

eMall – e-Mobility for All

Design Document

Beliakov Maksim
10913275

Aliyev Rustam
10897101

Yizhou Wu
10926468

January 8, 2023



POLITECNICO
MILANO 1863

Contents

1. Introduction.....	4
1.1 Purpose.....	4
1.2 Scope	4
1.3 Definitions, Acronyms, Abbreviations	4
1.3.1 Definitions	4
1.3.2 Abbreviations.....	4
1.4 Revision History	5
1.5 Reference Document.....	5
1.6 Document Structure	5
2. Architectural design.....	6
2.1 Overview: High-level components and their interaction	6
2.2 Component view	7
2.3 Deployment view.....	9
2.4 Runtime View	9
2.4.1 Driver Registration.....	10
2.4.2 General Charging Socket Booking	11
2.4.3 Charging Payment.....	12
2.4.4 Start Charging	12
2.4.5 Accept Suggestion	13
2.4.6 Login on CPMS (for CPO and CPO Operator).....	13
2.4.7 Create operator profile.....	14
2.4.8 Configure Charging Point.....	14
2.4.9 View and configure Charging Socket's detail	15
2.4.10 Interaction with eMSP system.....	16
2.5 Component interfaces	17
2.6 Selected architectural styles and patterns	18
3. User interface design.....	19
3.1 eMSP User interface design.....	19
3.2 CPMS User Interface Mockups	22
4. Requirements traceability	29
4.1 Mapping of components on requirements:	31
5. Implementation, integration and test plan	33
5.1 Implementation.....	33
5.2 Integration and test plan	33
6. Effort spent.....	40
6.1 Beliakov Maksim.....	40

6.2 Aliyev Rustam 40

6.3 Yizhou Wu..... 40

7. References 40

1. Introduction

1.1 Purpose

However, in order to widely spread the use of electrical vehicles among the population, the corresponding servicing infrastructure is required. First, charging points infrastructure is to be developed so that the users of electrical mobility are able to conveniently charge their cars. Therefore, the corresponding managing system (eMSP) for this infrastructure should be developed which must take care of providing the end user with the information about charging points nearby and their features, managing the charging actions such as booking, starting, finishing, payment of charging. Also, this managing system interacts with the charging point managing system (CPMS) which is responsible for setting the status of the point, charging a vehicle, interaction with an energy distribution system operator (DSOs).

As this document should guide the development of the system, several aspects should be concerned here. The high-level description of the components of the system and the way they interact is provided along with the architectural design. Also, the document contains approximate mockups for UI-design of the application and the plan for the implementation of the system and components interconnection.

1.2 Scope

This document concerns all the features provided by the eMSP system and the CPMS system, and the interactions between them.

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

Definition	Description
Special offer	Offer w.r.t user is used as a general term for any kind of commercial propositions e.g. discounts.
Application	The mobile application through which user gets access to features provided by eMSP

1.3.2 Abbreviations

Abbreviation	Description
RASD	Requirements Analysis and Specification Document
WP	World Phenomena
SP	Shared Phenomena

Abbreviation	Description
eMSP	Electric-mobility service provider
CPMS	Charging Point Management System
CP	Charging point (station)
CPO	Charging Point Operator
DSO	Distribution System Operator
GX	Goal number X
DX	Domain assumption number X
RX	Requirement number X

1.4 Revision History

Version	Date	Notes
0.1	2022/12/28	First parts of documents
0.2	2023/01/06	Integrated other parts.
1.0	2023/01/08	Refined the document and first release

1.5 Reference Document

- The specification document “01. Assignment RDD AY 2022-2023_v3”
- The Requirement Analysis and Specification Document “RASD2.pdf”

1.6 Document Structure

This document is structured in the following way:

1. The first chapter is an introduction and overview of the project, setting the context that led to its development, the goals to be achieved, and a general description of its functionality.
2. The second section describes the architectural design of the system to be. It gives a high-level overview of the architecture breaking each part down into components. The components are described as well as their interdependence in the component diagram. Besides, the section contains a component interface diagram, a deployment view and sequence diagrams describing the interactions between components.
3. In the third section UI design mockups of the eMSP and CPMS are presented.
4. The fourth section provides the requirement traceability matrix, where each of the components described in the second section is mapped to the requirements specified in the RASD. The mapping is based on whether the component contributes to the fulfilment of the requirement.
5. Section five describes the suggested implementation and intergrtion plan for the system.
6. Section six contains the effort spent on this report by the authors.
7. Section seven contains the references used.

2. Architectural design

2.1 Overview: High-level components and their interaction

Considering the fact, that this project is supposed to be used by hundreds of thousands of owners or electrical vehicles, it was decided to separate the project into several layers (tiers). Below is the list of them:

- **Presentation**

This layer is responsible for the interaction with the clients (User and CPOs). For the users it will be very light mobile application, in case of CPOs (Who are supposed to be trained stuff and should be able to control a lot of stuff) it will be heavy desktop application.

- **Business Logic**

All the logic and management stuff will be passed through this tier. Any operation done by Users and CPOs is being processed through the Business Logic layer. Also only this part of the system will have the availability to interact with external APIs (For example allocating energy from DSOs).

- **Data**

Because of the existence of lots of clients, there will be tons of data. All of this have to be stored in this part of the system.

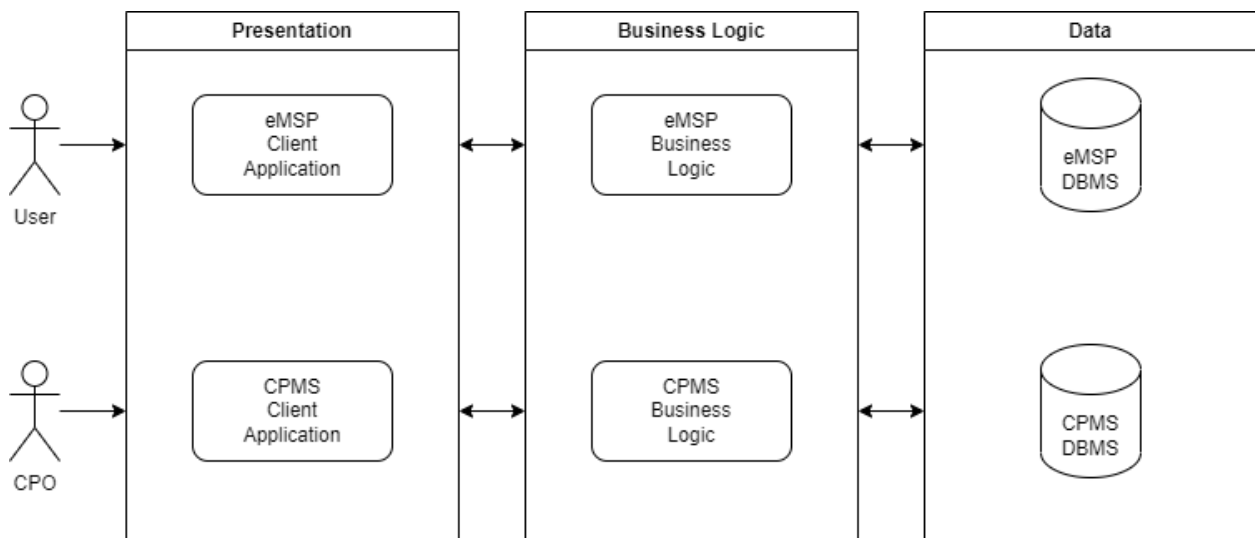


Fig 2.1: The overall view of the system

2.2 Component view

Here is the component diagram, which consists mainly of 3 parts, as in the previous figure:

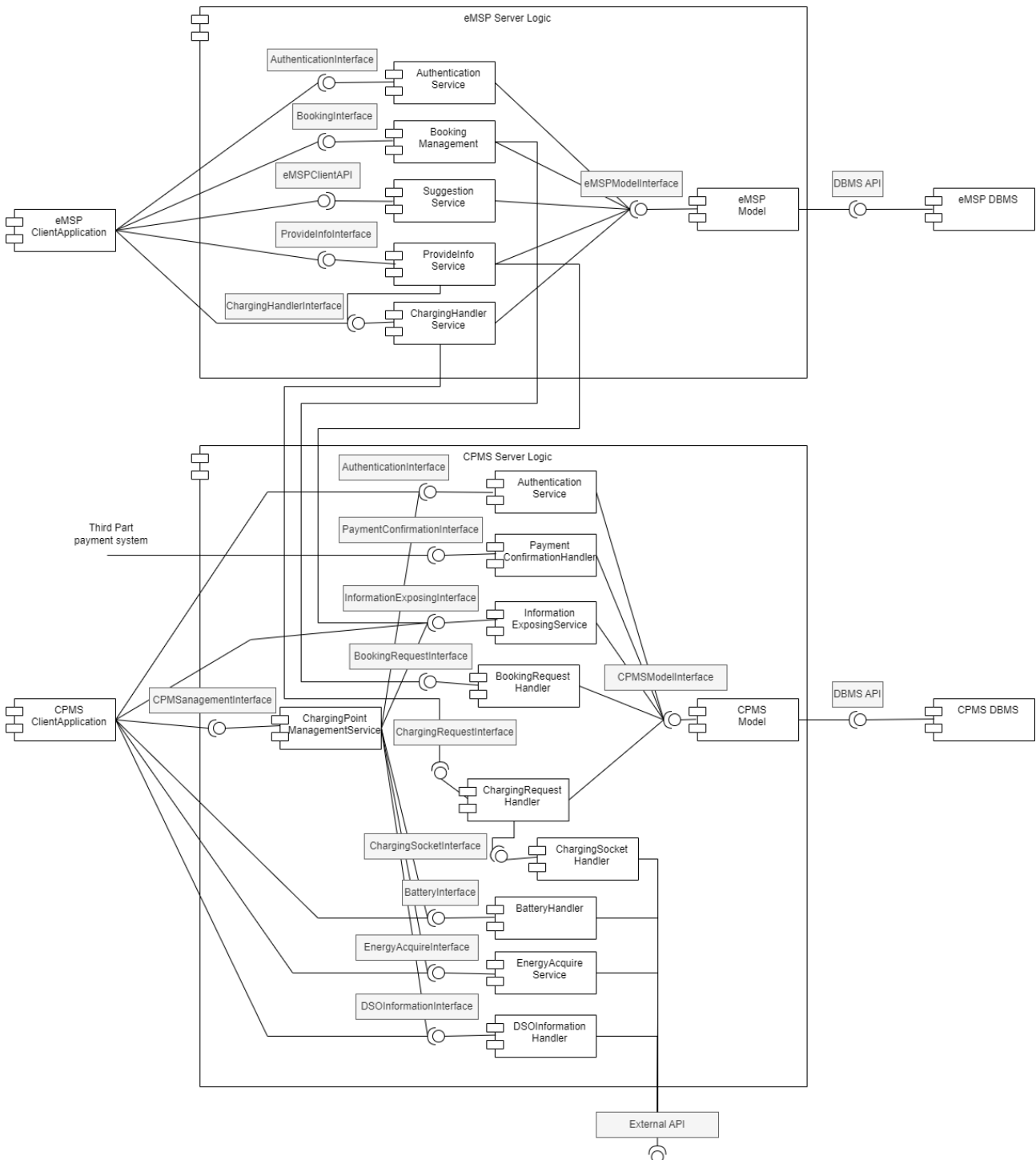


Fig 2.2: Component Diagram

List of system components:

