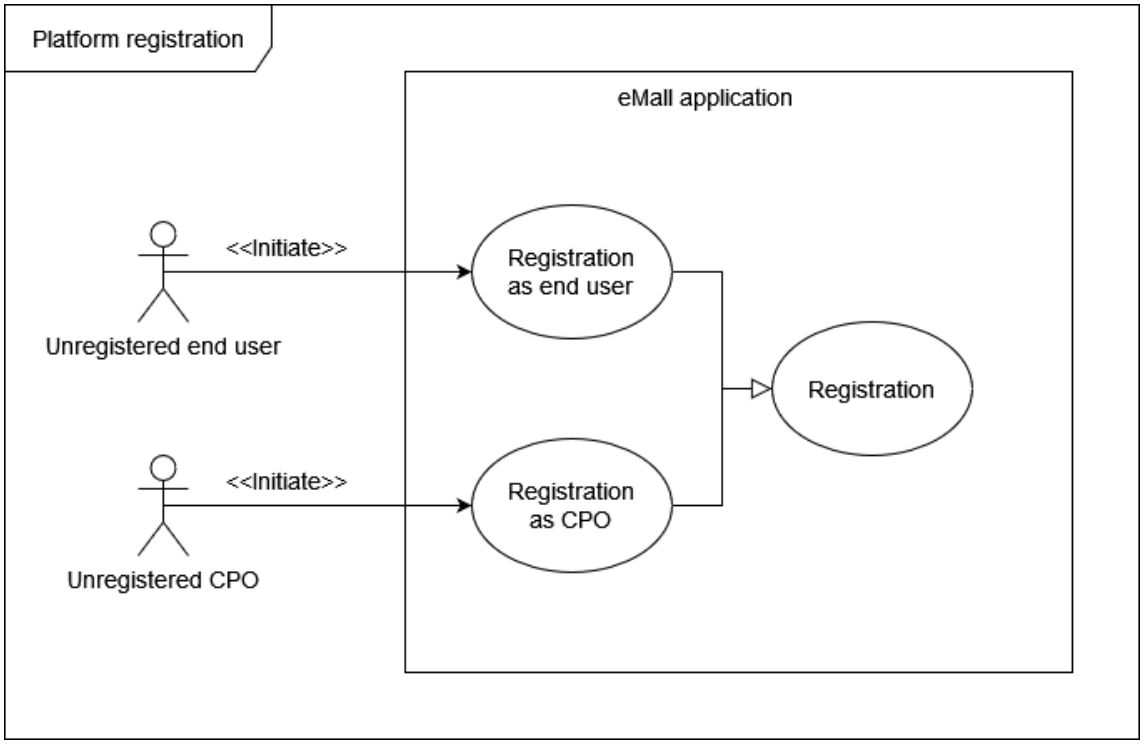


Figure 3.1: Registration operation use case diagram.

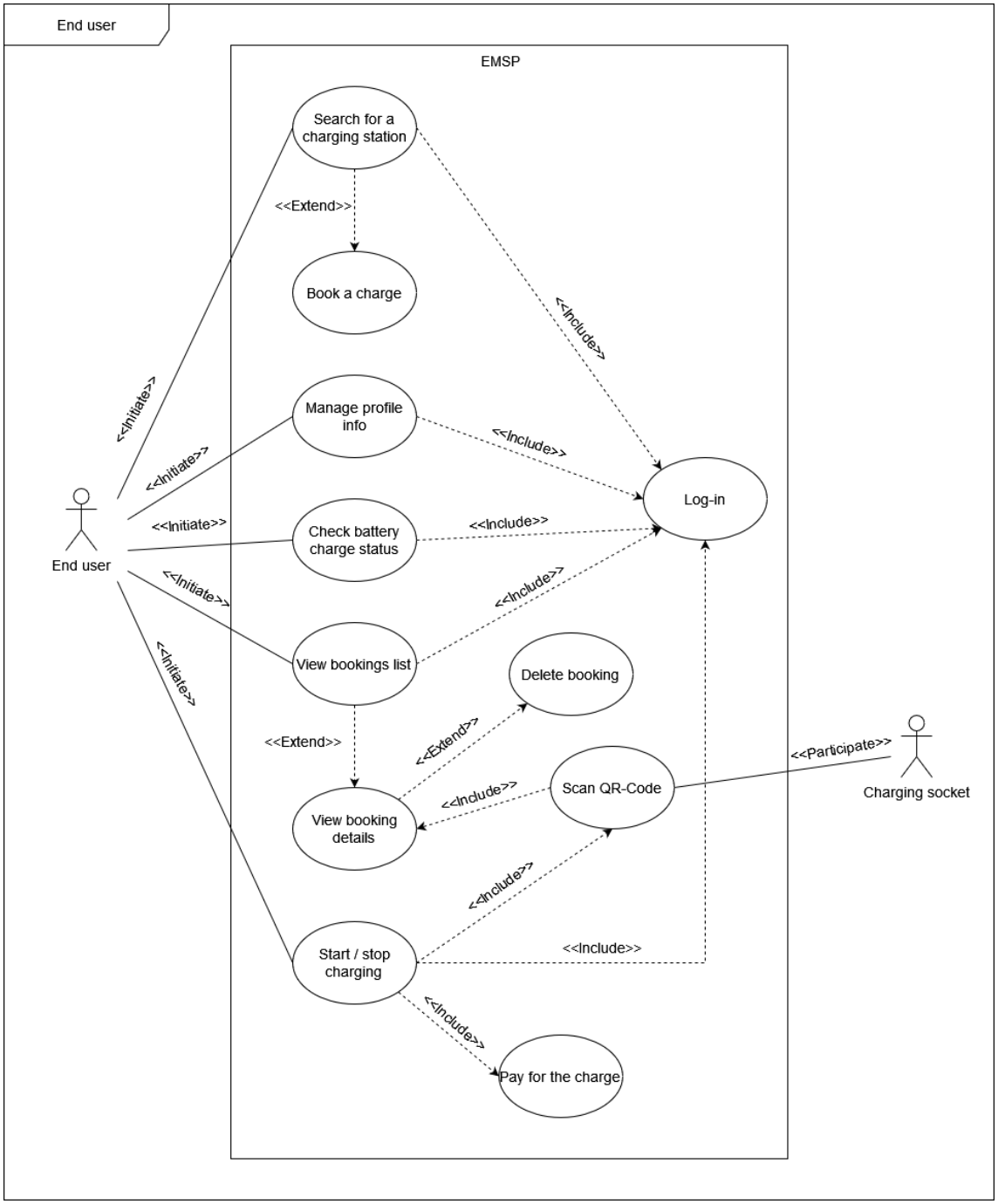


Figure 3.2: End user's operations use case diagram.

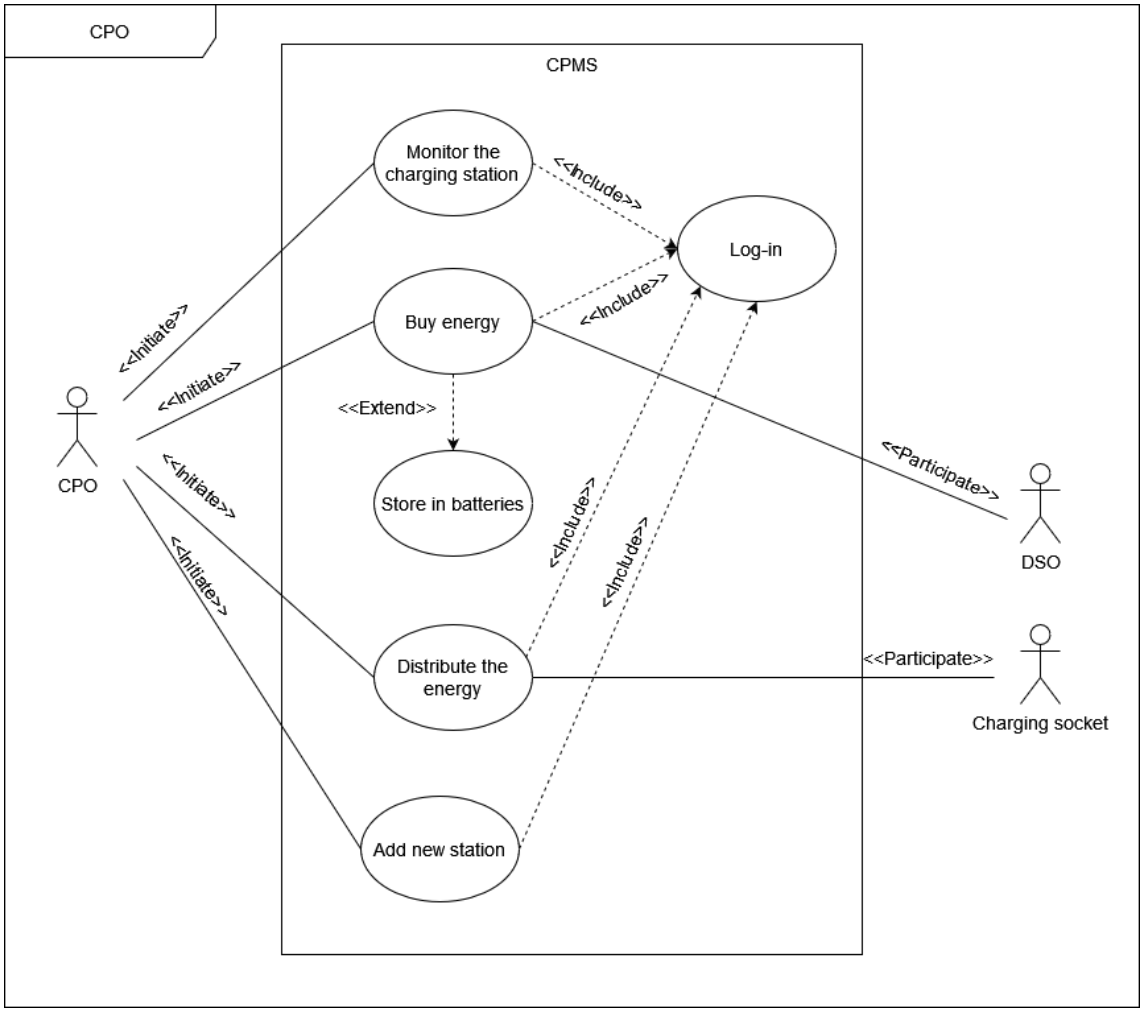


Figure 3.3: CPO's operations use case diagram.