- **eMSPDBMS** - Database Management System. Contains the database for storing persistent data.
- **eMSPClientApplication**
- **eMSPserverLogic**, composed by following components

- **eMSPAuthenticationService** - this component is responsible for logging in and registration functionalities for eMSP clients
- **BookingManagement** - this component is responsible for handling the booking procedure as well for conducting payment procedure
- **SuggestionService** - this component is responsible for generation suggestions and analysis the data it's based on.
- **eMSPModel** - This component contains an object-relation mapping (ORM) for eMSP that other components use when retrieving data from the database. Thus, it is the component solely responsible for communicating with the data tier.
- **ProvideInfoService** - this component is responsible for providing eMSPclient components with the informaton about charging stations, as well info about the driver's car battery level, schedule
- **ChargingHandlerService** - this component is responsible for sending start charging request, handle interrupt scenario, and receive charge completed notification from CPMS
- **CPMSDBMS** - Database Management System. Contains the database for storing persistent data.
- **CPMSClientApplication** - Client application component. E.g web browser session on a mobile phone.
- **CPMSServerLogic**, composed by following components
 - **AuthenticationService** - this component is responsible for logging in and registration functionalities for CPMS clients
 - **PaymentConfirmationHandler** - this component handle with incoming API synchronize request (by third party payment system) that communicate a completed booking has been payed by electric car driver.
 - **CPMSModel** - This component contains an object-relation mapping (ORM) for CPMS that other components use when retrieving data from the database. Thus, it is the component solely responsible for communicating with the data tier.
 - **InformationExposingService** - this component handles the API that expose Charging Points's information (available socket + prices + offer) to eMSP
 - **BookingRequestHandler** - this component handle the incoming booking requests (verify whether requested socket is still available in a CP, and if it does, confirm the booking)
 - **ChargingRequestHandler** - this component handles the incoming charge requests, it interact with ChargingSocketHandler in order to complete a charging process.
 - **ChargingSocketHandler** - this component handles the interaction with charging sockets through sensors/switches described in domain assumptions (verify connection with cars, start/stop charging)
 - **ChargingPointManagementService** - this component processes the request coming from CPMSClientApplication, playing the role of unique component inside the CPMS server that interacts with logged users, in order to isolate client application from other server's components processing inner logic. It administrates also the charging points, basing on user-defined conditions, and control mode, interacts automatically with BatteryHandler, EnergyAcquireService and CPMSModel.
 - **BatteryHandler** - this component handles the storage and usage of energy in batteries
 - **EnergyAcquireService** - this component is responsible for acquiring energy from DSOs and registering purchase made (invoice)
 - **DSOInformationHandler** - This component is responsible for fetching information on energy from DSOs

2.3 Deployment view

This section shows the hardware and tools required to build the system. The diagram is below:

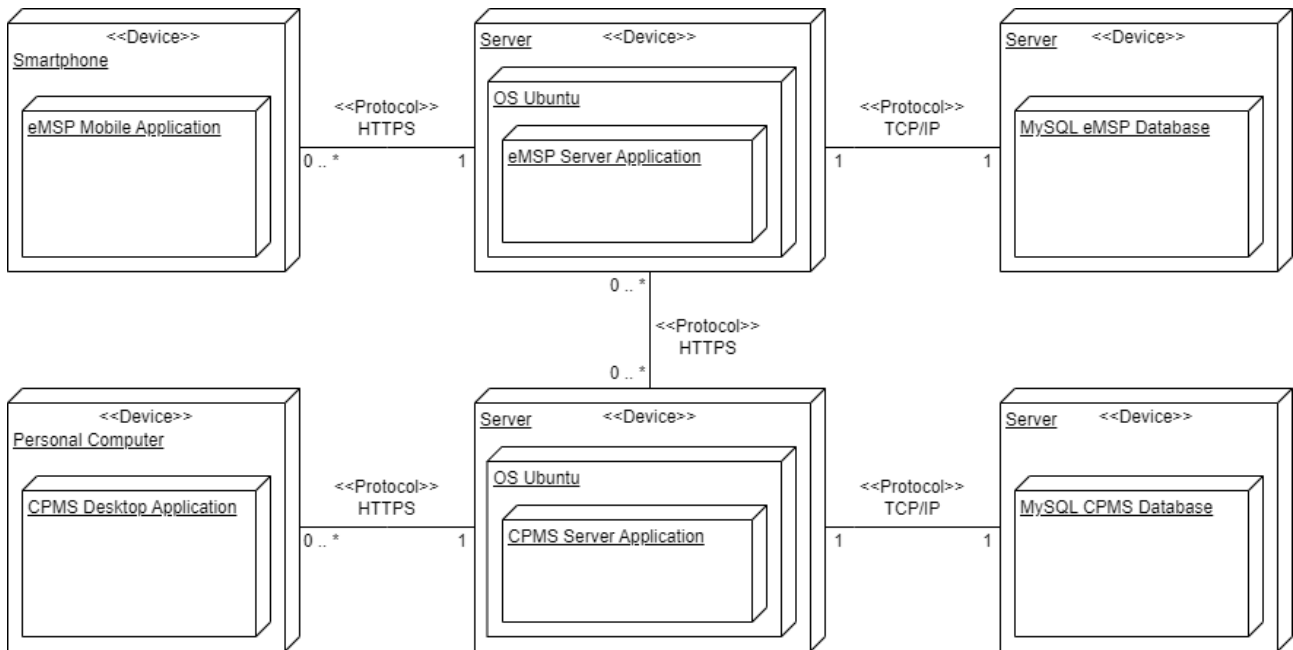


Fig. 2.3 : Deployment Diagram

- **Smartphone** - Simple smartphone with eMSP application installed and access to internet that everyone us using
- **Personal Computer** - Simple computer with special CPMS application installed and internet access
- **Server** - Computer dedicated to store a single application and make it server as a server

2.4 Runtime View

In this section diagrams describing the way in which components of the system interact for the main functionalities are presented.

The interaction of components belonging to different sybsystems is done by means of APIs

2.4.1 Driver Registration

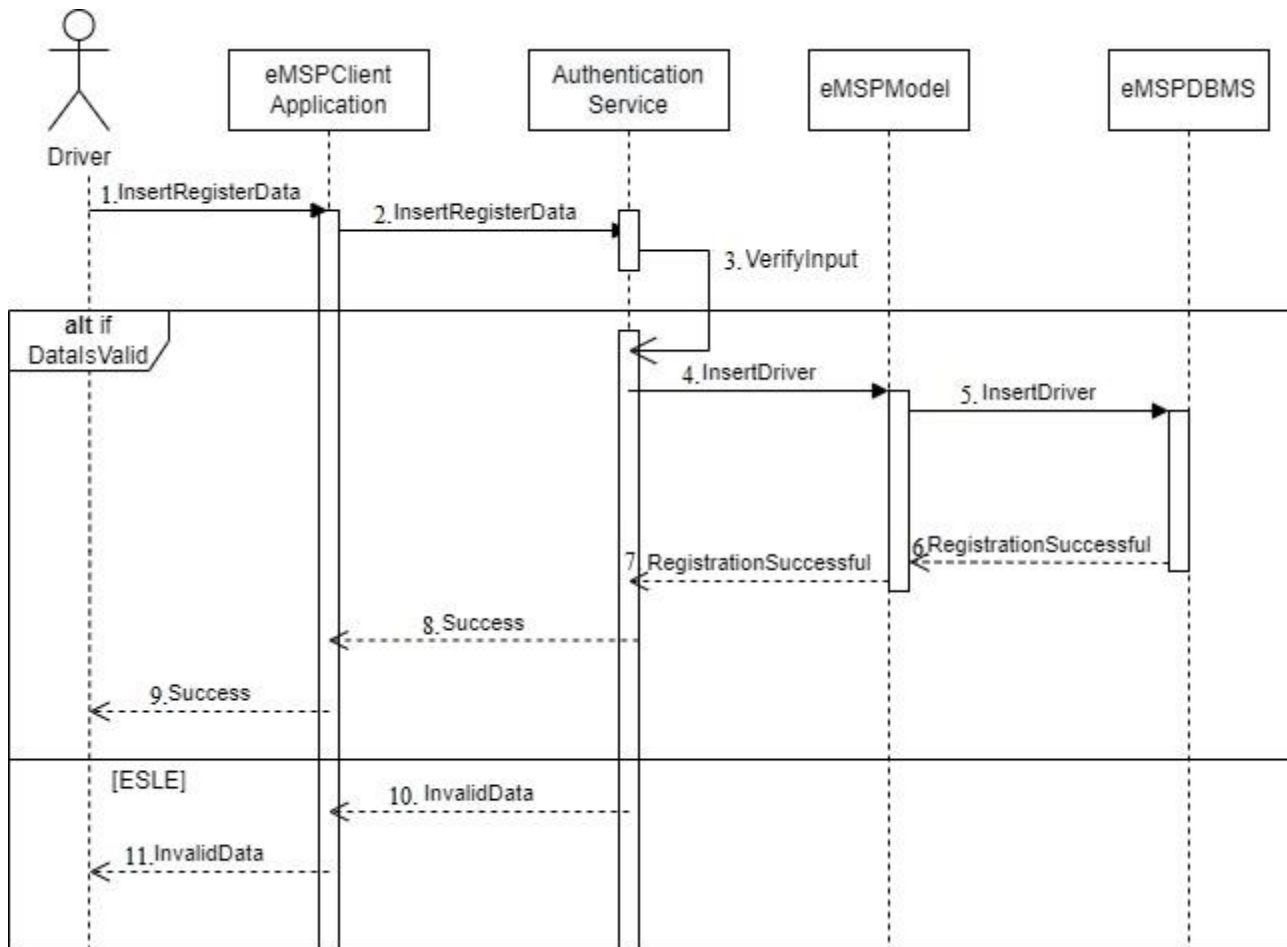


Fig. 2.4 : Driver Registration

2.4.2 General Charging Socket Booking

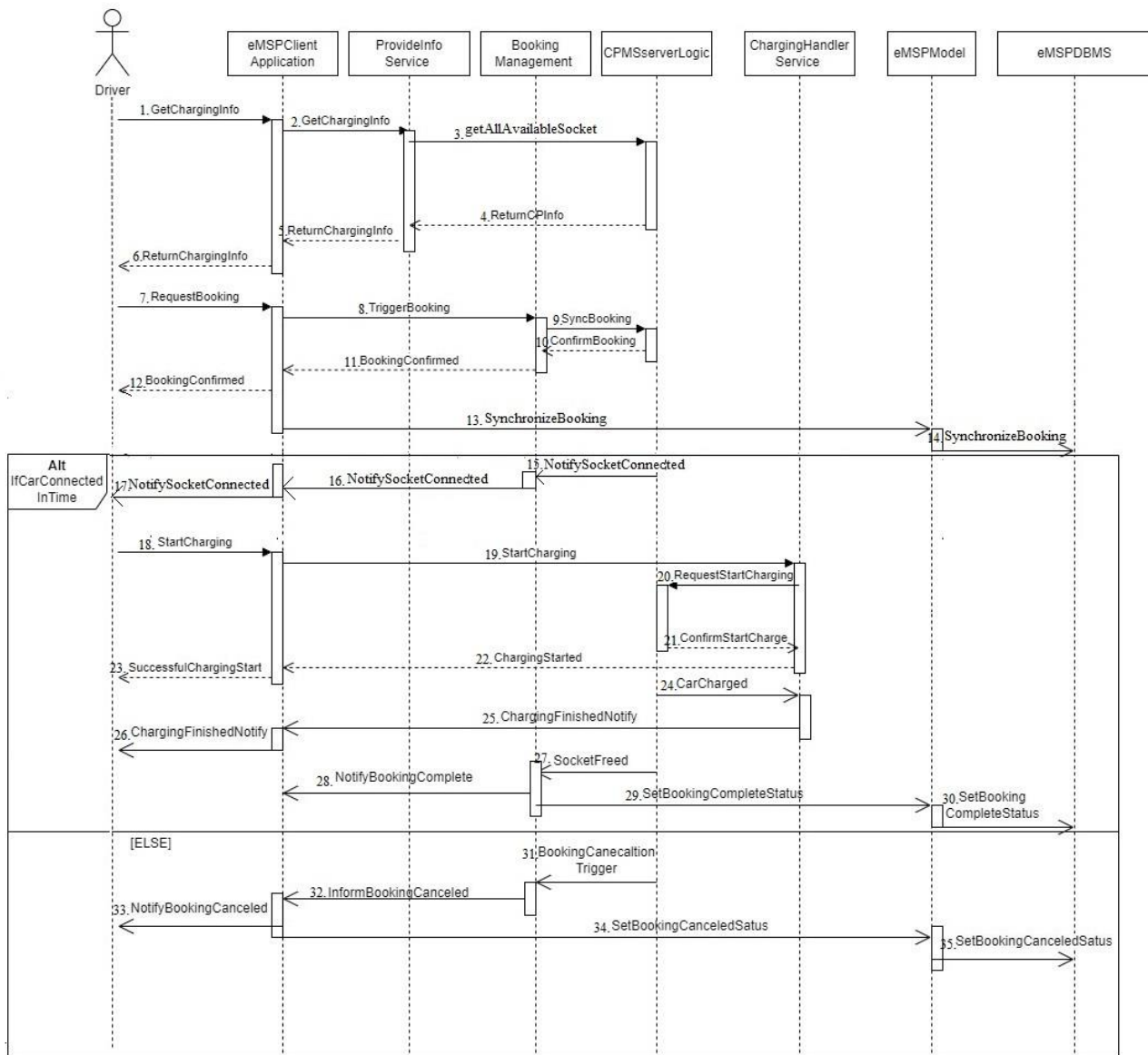


Fig. 2.5 : General Charging Socket Booking

2.4.3 Charging Payment

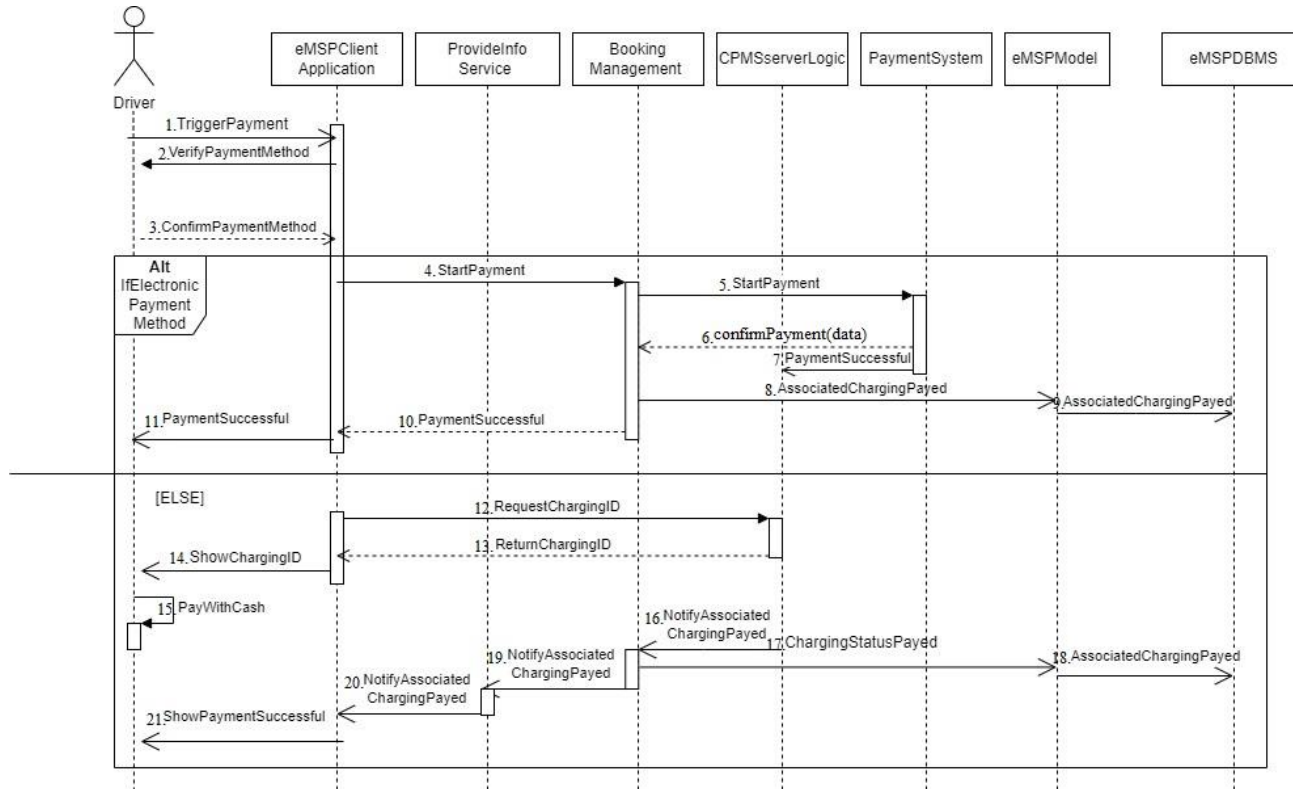


Fig. 2.6 : Charging Payment

2.4.4 Start Charging

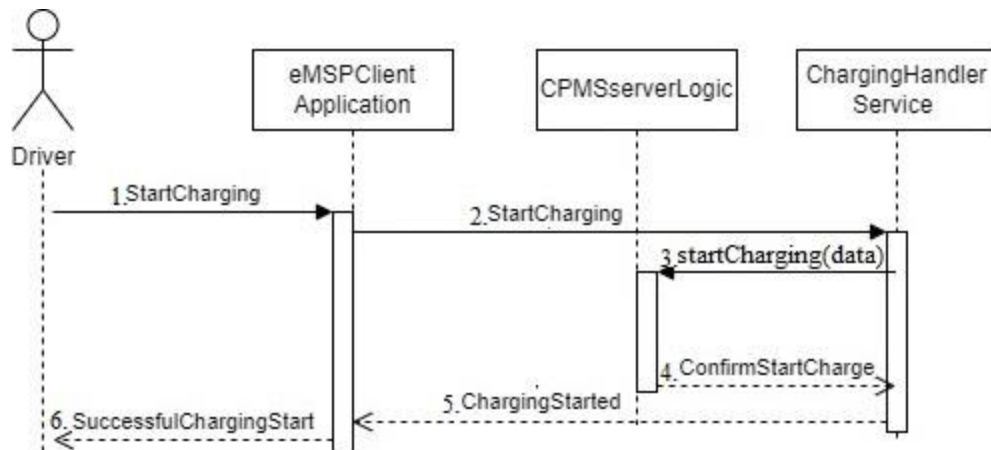


Fig. 2.7 : Start Charging

2.4.5 Accept Suggestion

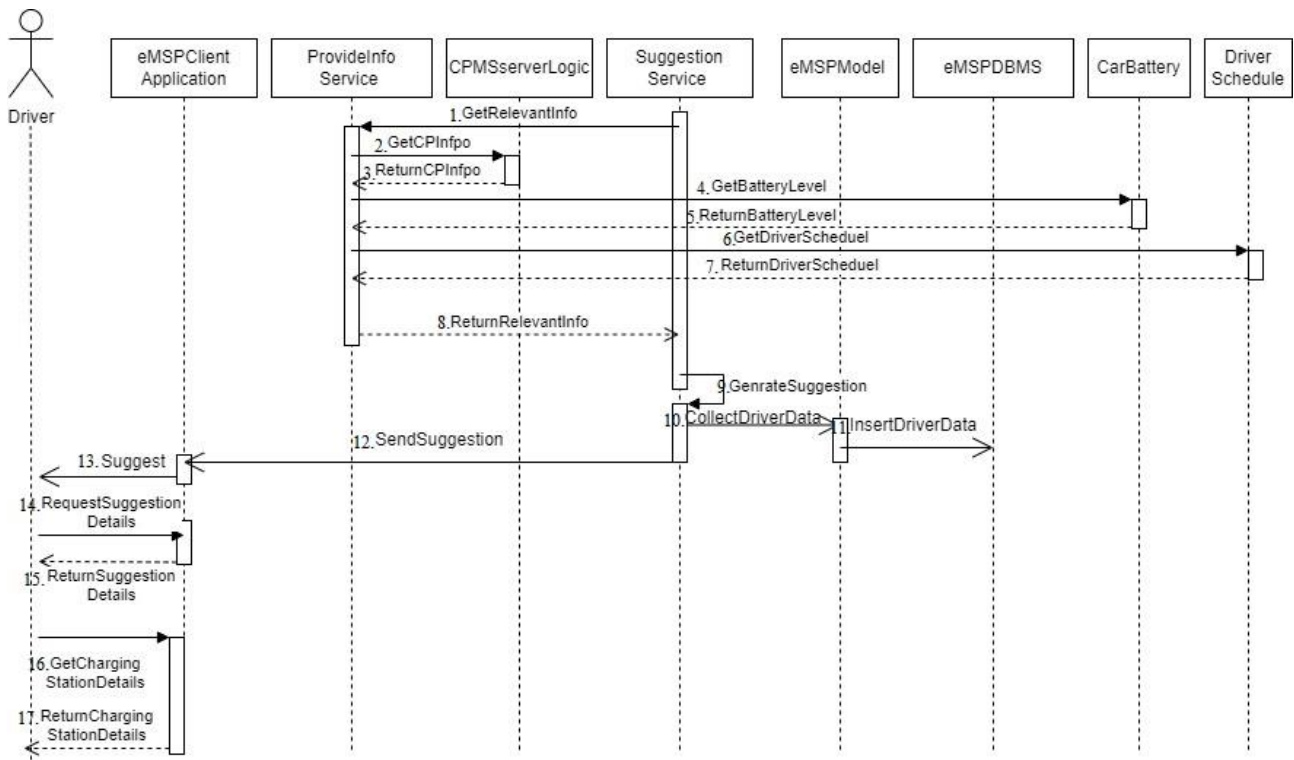


Fig. 2.8 : Accept Suggestion