3.2.2 Use Cases Description

Name	Platform registration
ID	UC.1
Actors	Unregistered End User, Unregistered CPO
Entry Condition	1) The actor is running the application.
Event Flow	1) The actor navigates to the sign-up page. 2) The end user fills the sign-up form with name, surname, email, username and password. 3) The unregistered CPO fills the sign-up form with name, surname, email, username and password. The CPO also needs to insert some business data. 4) The actor submits the registration. 5) The system sends a confirmation email to verify the actor's email. 6) The system stores the actor's data. 7) The system redirects the actor to the log-in page.
Exit Condition	The actor signed-up correctly.
Exceptions	1) The provided username or email are already registered. 2) The actor doesn't fill the form with correct information. All *execptions caused by user errors are notified by the system and managed in the Event Flow.*

Name	Platform log-in
ID	UC.2
Actors	End User, CPO
Entry Condition	1) The actor is running the application. 2) The actor is already registered.
Event Flow	1) The actor navigates to the log-in page. 2) The actor fills the log-in form with username and password. 3) The actor submits the log-in. 4) The system redirects the actor to the explore page.
Exit Condition	The actor is logged-in correctly.
Exceptions	1) The provided username is incorrect. 2) The provided password is incorrect. *execptions caused by user errors are notified by the system and managed in the Event Flow.*

Name	Book a charge
ID	UC.3
Actors	End User
Entry Condition	1) The end user is running the application. 2) The end user is logged-in. 3) The end user is in the explore page.
Event Flow	1) The end user is looking for a charging station to charge his vehicle according to system's suggestions. 2) The end user selects the desired charging station. 3) The end user selects the date and the charging socket type of the charging station. 4) The end user selects the desired time slot, if it's free. 5) The end user presses the booking button.
Exit Condition	The end user has successfully booked the charge for his vehicle at the desired charging station to an assigned charging socket in the designed time slot.
Exceptions	1) The user on the booking page waits too long in selecting the desired time slot, this way another user books the desired time slot faster than the other. The first user does not see this change and therefore selects that time slot, so the system tells him through a notification that there has been an error and that he must restart from point 2 of the event flows.

Name	Manage booking
ID	UC.4
Actors	End User
Entry Condition	1) The end user is running the application. 2) The end user is logged-in. 3) The end user has already booked a charge. 4) The end user is in the bookings view page.
Event Flow	1) The end user views his current bookings. 2) The end user views his expired bookings. 3) The end user selects a booking. 4) The end user views all the details about the selected booking. 5) The end user views the QR-Code of the booking.
Exit Condition	The user has viewed the booking.
Exceptions	*None*

Name	Delete Booking
ID	UC.5
Actors	End User
Entry Condition	1) The end user is running the application. 2) The end user is logged-in. 3) The end user has already booked a charge. 4) The end user is in the bookings view page.
Event Flow	1) The end user selects the booking to delete. 2) The end user deletes the booking by pressing the appropriate button.
Exit Condition	The end user has deleted successfully the booking.
Exceptions	*None*

Name	Charging process
ID	UC.6
Actors	End User, Charging Socket
Entry Condition	1) The end user is running the application. 2) The end user is logged-in. 3) The end user has already booked a charge. 4) The end user has a QR-Code. 5) The charging socket is installed.
Event Flow	1) The end user goes to the assigned charging socket. 2) The end user scans the Qr-Code of the booking using the QR-Code scanner on the charging socket. 3) The system validates the booking. 4) The end user charges his vehicle. 4) The system notifies the user that the charge is completed. 5) The system subtracts automatically the price of the charge.
Exit Condition	The system allows the charging request and substratcs the related money.
Exceptions	1) The Qr-Code is invalid or expired, in this case the charging process doesn't start. 2) If the end user does not have enough money on his payment method, the system tries, periodically, to subtract the amount of charge.

Name	Purchase of energy
ID	UC.7
Actors	CPO, DSO
Entry Condition	1) The CPO is running the application. 2) The CPO is logged-in. 3) The CPO owns the charging station. 4) The DSO has an offer for the energy.
Event Flow	1) The CPO decides to buy energy for his charging station. 2) The CPO makes a purchase of energy from a DSO through the CPMS. 3) The CPO can decide whether to store the purchased energy. 4) The CPO can decide whether to distribute the purchased energy to charging sockets.
Exit Condition	The CPO has bought energy from the DSO.
Exceptions	None

Name	Manage charging station
ID	UC.8
Actors	CPO
Entry Condition	1) The CPO is running the application. 2) The CPO is logged-in. 3) The CPO owns the charging station.
Event Flow	1) The CPO visualizes the internal status of the station. 2) The CPO visualizes the external status of the station.
Exit Condition	The CPO has managed the status of his charging station.
Exceptions	None

Name	Add a charging station
ID	UC.9
Actors	CPO
Entry Condition	1) The CPO is running the application. 2) The CPO is logged-in. 3) The CPO is in the add charging station page.
Event Flow	1) The CPO decides to register a new charging station. 2) The CPO fills the form with the infrastructure data of the charging station. 3) The system stores the new charging station's data.
Exit Condition	A new charging station has been created by the CPO.
Exceptions	1) If the CPO fills the form with incorrect data, the system notifies him the error. 2) If the infrastructure data are already registered, the system doesn't store the data and notifies this error to the user.

3.2.3 Sequence Diagrams

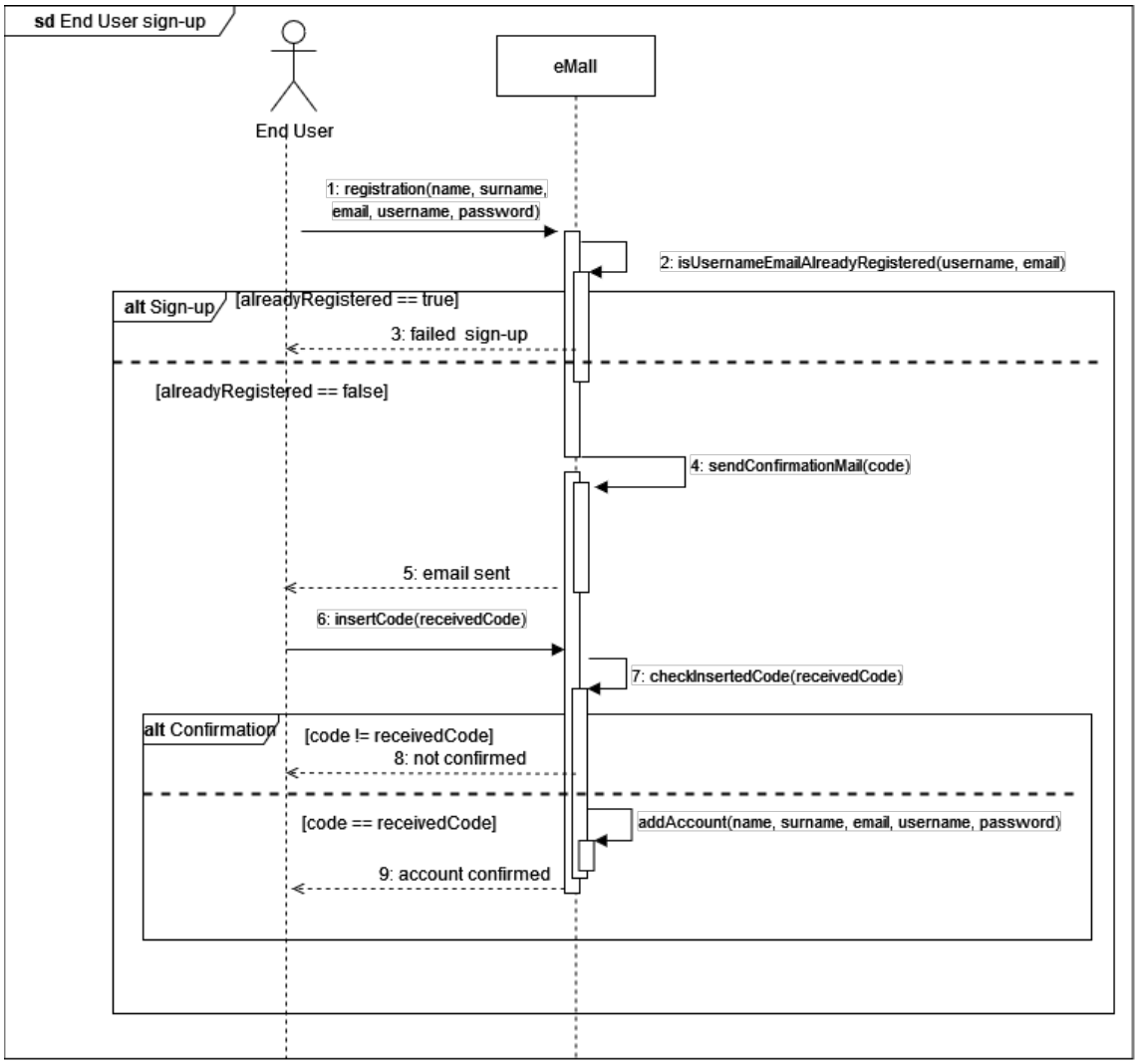


Figure 3.4: End user's registration sequence diagram.