2.4.6 Login on CPMS (for CPO and CPO Operator)

This diagram shows login procedure of both CPO and CPO Operator profile on CPMS.

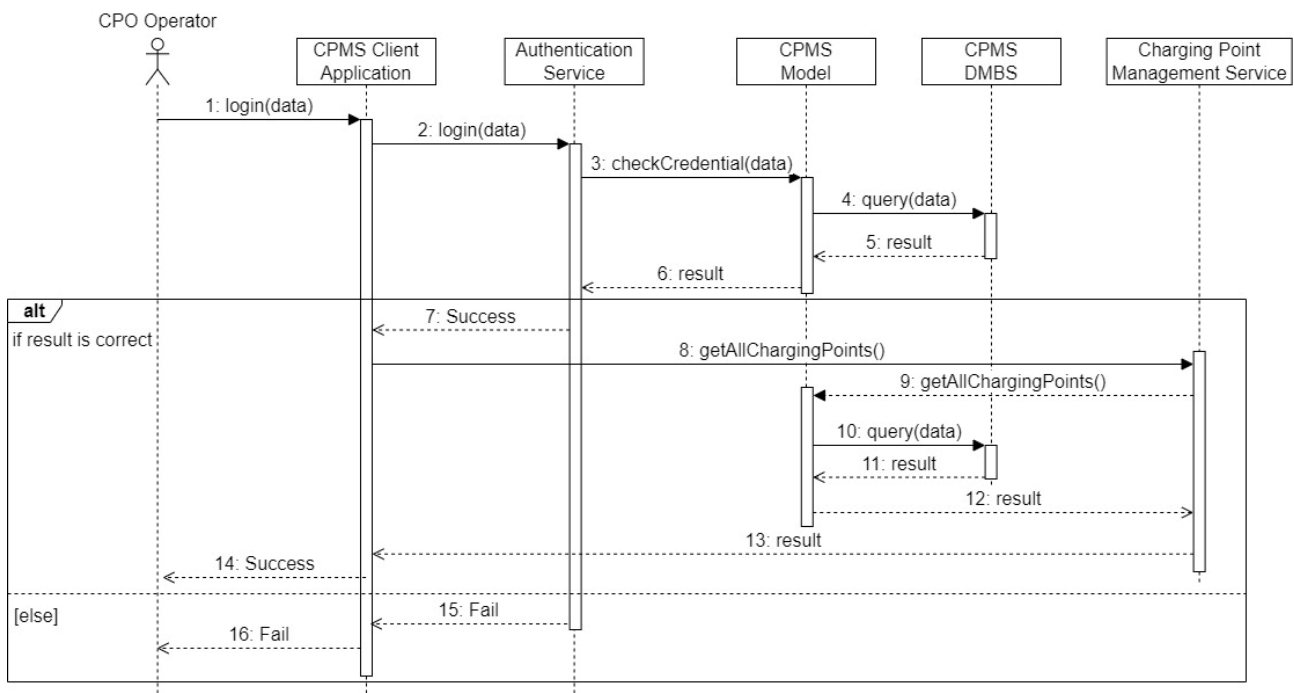


Fig. 2.9 : Login on CPMS

2.4.7 Create operator profile

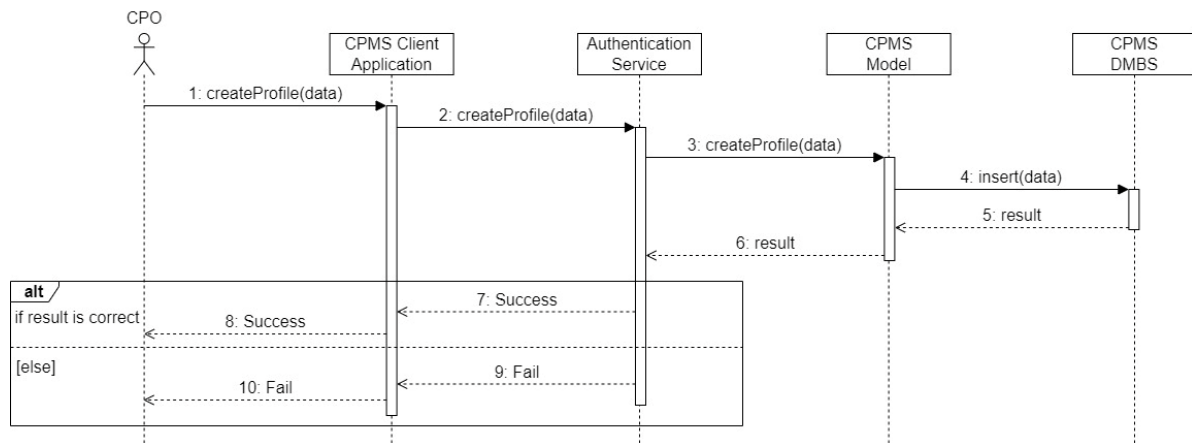


Fig.

2.10 : Create operator profile

2.4.8 Configure Charging Point

This diagram shows in particular, the process of :

- viewing details about a Charging Point
- editing them
- configuring the control mode
- changing source of energy used for charging.

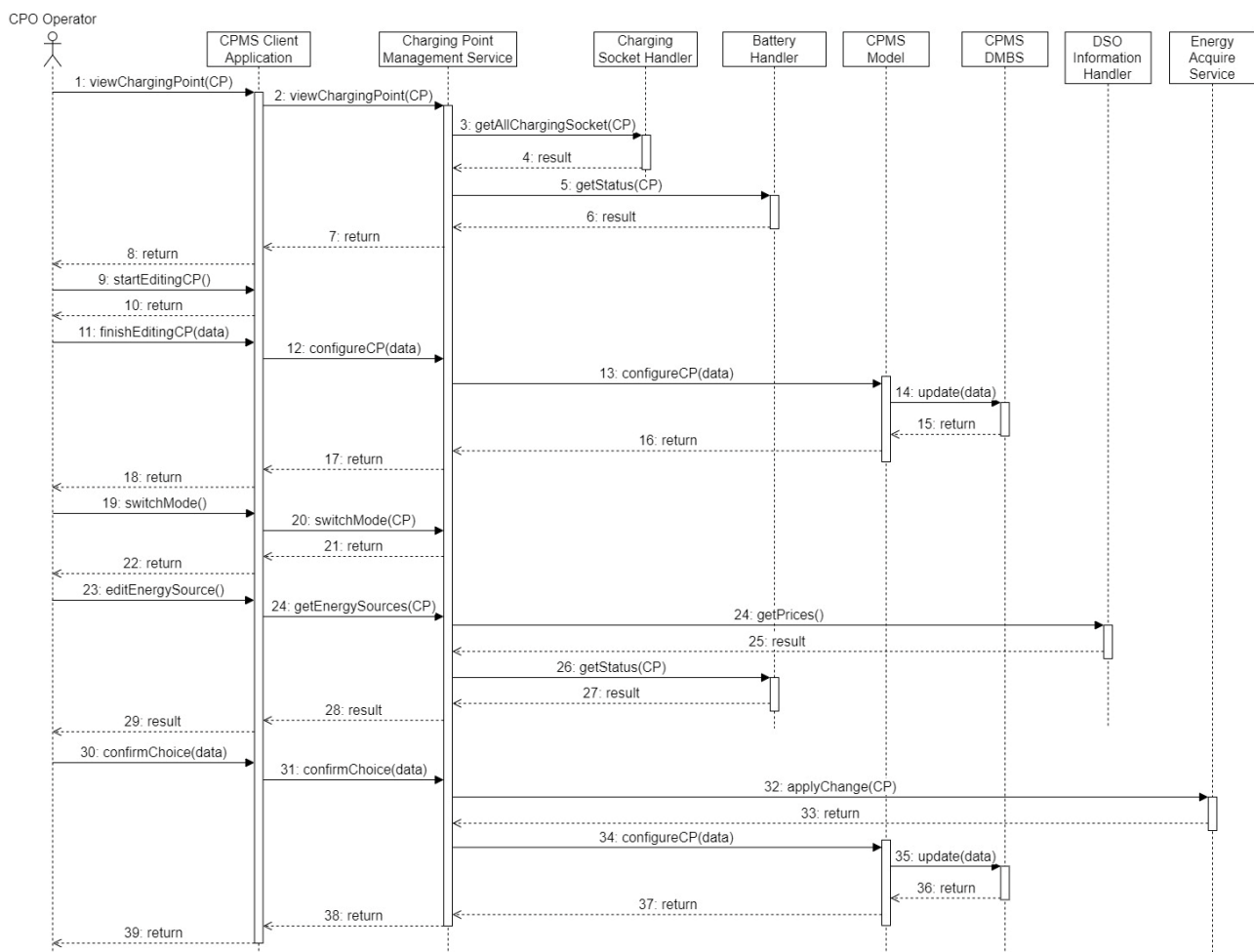


Fig. 2.11 : Configure Charging Point

2.4.9 View and configure Charging Socket's detail

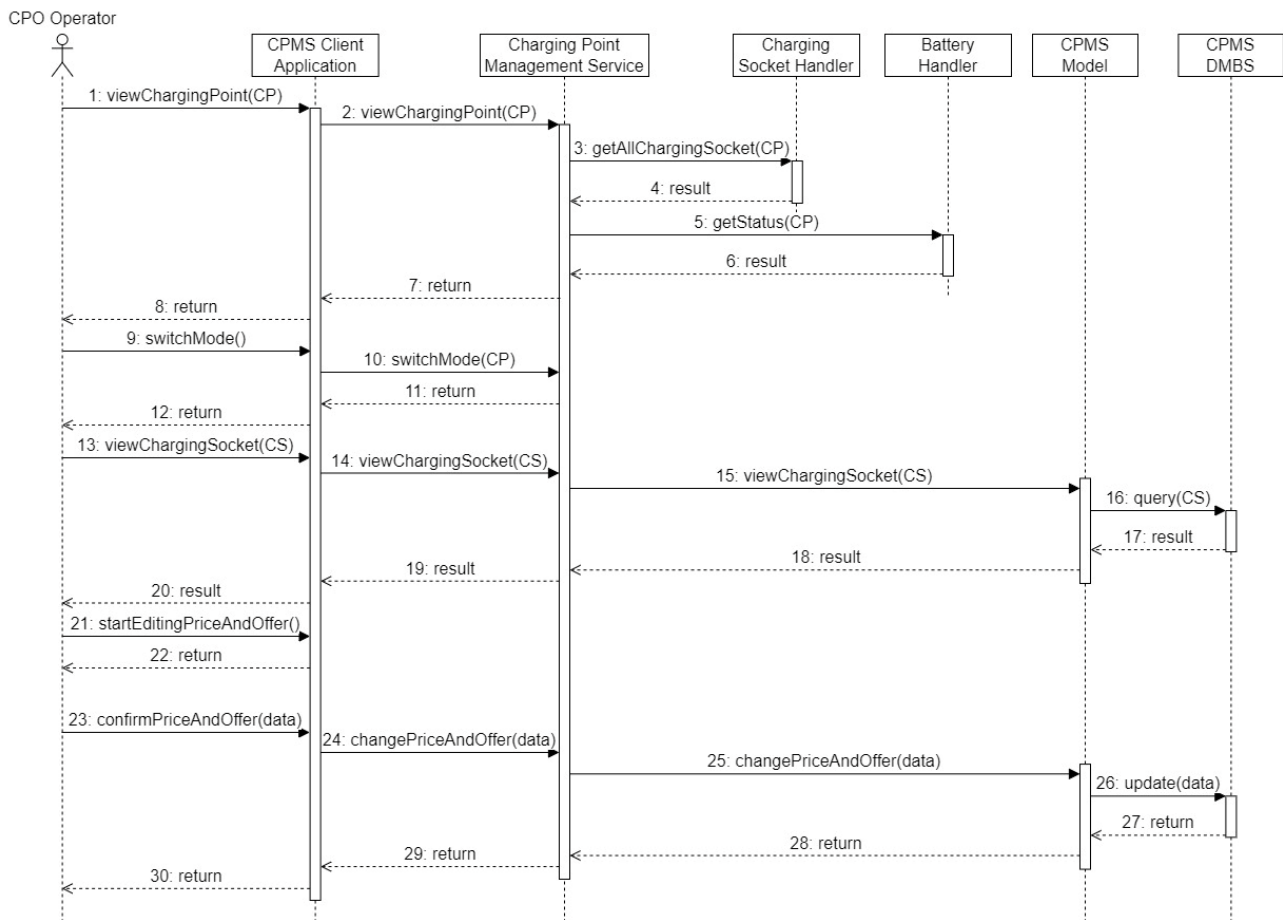


Fig. 2.12 : View and configure Charging Socket's detail

2.4.10 Interaction with eMSP system

this diagram shows a complete interaction between eMSP system with CPMS system focusing on CPMS's component, the one that put focus on eMSP system is shown previously in “2.4.2 General Charging Socket Booking” and “2.4.4 Start Charging”.

A complete interaction starts from request of available charging sockets, request of booking, request of charging and confirm of relative payment. The first 3 request are initiated by Electric car driver through eMSP application directly to CPMS, while the last phase's message comes from third-party payment system.