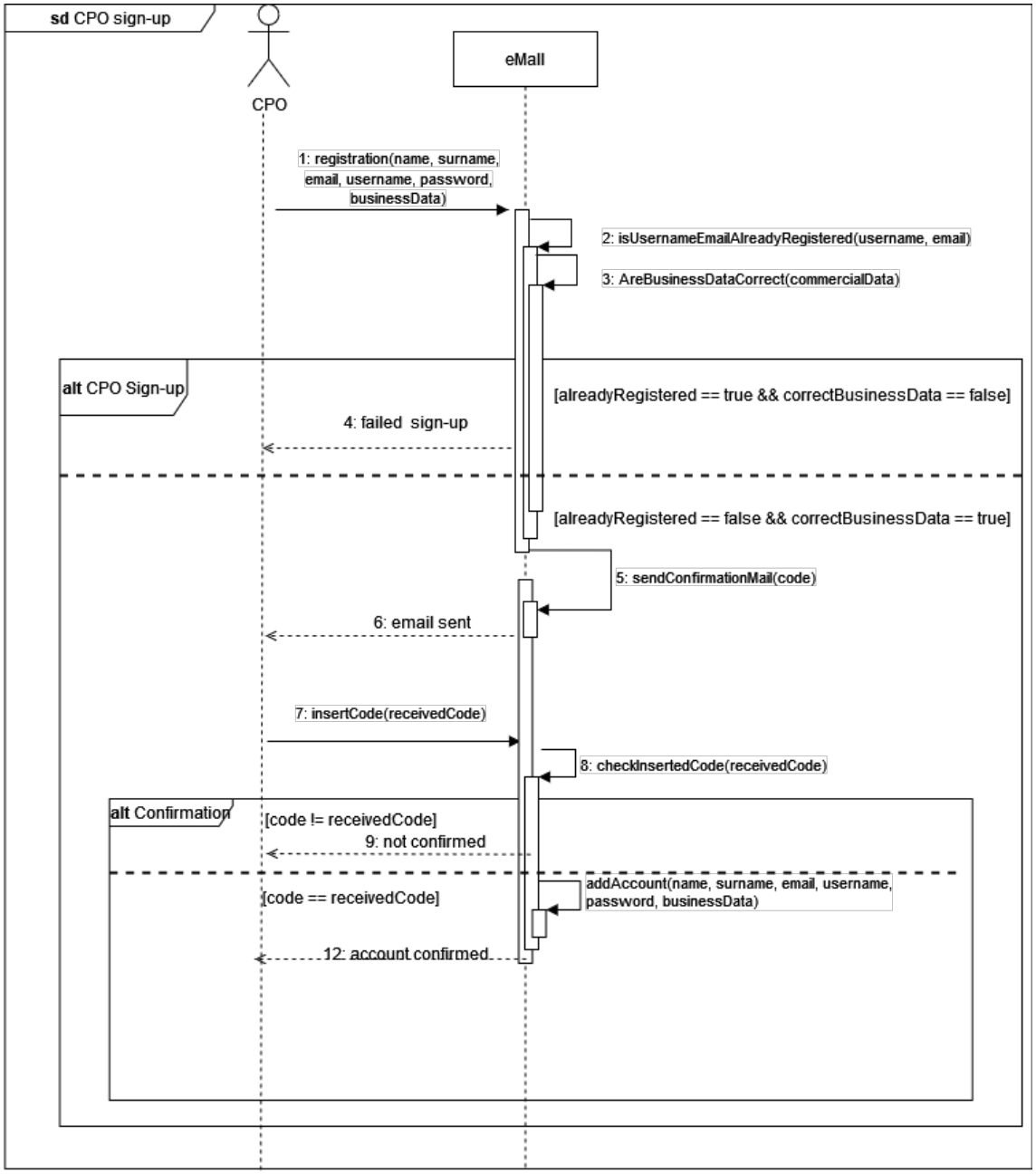


Figure 3.5: CPO's registration sequence diagram.

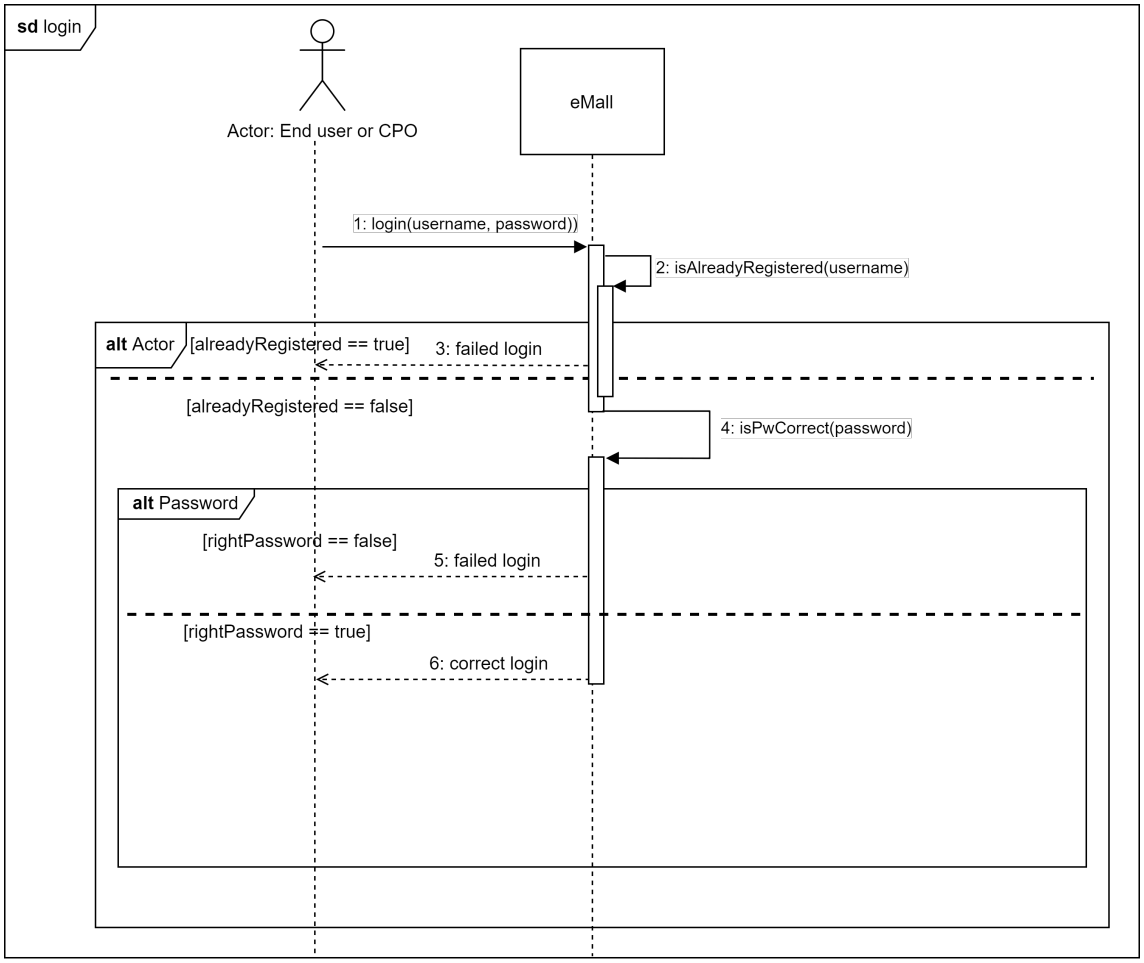


Figure 3.6: Login sequence diagram.

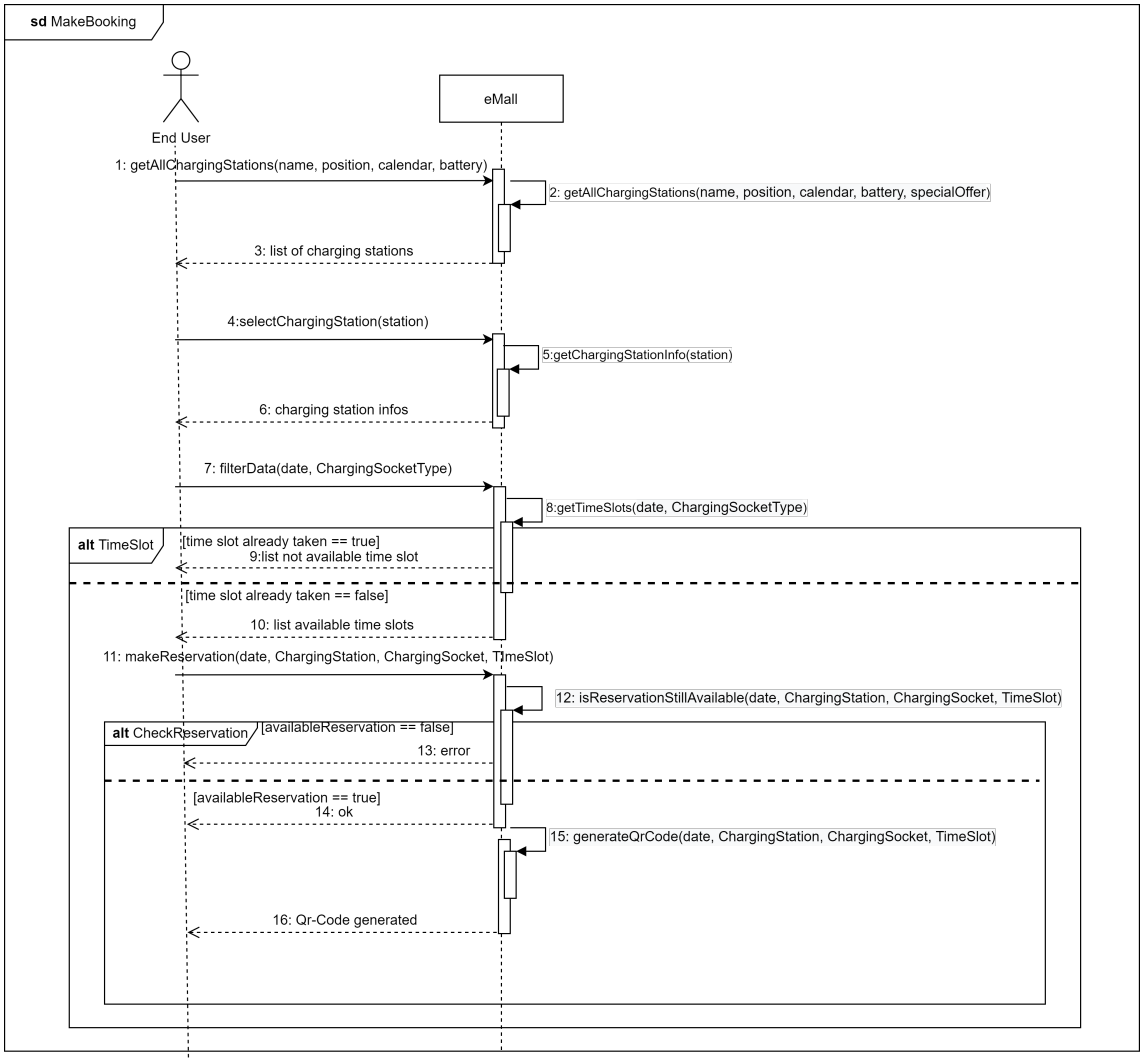


Figure 3.7: End user booking a charge sequence diagram.

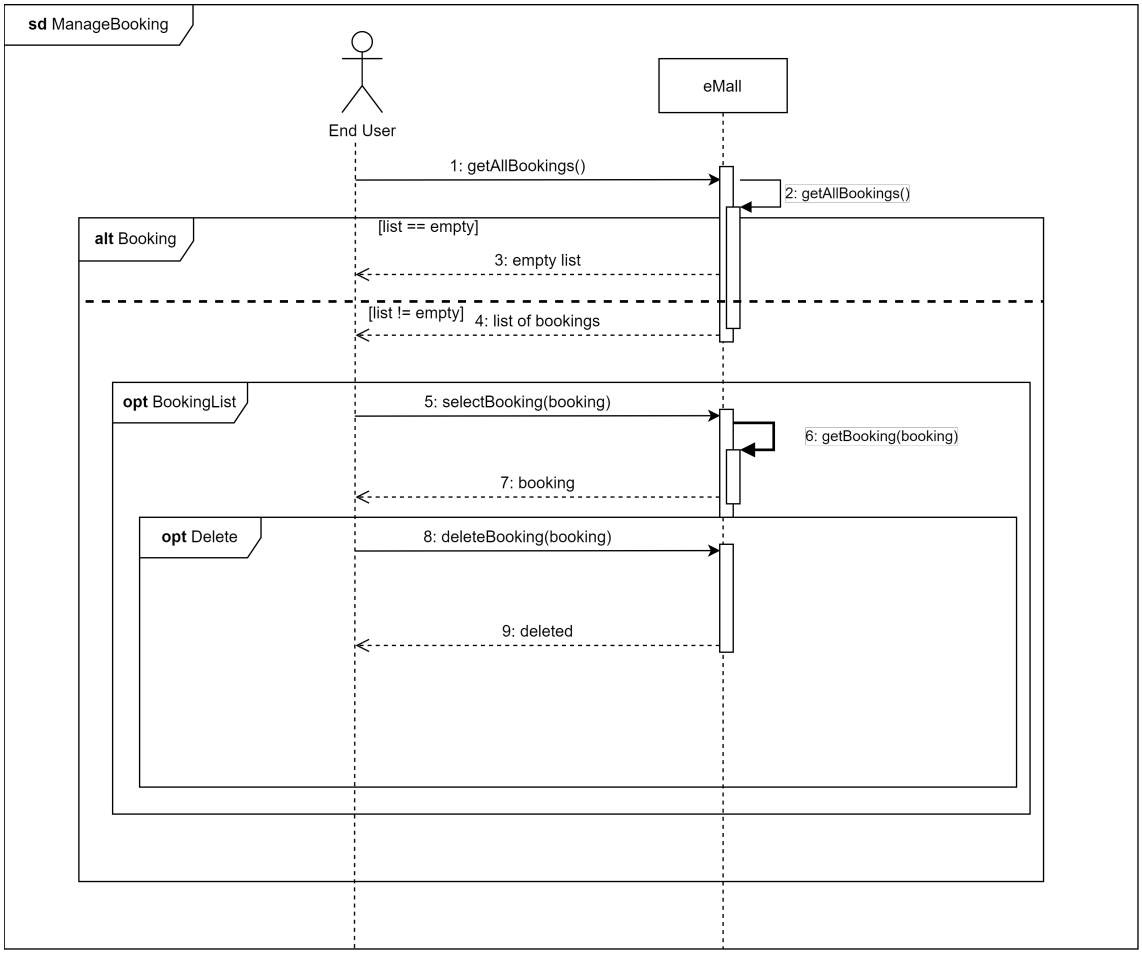


Figure 3.8: End user managing booking sequence diagram.