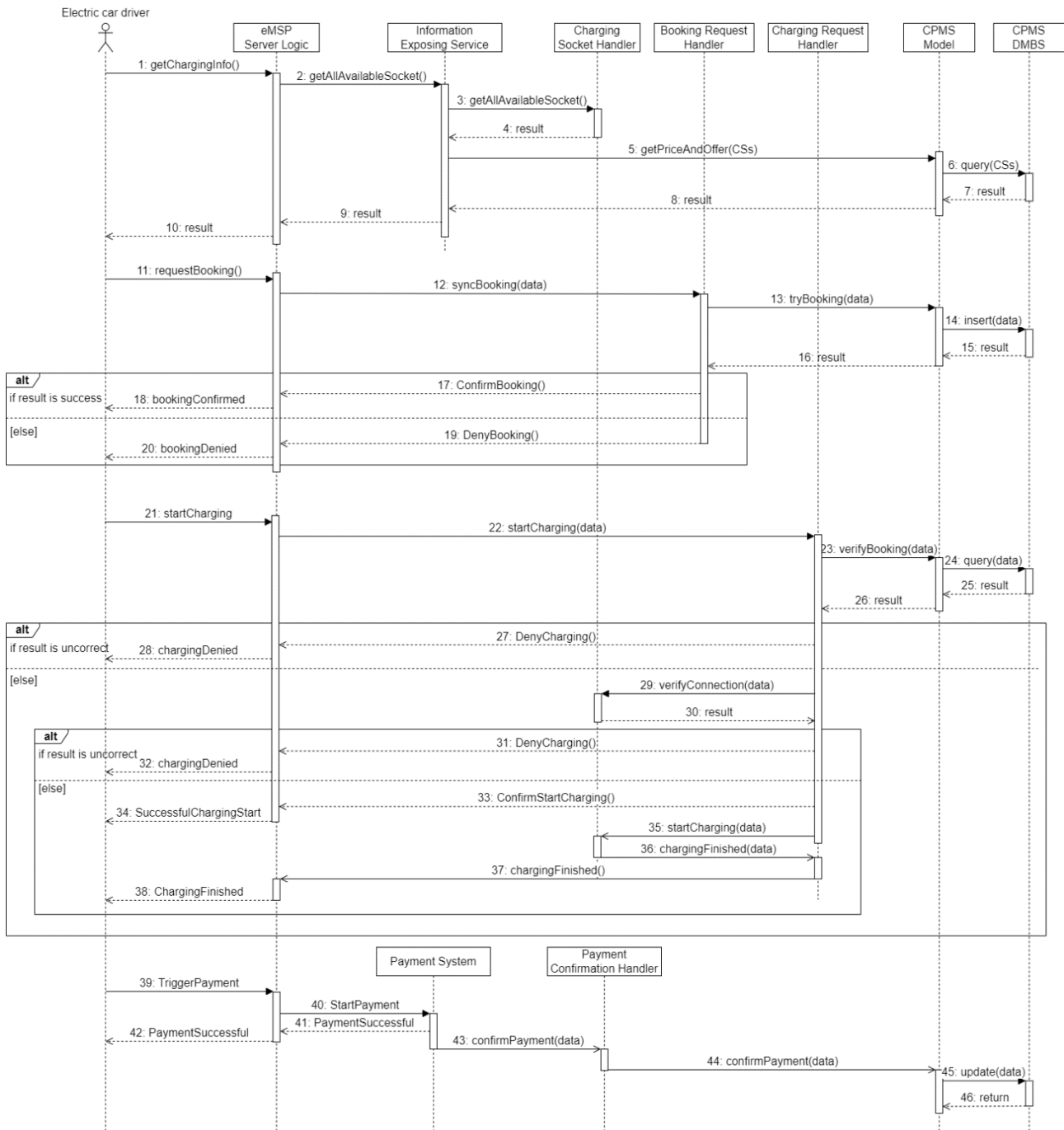


Fig. 2.13 : Interaction with eMSP system

2.5 Component interfaces

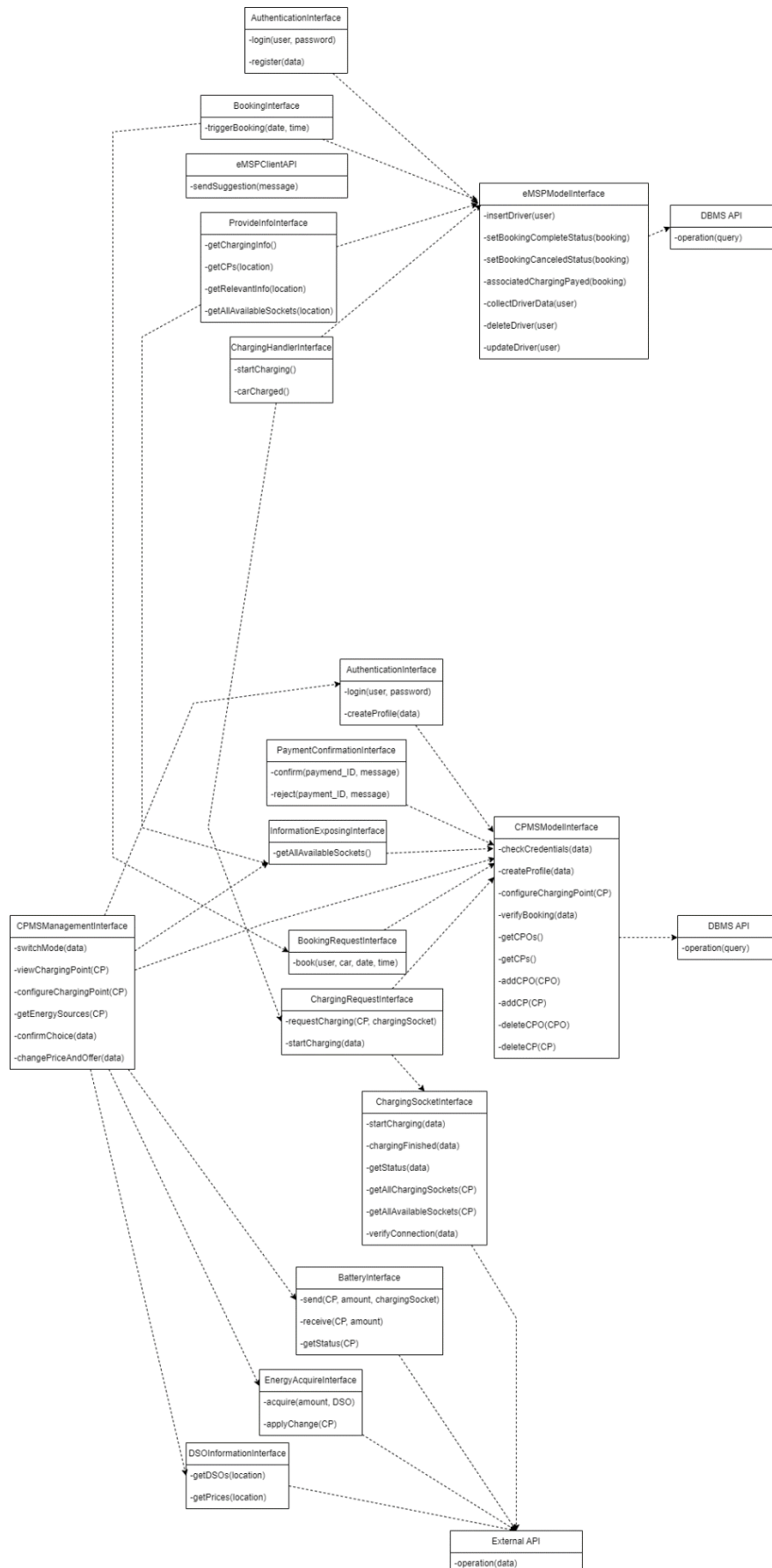


Fig. 2.14 : Component interface diagram

2.6 Selected architectural styles and patterns

Three-Tier architectureⁱ

The 3-tier architectural style was chosen for this project. As previously mentioned, the reason for such choice was to make it easy for the developer to work on it. Also, considering the fact that the system will have thousands of users, it will be impossible to keep all the information and logic in one 'box'. Ordinary users have to deal only with the presentation level, while developers will have to take care of the logic and database parts. To put it simply, presentation is front-end level, business logic is back-end level, and database is simply a database.

3. User interface design

3.1 eMSP User interface design

Car Driver regidtration mockup

eMSP registration

Driver Personal details:

Name

Surname

Birthdate

email adress

password

driving licience number

Debit card details:

debit card number

debit card expiry date

debit card cvv

Car details:

Car model

Associated electronic id token

REGISTER

Fig. 3.1 : Car Driver regidtration mockup

eMSP booking description view mockup

eMSP booking description view

eMSP

Current booking

Booking timeframe: 08:57 10:57

⚡ Charging socket type: Slow

Charging socket id: 1qwert34

Price: 7\$/kW

Charging status: accepted

PAY

START CHARGING

CANCEL BOOKING

Fig. 3.2 : eMSP booking description view mockup

eMSP charging stations map mockup

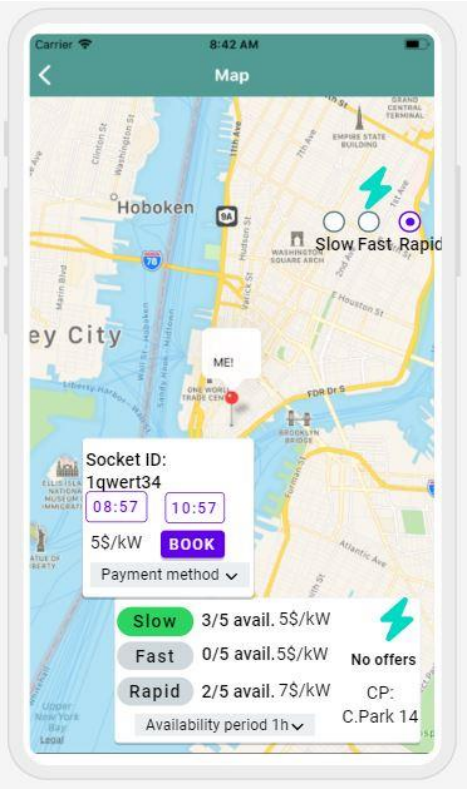


Fig. 3.3 : eMSP charging stations map mockup

eMSP initial view mockup

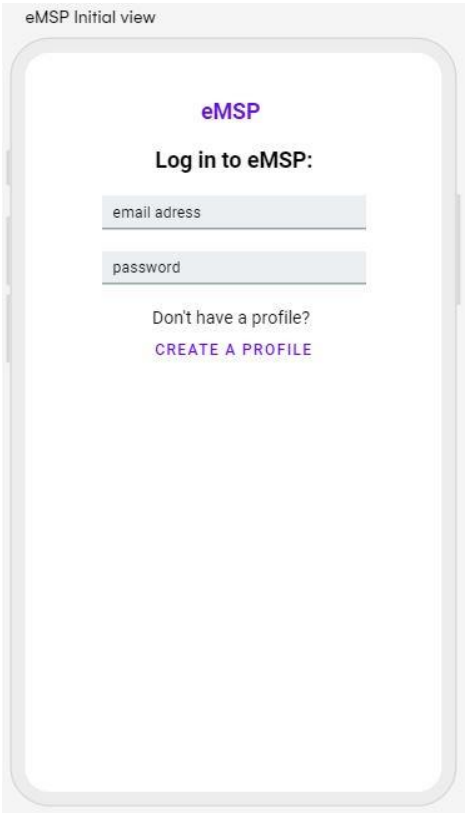


Fig. 3.4 : eMSP initial view mockup

eMSP main menu mockup

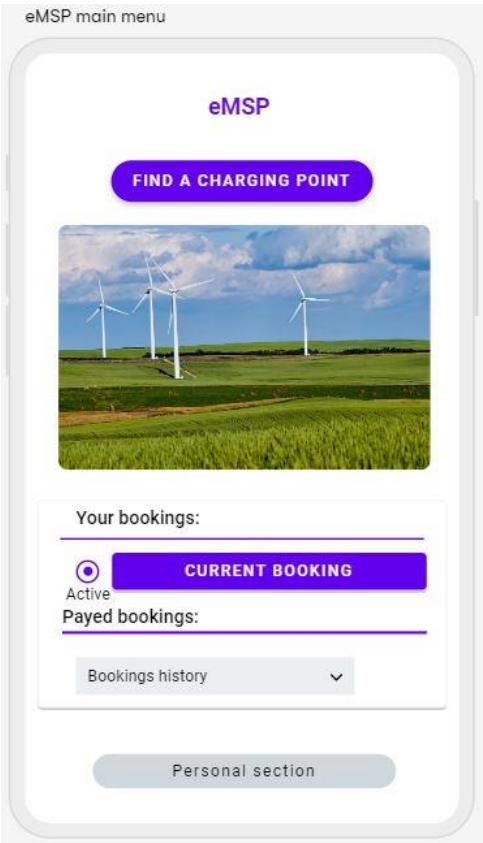


Fig. 3.5 : eMSP main menu mockup

eMSP section describing the suggestion mockup

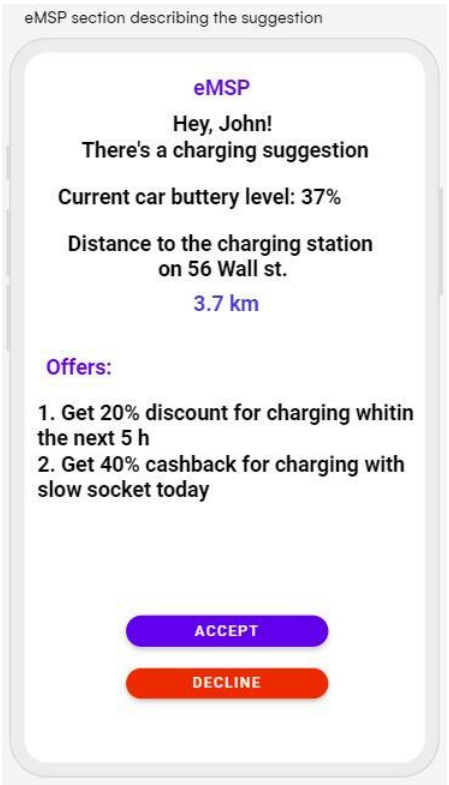


Fig. 3.6 : eMSP section describing the suggestion mockup

3.2 CPMS User Interface Mockups

Login to CPMS

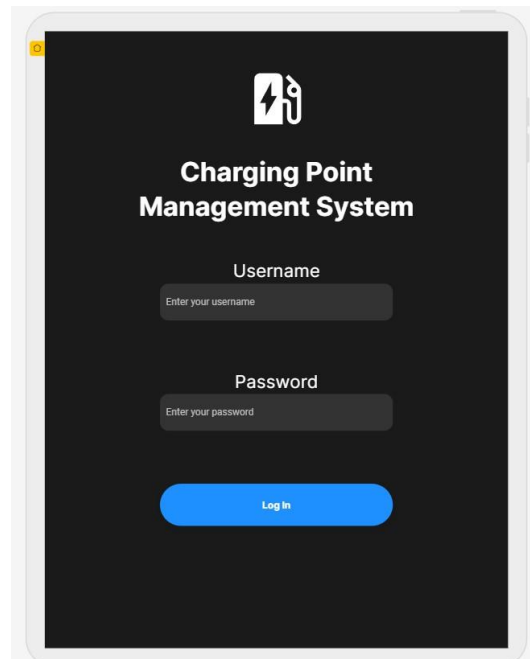


Fig. 3.7 : Login to CPMS

Initial view for CPO

After log-in, user see the below view containing all charging points under their administration. In particular, the CPO Operator do not have access to the blues button, which allow only CPO to “Create new Charging Point” and to “Create new Operator profile”

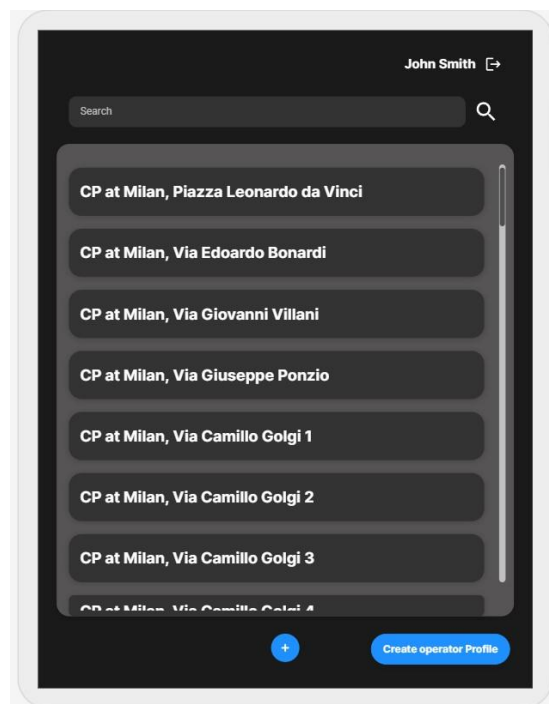
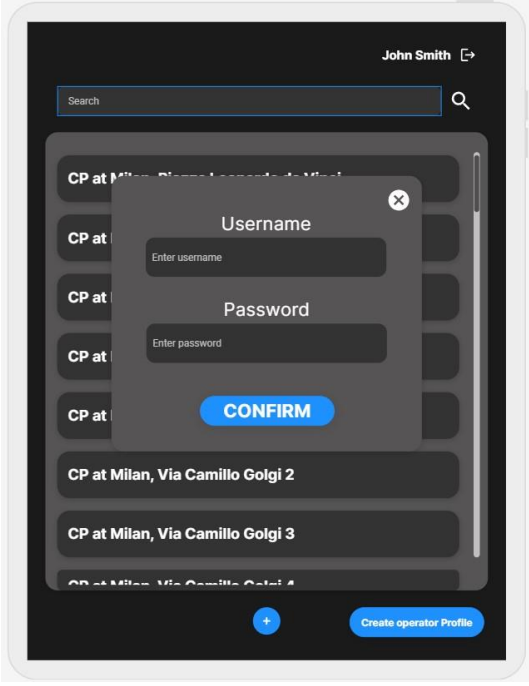


Fig. 3.8 : Initial view for CPO

CPO operator profile creation window

The “Create new Operator profile” mentioned previously makes CPMS prompt below form where CPO could configure credential for a new Operator profile.



The screenshot shows a mobile application interface for creating a new operator profile. At the top, the user's name "John Smith" is displayed with an edit icon. Below this is a search bar. A modal form is open in the center, titled "CP at Milan, Via Camillo Golgi 2" (partially visible). The form contains two input fields: "Username" with a placeholder "Enter username" and "Password" with a placeholder "Enter password". A blue "CONFIRM" button is at the bottom of the modal. Below the modal, a list of locations is visible, including "CP at Milan, Via Camillo Golgi 2" and "CP at Milan, Via Camillo Golgi 3". At the bottom of the screen, there is a blue "+" button and a "Create operator Profile" button.

Fig. 3.9 : CPO operator profile creation window

Charging Point Detail view & editing view

By clicking on one of charging point from list at initial view, below view will be shown, containing detailed information about charging point, list of charging sockets present in it, a switch for energy source control mode, and 2 buttons. In particular, the CPO Operator have access only to switch, which will make “Energy source” field clickable, leading user to a view that allows them to select the desired sources of energy to be used. The blues button are accessible only to CPO, allowing them to “Create new Charging Socket” and to “Configure Charging Point Detail”.

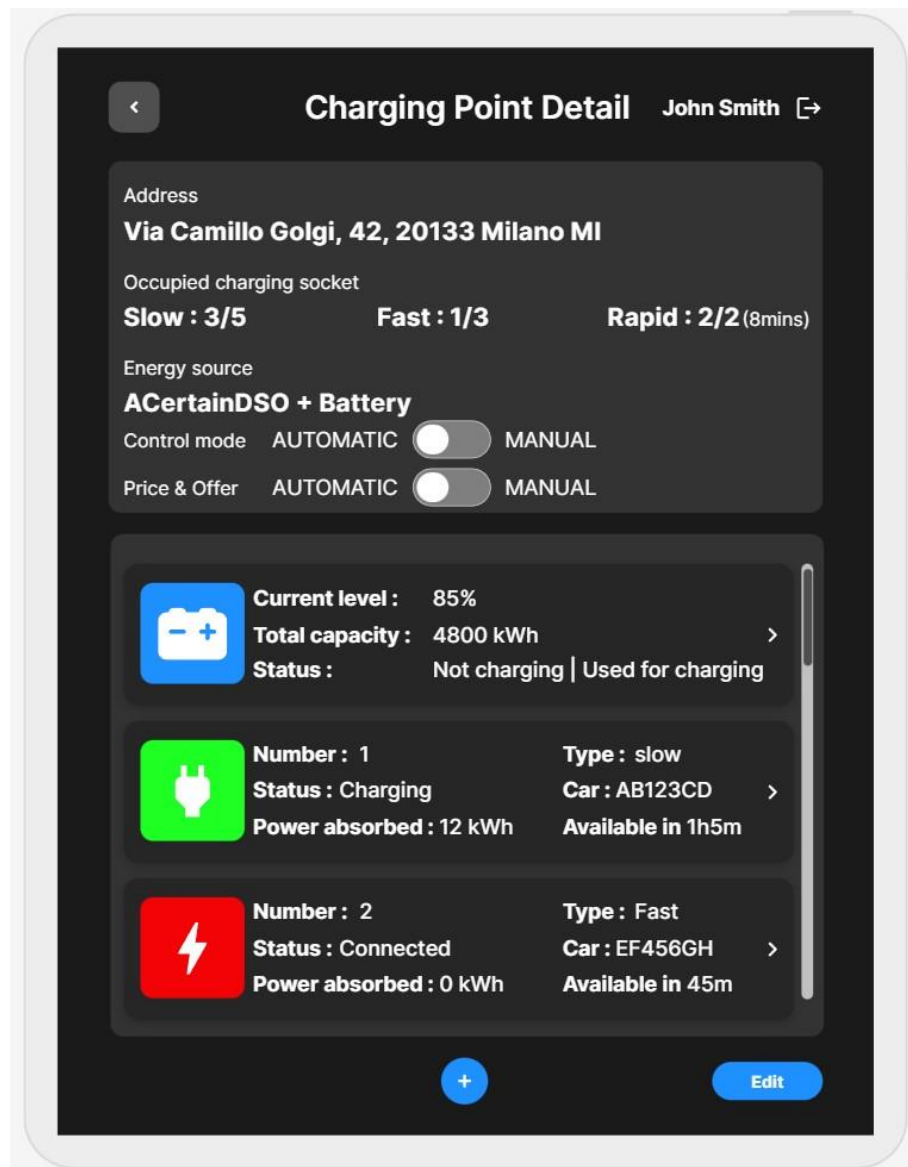


Fig. 3.10 : Charging Point Detail view

At Editing view, several fields will become editable, and 4 buttons will be displayed, which will prompt a window allowing CPO to modify the corresponding conditions.