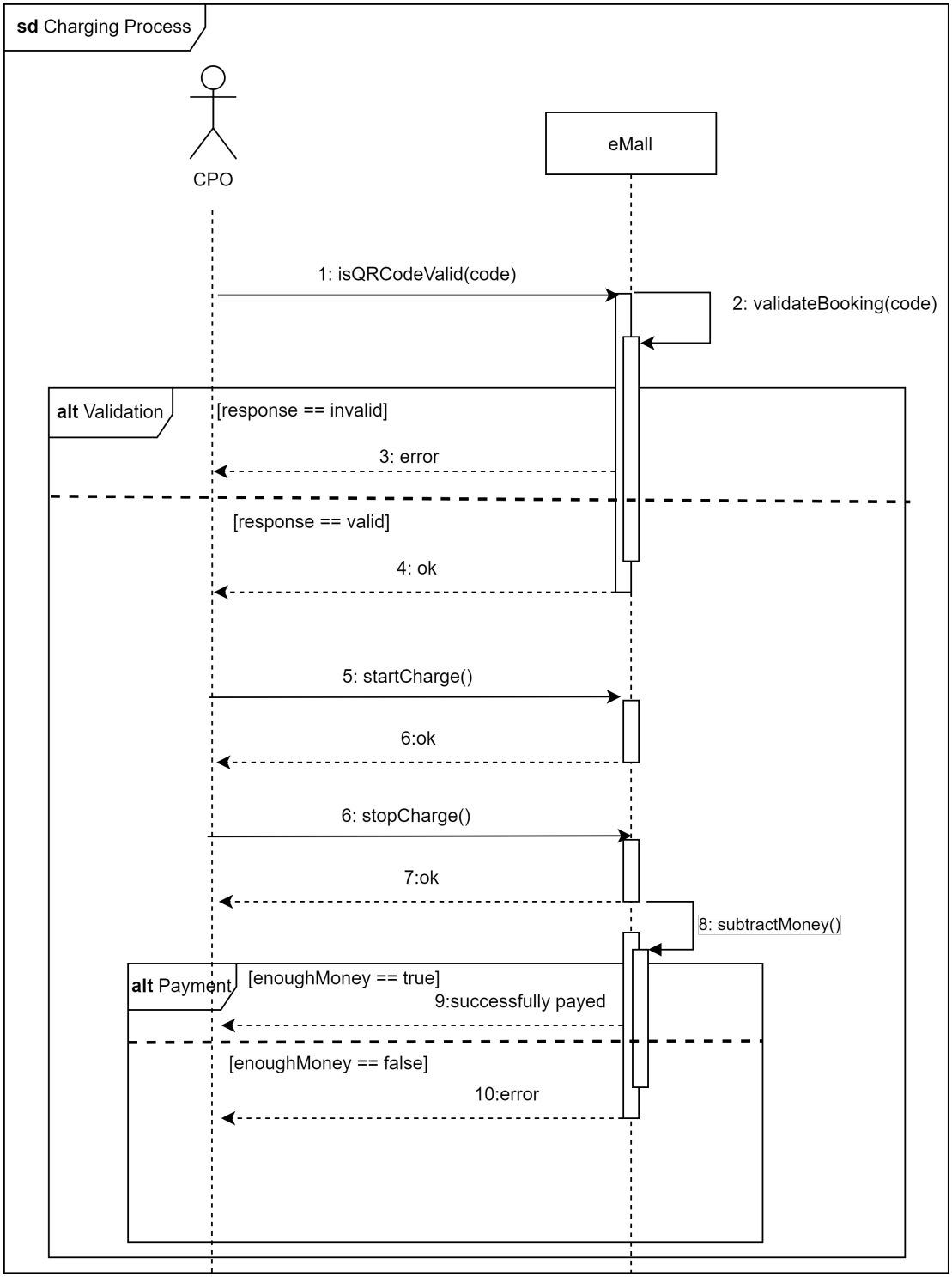


Figure 3.9: Charging process sequence diagram.

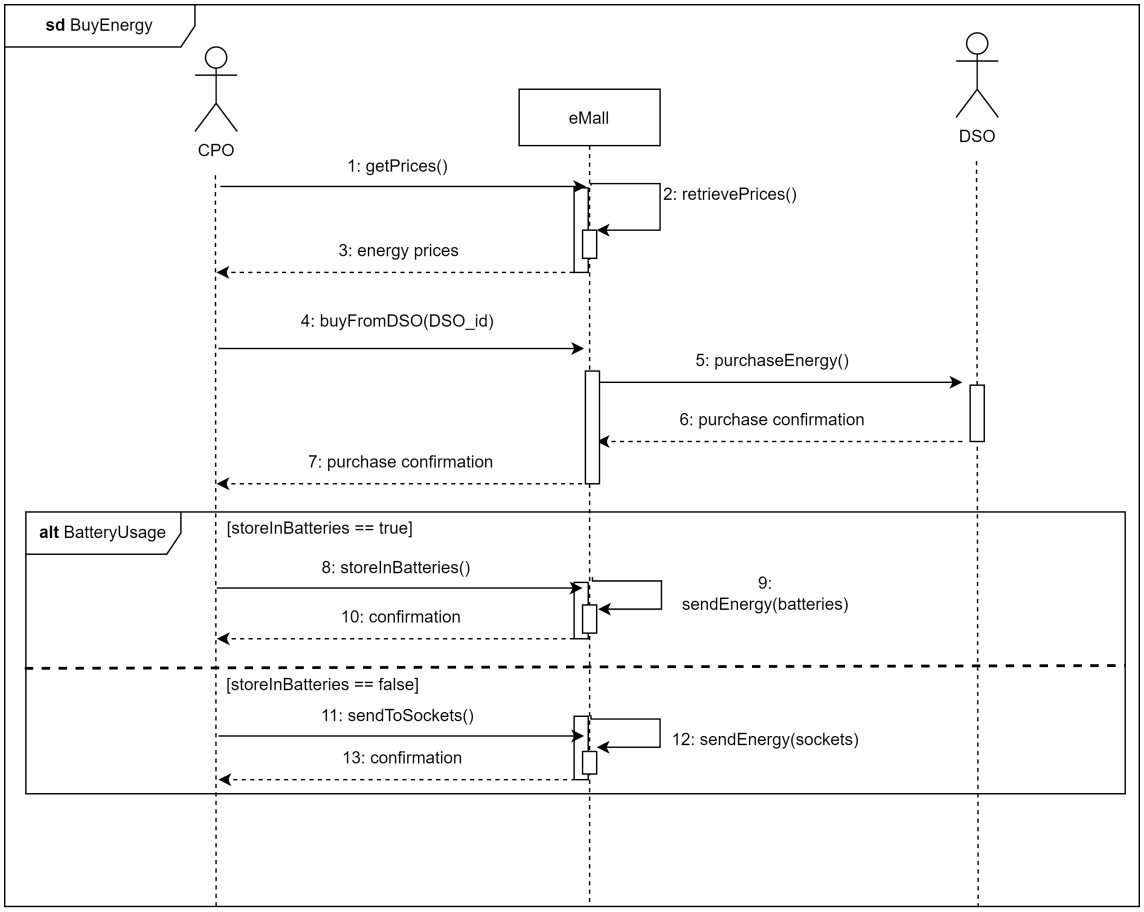


Figure 3.10: CPO puchasing energy sequence diagram.

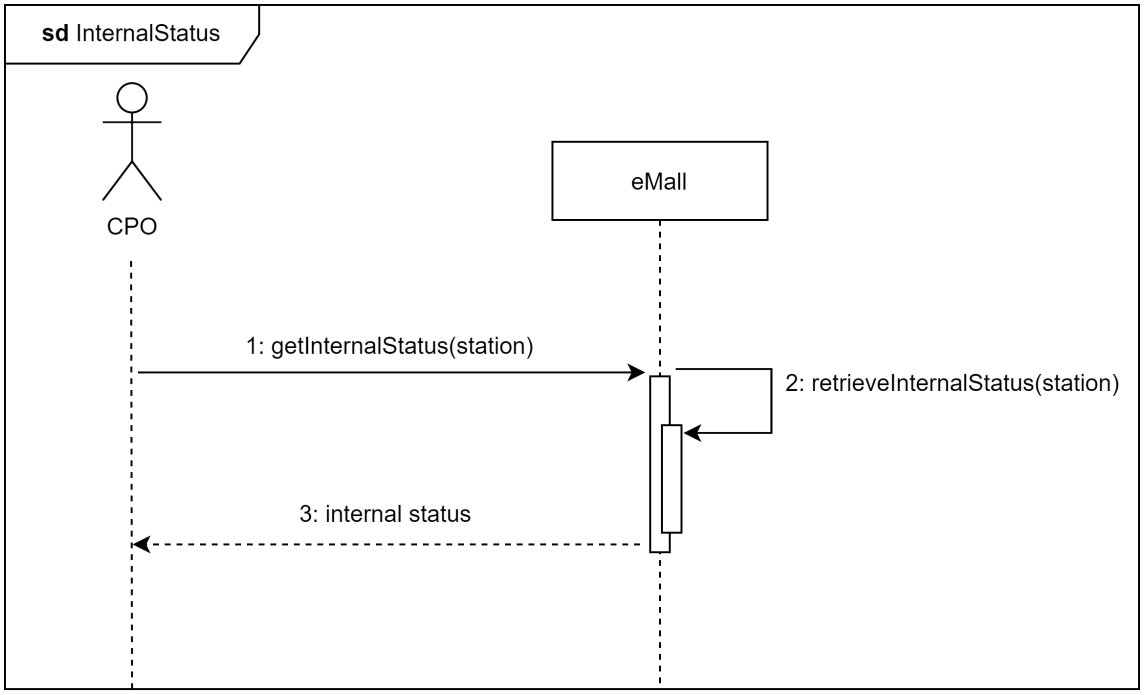


Figure 3.11: Internal status sequence diagram.

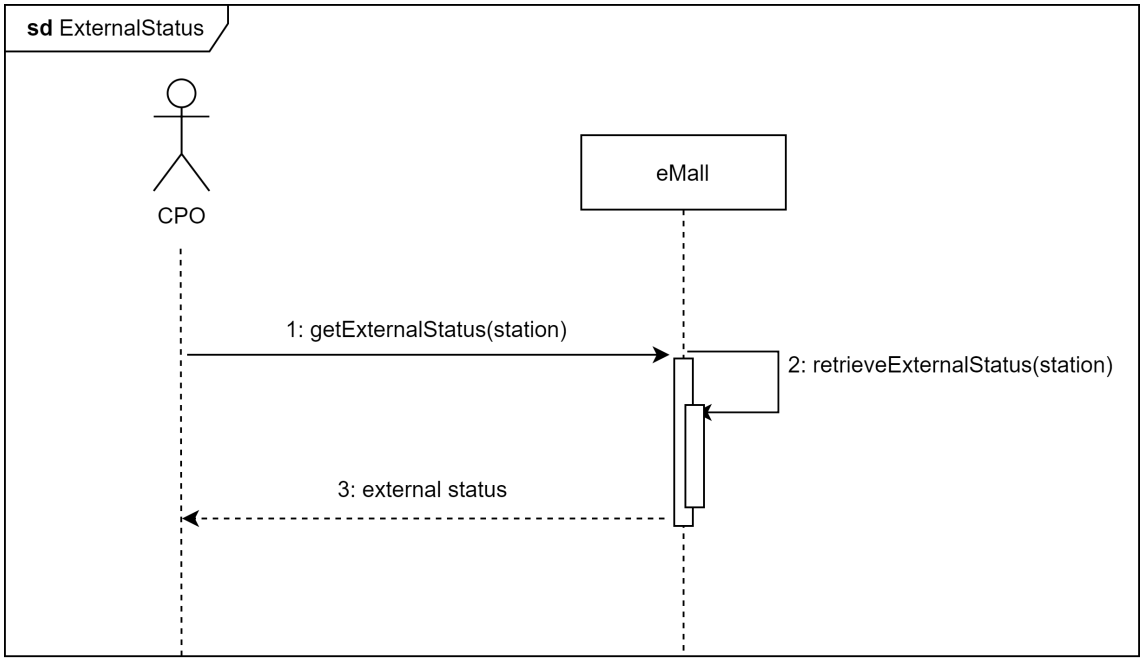


Figure 3.12: External status sequence diagram.

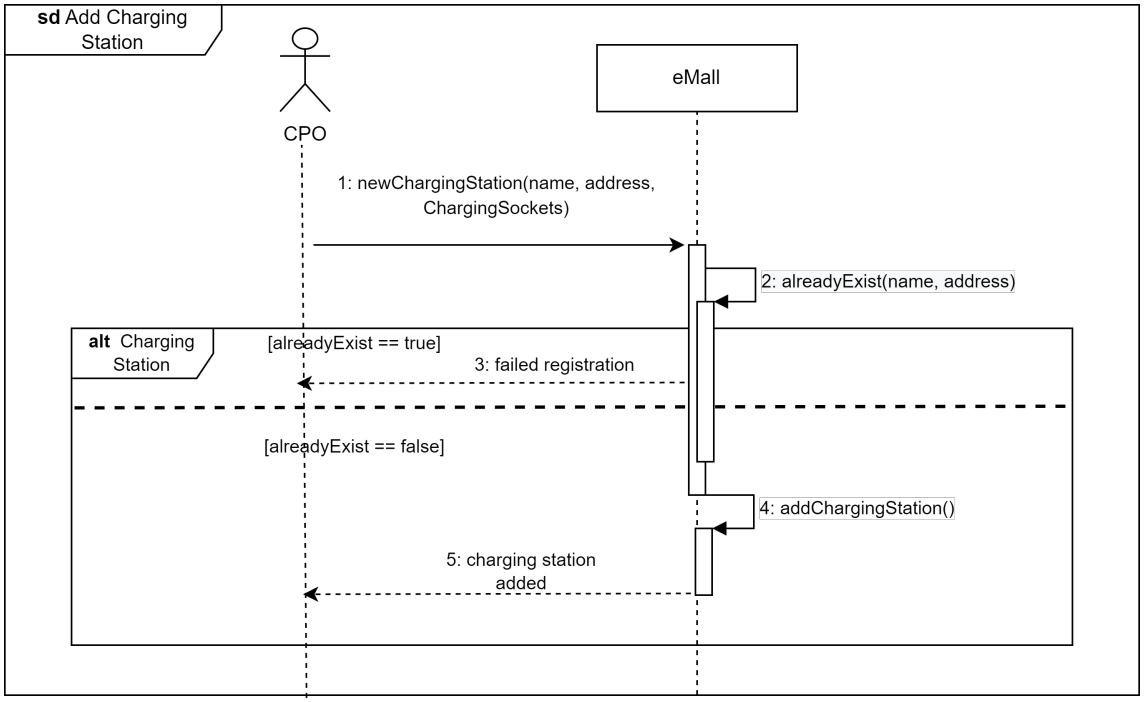


Figure 3.13: Add charging station sequence diagram.

List of functional requirements

Requirements	Description
R.1	The system must allow the end user to register himself to the system by providing all the mandatory information.
R.2	The system must allow the end user to register a payment method.
R.3	The system must verify the uniqueness of the mail and username of the end user to be registered.
R.4	The system must send a confirmation on the end user's email after his registration.
R.5	The system must allow the end user to log into their account by entering their username and password.
R.6	The system must allow the end user to book a charging socket of a certain charging station in a specific time slot.
R.7	The system must generate a QR-Code after the end user's reservation.
R.8	The system must show to the end user the suggested charging stations.
R.9	The system must show to the end user the available and unavailable time slots of a certain charging socket.
R.10	The system must access to the end user's location through GPS.
R.11	The system must access to the end user's calendar.
R.12	The system must access to the end user's vehicle status battery.
R.13	The system must allow the end user to delete his reservation.
R.14	The system must notify the end user when the charge of his vehicle is completed.

R.15	The system must subtract the cost of the charge from end user's payment method.
R.16	The system must send to the end user an email with the invoice for the used service.
R.17	The system must allow the end user to book a time slot.
R.18	The system must stop the charge when the upper bound of time slot is exceeded.
R.19	The system must allow the end user to see his current bookings.
R.20	The system must allow the end user to search for a certain charging station.
R.21	The system must allow the end user to stop the charging process.
R.22	The system must allow the end user to start the charging process.
R.23	The system must allow the end user to monitor the charging process.
R.24	The system must prevent the end user from reserving a charging socket in a certain time slot, if he already has another reservation in that time slot.
R.25	The system must prevent at least two different end users from reserving the same charging socket in the same time slot.
R.26	The system must allow the CPO to register himself to the system by providing all the mandatory and commercial information.
R.27	The system must verify the uniqueness of the mail of the CPO to be registered.
R.28	The system must verify the correctness of the CPO's commercial data.
R.29	The system must send a confirmation on the CPO's email after his registration.
R.30	The system must allow the CPO to log into their account by entering their username and password.
R.31	The system must allow the CPO to know the internal and external status of charging sockets of his charging stations.
R.32	The system must allow the CPO to monitor the charging process.
R.33	The system must allow the CPO to acquire information about the price of energy from DSOs.
R.34	The system must allow the CPO to acquire energy from DSOs.
R.35	The system must allow the CPO to decide whether to store or not energy.
R.36	The system must allow the CPO to decide whether to use available energy in the batteries.
R.37	The system must forbid to charge the vehicle to a user that scans a QR-Code which is not valid (expired, wrong etc.).
R.38	The system must indicate whether a slot is booked or not, and it must forbid to book an already booked time slot.
R.39	The system must manage the concurrent booking of a slot: if a user waits too much to book a slot and meanwhile another user books that slot, the system must notify the first user when he tries to perform the booking.

R.40	The system must allow the CPO to add a new charging station.
R.41	The system must allow the CPO to view, via the CPMS, the charge percentage of the battery cluster.
R.42	If the user initiates charging and there is not enough charge in the battery cluster, the system, to avoid service disruption, allows the charging station to receive power directly from the DSO channel.

3.2.4 Mapping on requirements

Goals	Domain Assumptions	Requirements
G.1	D.1, D.2, D.3, D.4, D.5, D.12, D.15	R.1, R.3, R.4, R.5, R.8, R.10, R.11, R.12, R.20, R.24, R.25
G.2	D.1, D.3, D.14, D.15, D.17	R.1, R.3, R.4, R.5, R.6, R.7, R.9, R.13, R.17, R.19, R.38, R.39
G.3	D.1, D.3, D.6, D.7, D.8, D.15, D.18	R.1, R.3, R.4, R.5, R.14, R.18, R.21, R.22, R.23, R.37
G.4	D.1, D.3, D.15, D.24, D.26	R.1, R.2, R.3, R.4, R.5, R.15, R.16
G.5	D.9, D.10, D.11, D.12, D.13, D.16, D.19, D.20, D.22, D.23, D.28, D.29, D.30	R.26, R.27, R.28, R.29, R.30, R.31, R.32, R.40, R.41, R.42
G.6	D.9, D.10, D.11, D.16, D.21, D.28, D.29, D.30	R.26, R.27, R.28, R.29, R.33, R.34
G.7	D.9, D.10, D.11, D.13, D.15, D.25, D.27, D.28, D.29, D.30	R.26, R.27, R.28, R.29, R.35, R.36, R.42

Why this mapping? The table above reports the mapping of assuptions and requirements on the goals. It is useful so that you have a clear and immediate idea about which goals are involved in a certain assumption or requirement. In general, we have a division between the end user related goals (G1, G2, G3, G4) and the CPO goals (G5, G6, G7). As we can see from the table above some requirements / assumption are repeated, this is because some of them are essential for all the goals, for example the end user login or the CPO login.

3.2.5 Traceability matrix

Raw ID	Goal ID	Req ID	Use Case ID
r1	G.1	R.1	UC.1
r2	G.1	R.3	UC.1
r3	G.1	R.4	UC.1
r4	G.1	R.5	UC.2
r5	G.1	R.8	UC.3
r6	G.1	R.9	UC.3
r7	G.1	R.10	UC.3
r8	G.1	R.11	UC.3
r9	G.1	R.12	UC.3
r10	G.1	R.20	UC.3
r11	G.1	R.24	UC.3
r12	G.1	R.25	UC.3
r13	G.2	R.1	UC.1
r14	G.2	R.3	UC.1
r15	G.2	R.4	UC.1
r16	G.2	R.5	UC.2
r17	G.2	R.6	UC.3
r18	G.2	R.7	UC.3
r19	G.2	R.9	UC.3
r20	G.2	R.13	UC.5
r21	G.2	R.17	UC.3

r22	G.2	R.19	UC.4
r23	G.2	R.38	UC.3
r24	G.2	R.39	UC.3
r25	G.3	R.1	UC.1
r26	G.3	R.3	UC.1
r27	G.3	R.4	UC.1
r28	G.3	R.5	UC.2
r29	G.3	R.14	UC.6
r30	G.3	R.18	UC.6
r31	G.3	R.21	UC.6
r32	G.3	R.22	UC.6
r33	G.3	R.23	UC.6
r34	G.3	R.37	UC.6
r35	G.4	R.1	UC.1
r36	G.4	R.2	UC.1
r37	G.4	R.3	UC.1
r38	G.4	R.4	UC.1
r39	G.4	R.5	UC.2
r40	G.4	R.15	UC.6
r41	G.4	R.16	UC.6
r42	G.5	R.26	UC.1
r43	G.5	R.27	UC.1
r44	G.5	R.28	UC.1
r45	G.5	R.29	UC.1
r46	G.5	R.30	UC.2
r47	G.5	R.31	UC.8
r48	G.5	R.32	UC.8
r49	G.5	R.40	UC.9
r50	G.5	R.41	UC.8
r51	G.5	R.42	UC.7
r52	G.6	R.26	UC.1
r53	G.6	R.27	UC.1
r54	G.6	R.28	UC.1
r55	G.6	R.29	UC.1
r56	G.6	R.30	UC.2
r57	G.6	R.33	UC.7
r58	G.6	R.34	UC.7
r59	G.7	R.26	UC.1
r60	G.7	R.27	UC.1
r61	G.7	R.28	UC.1
r62	G.7	R.29	UC.1
r63	G.7	R.30	UC.2
r64	G.7	R.35	UC.7
r65	G.7	R.36	UC.7
r66	G.7	R.42	UC.7

3.3 Performance Requirements

This section presents the performance requirements which need to be fulfilled to make the whole system as usable as possible.

The eMall system concentrates the majority of the computation server-side to maximize usability, and speed-up as much as possible all the operations. The service works 24/7 but as we can imagine the most of the usage is during the daytime, so servers maintenance will be made overnight. The only interaction between the user and the system that is not managed by the platform is the QR-Code scanning operation, so at least the 95% of the time the QR-Code scanner must scan the QR-Code in less than 1 second.

3.4 Design Constraints

3.4.1 Standards compliance

Due to the various interactions between platform's parts (eMSP, CPMS, servers etc.) some standards are followed to grant consistency in these interactions. First of all the data is stored / transferred using standards defined by the HTTP protocol. In particular all the interactions are made through this protocol: obviously, it will used a JSON standard to encourage the use of a RESTful approach.

3.4.2 Hardware limitations

The hardware constraint on the eMSP part are related to the absence of hardware modules such as GPS, Wi-Fi/LTE modules. The whole system is also constrained by the server-side overloading, this can be handled increasing the server-side scalability in terms of hardware power.

3.4.3 Any other constraint

Obviously the system usability relies also on the users' common sense, unfortunately this constraint can't be handled by the IT infrastructure.

3.5 Software System Attributes

3.5.1 Reliability

Non Functional Requirements	Description
NFR.1	eMall encourages the spread of electric vehicles by regulating their charging processes throughout the day.
NFR.2	eMall users must therefore be able to book a charge of their vehicle at a charging without running into simultaneous bookings with other users.
NFR.3	The system proves its reliability by managing the aforementioned problem in 99.9% of cases.

3.5.2 Availability

Non Functional Requirements	Description
NFR.1	End users and CPOs must be able to act freely on the platform without logistical constraints (e.g. at any time of day)
NFR.2	In the case of unplanned system downtime all features will be available as soon as possible: to limit this unpleasant occurrence it is important that eMall is equipped with a valid infrastructure to rely on, i.e. redundant servers.

3.5.3 Security

Non Functional Requirements	Description
NFR.1	Personal data provided by end users are not subject to disclosure or commercial purposes.
NFR.2	All types of users involved in using the system have specific privileges based on their role, which is identified through the login operation.
NFR.3	All data and information that are transferred and stored are subject to strong encryption (e.g. SSL/HTTPS, SHA-256 hash, etc.).
NFR.4	eMall includes a payment functionality which is handled through SCA.

3.5.4 Maintainability

Non Functional Requirements	Description
NFR.1	The entire system must be built in an intelligent way to ensure a correct and prosperous temporal evolution of the software: it is necessary to design and develop any future aspects and additions.
NFR.2	To avoid inconvenience in solving any type of problem (e.g. server downtime), maintenance services are carried out during the night (10:00 p.m - 6:00 a.m.).

3.5.5 Portability

Non Functional Requirements	Description
NFR.1	Due to the fact that eMall is a distributed system, and it doesn't rely on a specific hardware or software, it can be used / accessed in multiples way.

Chapter 4

Formal Analysis Using Alloy

In this chapter the Alloy model of the eMall system is implemented, describing the main constraints and focusing in particular on the consistency of the world.
For this purpose, 2 different sub-worlds are created which allow you to view their characteristics.

```
-- Signatures

sig Username {}
sig Password {}
sig Email {}
sig Name {}
sig Surname {}
sig BusinessData {}
sig DSOId {}
sig DSOName {}
sig SocketPrice {}
sig DSOPrice {}
sig CardNumber {}
sig Cvv {}
sig Date {}
sig ExpiredDate {}
sig BookingCode {}
sig ChargingStationId {}
sig ChargingStationName {}
sig Address {}
sig BatteryClusterPercentage {}
sig BatteryClusterCapacity {}
abstract sig BookingStatus {}
one sig Valid extends BookingStatus {}
one sig Expired extends BookingStatus {}
abstract sig ChargingSocketStatus {}
one sig Free extends ChargingSocketStatus {}
one sig Busy extends ChargingSocketStatus {}
abstract sig SocketType {}
one sig Slow, Fast, Rapid extends SocketType {}

abstract sig User {
        username: Username,
        password: Password,
        email: Email,
        name: Name,
        surname: Surname
}

sig EndUser extends User {
        paymentMethod: some PaymentMethod
}

sig PaymentMethod {
        cardNumber: CardNumber,
        cvv: Cvv,
```

```
        expireDate: ExpiredDate
}

sig ChargingStation {
        id: ChargingStationId,
        name: ChargingStationName,
        address: Address,
        batteryClusterPercentage: one BatteryClusterPercentage,
        batteryClusterCapacity: one BatteryClusterCapacity,
        sockets: some ChargingSocket
}

sig ChargingSocket {
        number: one Int,
        type: SocketType,
        status: ChargingSocketStatus,
        price: SocketPrice
} { number ≥ 1 }

sig DSO {
        id: DSOId,
        name: DSOName,
        energyPrice: DSOPrice
}

sig CPO extends User {
        businessData: BusinessData,
        stations: some ChargingStation,
        CPO_DSOs: some DSO
}

sig Booking {
        code: BookingCode,
        date: Date,
        socketNumber: one ChargingSocket,
        start: one Int,
        end: one Int,
        user: one EndUser,
        station: one ChargingStation,
        status: BookingStatus
} {start < end ∧ start ≥ 0 ∧ end ≤ 24 }

-- Facts

-- User
fact usernamesAreUnique {
        no disj u1, u2: User | u1.username = u2.username
}

fact emailsAreUnique {
        no disj u1, u2: User | u1.email = u2.email
}

fact usernameExistsOnlyWithUser {
        all un: Username | one u: User | un in u.username
}

fact emailExistsOnlyWithUser {
        all el: Email | one u: User | u.email in el
}

fact passwordExistsOnlyWithUser {
        all pw: Password | one u: User | u.password in pw
}

fact nameExistsOnlyWithUser {
        all nm: Name | one u: User | u.name in nm
```

```
}

fact surnameExistsOnlyWithUser {
        all sm: Surname | one u : User | u.surname in sm
}

-- CPO
fact businessDataAreUnique {
        no disj c1, c2: CPO | c1.businessData = c2.businessData
}

fact businessDataExistsOnlyWithCpo {
        all bd: BusinessData | one cpo: CPO | cpo.businessData in bd
}

fact chargingStationsExistsOnlyWithCpo {
        all cg: ChargingStation | one cpo: CPO | cg in cpo.stations
}

-- Booking
fact allBookingsAreUnique {
        no disj b1, b2: Booking | b1.code = b2.code
}

fact noBookingOverlapping {
        all disj b1, b2 : Booking | not (b1.date = b2.date ∧ b1.station = b2.
            ↪ station ∧ b1.socketNumber = b2.socketNumber ∧
                                                b2.start ≥ b1.start ∧ b2.end ≤ b1.
                                                    ↪ end  ∨
                                                b2.start ≥ b1.start ∧ b2.end ≥ b1.
                                                    ↪ end ∧ b2.start < b1.end  ∨
                                                b2.start =< b1.start ∧ b2.end ≥ b1.
                                                    ↪ end  ∨
                                                b2.start =< b1.start ∧ b2.end =< b1.
                                                    ↪ end ∧ b1.start  < b2.end)
}

fact codeExistsOnlyWithBooking {
        all cd: BookingCode | one b: Booking | cd in b.code
}

fact onlyOneBookingStatusAtTime {
        all b: Booking | one bst: BookingStatus | bst in b.status
}

-- Charging Station
fact chargingStationIdAreUnique {
        no disj cs1, cs2: ChargingStation | cs1.id = cs2.id
}

fact chargingStationAddressesAreUnique {
        no disj cs1, cs2: ChargingStation | cs1.address = cs2.address
}

fact oneChargingStationBelongToOneCPO {
        all cs: ChargingStation | one c: CPO | cs in c.stations
}

fact nameExistsOnlyWithChargingStation {
        all nm: ChargingStationName | one cs: ChargingStation | nm in cs.name
}

fact idExistsOnlyWithChargingStation {
        all i: ChargingStationId | one cs: ChargingStation | i in cs.id
}

fact addressExistsOnlyWithChargingStation {
```

```
        all addr: Address | one cs: ChargingStation | addr in cs.address
}

fact batteryClusterPercentageExistsOnlyWithChargingStation {
        all bcp: BatteryClusterPercentage | one cs: ChargingStation | bcp in cs.
            ↪ batteryClusterPercentage
}

fact batteryClusterCapacityExistsOnlyWithChargingStation {
        all bcc: BatteryClusterCapacity | one cs: ChargingStation | bcc in cs.
            ↪ batteryClusterCapacity
}

-- Charging Socket
fact socketIdAreUnique {
        no disj cs1, cs2: ChargingSocket | one cs: ChargingStation | cs1.number =
            ↪ cs2.number ∧ cs1 in cs.sockets ∧ cs2 in cs.sockets
}

fact oneChargingSocketBelongToOneChargingStation {
        all csk: ChargingSocket | one cs: ChargingStation | csk in cs.sockets
}

fact priceExistsOnlyWithChargingSocket {
        all pr: SocketPrice | one cs: ChargingSocket | pr in cs.price
}

fact onlyOneStatusAtTime {
        all cs: ChargingSocket | one css: ChargingSocketStatus | css in cs.status
}

-- Payment Method
fact paymentMethodExistsOnlyWithEndUser {
        all pm: PaymentMethod | some u: EndUser | pm in u.paymentMethod
}

fact numberExistsOnlyWithPaymentMethod {
        all cn: CardNumber | one pm: PaymentMethod | cn in pm.cardNumber
}

fact cvvExistsOnlyWithPaymentMethod {
        all cv: Cvv | one pm: PaymentMethod | cv in pm.cvv
}

fact dateExistsOnlyWithPaymentMethod {
        all dt: ExpiredDate | one pm: PaymentMethod | dt in pm.expireDate
}

-- DSO
fact idExistsOnlyWithDSO {
        all i: DSOId | one dso: DSO | i in dso.id
}

fact nameExistsOnlyWithDSO {
        all nm: DSOName | one dso: DSO | nm in dso.name
}

fact energyPriceExistsOnlyWithDSO {
        all ep: DSOPrice | one dso: DSO | ep in dso.energyPrice
}

-- Predicates

pred world1 {
        #Booking = 2
        #EndUser = 1
        #ChargingStation = 1
```

```
        #ChargingSocket = 1
        #Date = 1
        #DSO = 1

        all b: Booking |  (b.status = Valid)
        all cs: ChargingSocket |  (cs.status = Free or cs.status = Busy)
        all cs: ChargingSocket |  (cs.type = Fast or cs.type = Slow or cs.type =
            ↪ Rapid)
}

run world1 for 4 but 6 Int


pred world2 {
        #ChargingStation = 2
        #ChargingSocket = 6
        #CPO = 2
        #DSO = 1
        #Date = 0
        #EndUser = 0

        all cs: ChargingSocket |  (cs.status = Busy or cs.status = Free)
        all cs: ChargingSocket |  (cs.type = Slow or cs.type = Fast or cs.type =
            ↪ Rapid)
}

run world2 for 6

-- Assertions

assert allBookingsNoOverlapped {
        all disj b1, b2 : Booking | ((b1.date ≠ b2.date ∨ b1.station ≠ b2.station ∨
            ↪  b1.socketNumber ≠ b2.socketNumber) ∨
                                                b2.start ≥ b1.end ∨ b1.start ≥ b2.
                                                    ↪ end)
}
check allBookingsNoOverlapped

assert noEqualChargingSocketInSameChargingStation {
        all disj cs1, cs2: ChargingSocket | all cs: ChargingStation |  ((cs1 in cs.
            ↪ sockets ∧ cs2 in cs.sockets) implies cs1.number ≠ cs2.number)
}
check noEqualChargingSocketInSameChargingStation
```

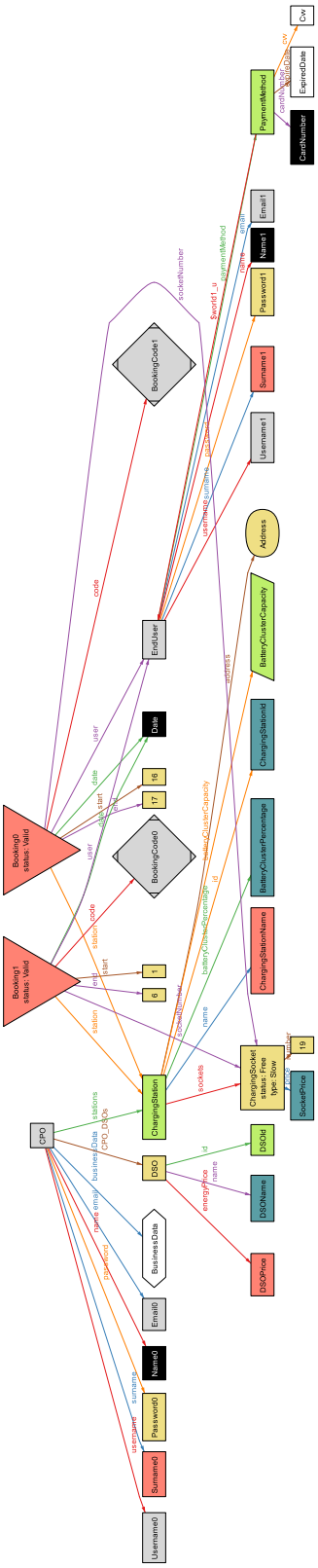


Figure 4.1: World 1

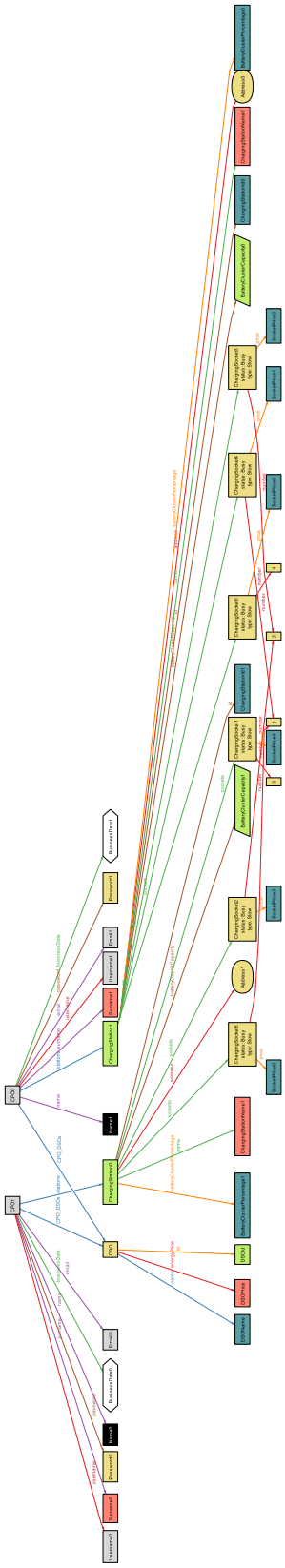


Figure 4.2: World 2

4.1 First World

The goal of the first world in the figure 4.1 consists in representing the booking function: the valid booking functions are then displayed relating to a charging station which contains a charging socket (which can be busy or free and fast, slow or rapid).
It is important to note that the bookings made for the same day are not overlapping in terms of timeslots.

4.2 Second World

The goal of the second world in the figure 4.2 is to represent the structure of the charging stations and related charging sockets, paying particular attention to the fact that there are no identical charging socket codes in the same charging stations.

Chapter 5

Effort Spent

5.1 Team discussions

Topic	Hours
General discussion	1.5h
Introduction	3h
Briefing on overall description	2.5h
Briefing on requirements	3.5h
Briefing on alloy	3h
Final revision	8h

5.2 Balestrieri Niccolò

Topic	Hours
Product perspective	2h
Mapping on requirements	1.5h
Product functions	3h
Functional requirements	2.5h
Alloy	2.5h
Mockups design	3h

5.3 Bertogalli Andrea

Topic	Hours
External interface requirements	3h
Performance requirements	1.5h
Software System Attributes	1.5h
Mockups design	3h
Alloy	3h
Functional requirements	2.5h

5.4 Tombini Nicolò

Topic	Hours
Mapping on requirements	3h
Class, Sequence, State diagram	2h
Scenarios	2h
Alloy	3h
Design constraints	1.5h
Mockups design	3h

Chapter 6

References

- *IEEE Software Requirements Speification Template.*
- *Atlante.*
- *Electric Vehicle CPMS and Secondary Substation Management.*