Energy Source Selection view

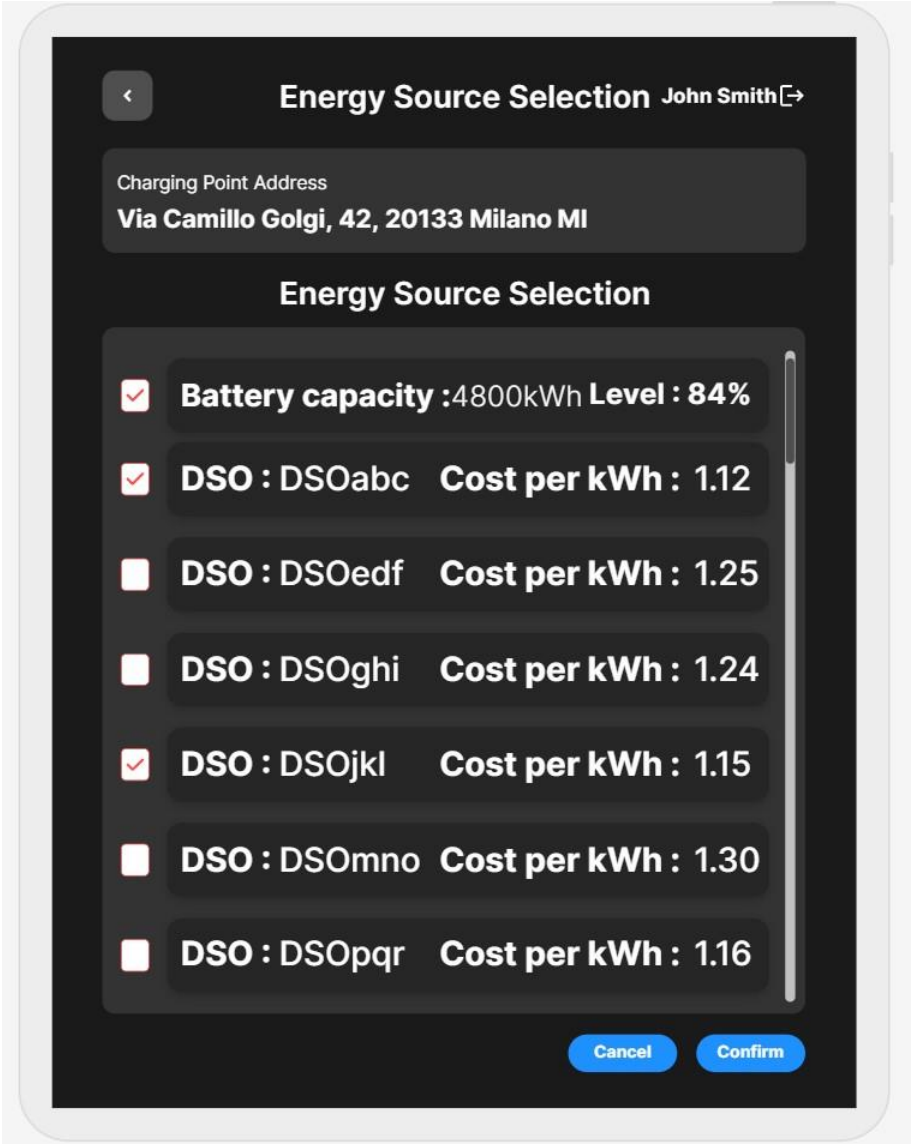


Fig. 3.12 : Energy Source Selection view

Charging Socket's booking history view & Price and Offer configuration window

By clicking on one of charging sockets from list at Charging Point Detail view, below view will be shown, containing history of bookings associated with such charging socket, a blue button accessible only by CPO used to associate socket with its sensors (w.r.t. Domain Assumption 17 : “The status of charging socket in a CP is monitored by sensors and transferred to CPMS”), and a red button accessible by all user to configure the price and offer related to this socket manually.

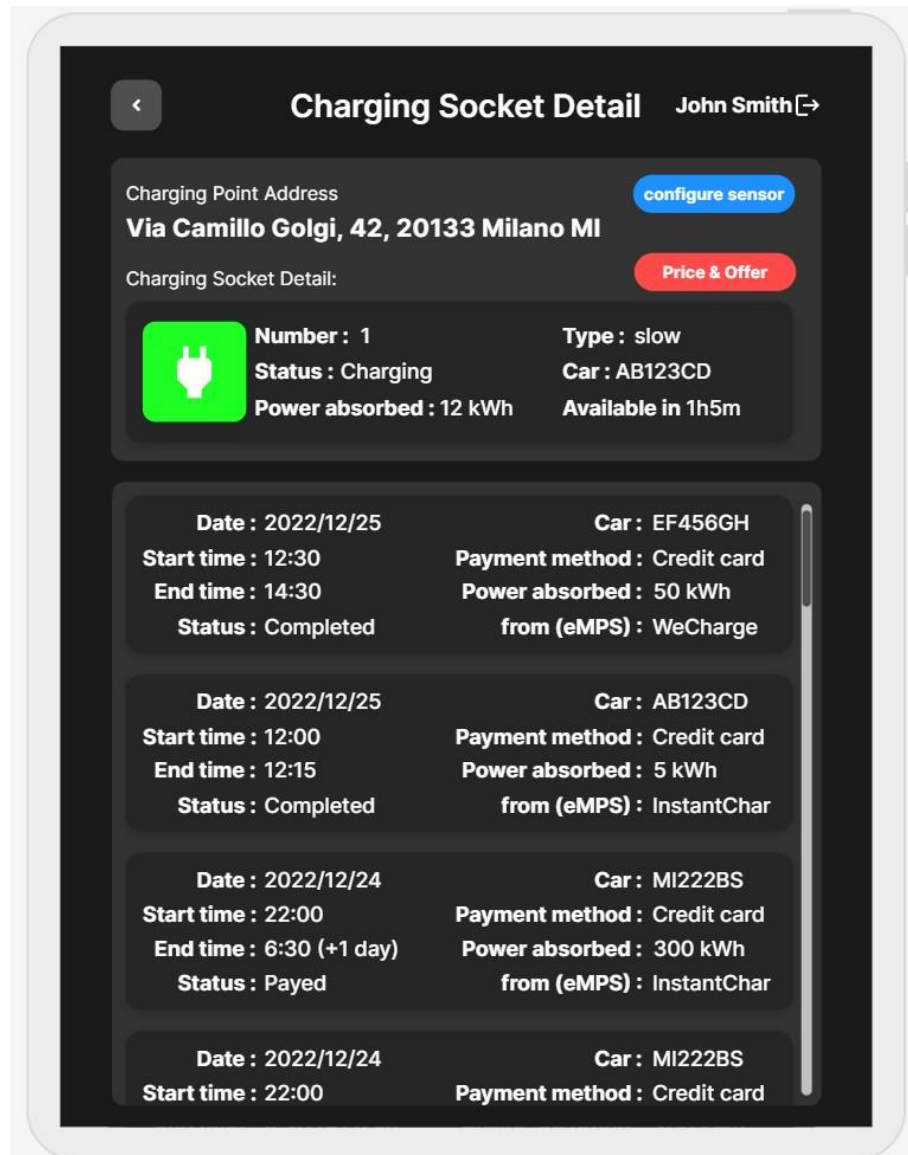


Fig. 3.13 : Charging Socket's booking history view

4. Requirements traceability

This section contains a table explaining what components are required in order to fulfil each of the requirements specified in the RASD. To save some space, the components have been given abbreviations as shown in list below.

- **eDB** - eMSPDBMS
- **eCA** - eMSPClientApplication
- **eAS** - eMSPAuthenticationService
- **eBM** - eMSP Booking Management
- **eSS** - eMSP SuggestionService ****
- **eM** - eMSPModel
- **ePIS** - ProvideInfoService ****
- **eCHS** - ChargingHandlerService
- **cpDB** - CPMSDBMS
- **cpCA** - CPMSClientApplication
- **cpAS** - AuthenticationService
- **cpPCH** - PaymentConfirmationHandler
- **cpMODEL** - CPMSModel
- **cpIES** - InformationExposingService
- **cpBRH** - BookingRequestHandler
- **cpCRH** - ChargingRequestHandler
- **cpCSH** - ChargingSocketHandler
- **cpCPMS** - ChargingPointManagementService
- **cpBH** - BatteryHandler
- **cpEAS** - EnergyAcquireService
- **cpDIH** - DSOInformationHandler

The requirements are given below:

Functional Requirements

Requirement	Description
R1(eMSP)	The eMSP subsystem shall allow the User to see the estimated time before a charging socket of a certain type at a certain CP is freed
R2(eMSP)	The eMSP subsystem shall allow the user to book a charging socket of a certain type at a certain charging station for a certain timeframe maximum 15 min before the charging should start
R3(eMSP)	The eMSP subsystem shall allow the user to choose a payment method (i.e. cash, card, paypal) during the booking procedure
R4(eMSP)	The eMSP subsystem shall allow the user to trigger the start of a charging process when the car is connected to a charging socket
R5(eMSP)	The eMSP subsystem shall not allow the user to trigger the start of a charging process when the car is not connected to a charging socket
R6(eMSP)	The eMSP subsystem shall allow user to know the locations of the charging stations within the range of 15 min reachability according to the car navigation system
R7(eMSP)	The eMSP subsystem shall allow user to know about the costs and availability of charging at charging stations within the range of 15 min reachability according to the car navigation system
R8(eMSP)	The eMSP subsystem shall allow user to know about the special offers of charging at charging stations within the range of 15 min reachability according to the car navigation system
R9(eMSP)	The eMSP subsystem shall notify user when the charging is finished

Requirement	Description
R10(eMSP)	The eMSP subsystem shall have the payment interfaces for user so that he can manage payment method details, receive receipts of payments
R11(eMSP)	The eMSP subsystem shall analyse the battery charge level, occupations of the user according to the calendar, special offers by the CPOs and availability of charging sockets in order to make a suggestion to the user of getting a charge at a certain charging station
R12(eMSP)	The eMSP subsystem shall allow user to create an account and use its functionalities
R1(CPMS)	The system must allow registered Operator and CPO to login
R2(CPMS)	The system must allow CPO company to create Operator profile for its employee
R3(CPMS)	The system must allow CPO profile to create Charging Point
R4(CPMS)	The system must allow CPO profile to update a Charging Point's information
R5(CPMS)	The system must allow CPO profile to create Charging socket
R6(CPMS)	The system must allow CPO profile to update a Charging socket's information
R7(CPMS)	The system shall allow CPO profile to configure the conditions according to which CPMS will automatically change the source of energy used for charging in a CP between DSOs and batteries if present
R8(CPMS)	The system shall allow CPO profile to configure the conditions according to which the batteries in a CP will get charged, if present
R9(CPMS)	The CPMS shall display the information on available batteries present in a Charging Point to its user
R10(CPMS)	The CPMS shall display the information about number of vehicles being charged in a CP, amount of power absorbed, and time left to the end of the charge for each vehicle to its user
R11(CPMS)	The CPMS shall display the information about booking relative to a specific charging socket to its user
R12(CPMS)	The CPMS should get the current price of energy from available DSOs
R13(CPMS)	The system shall allow Operator profile to change the Energy source controlling Mode of charging point
R14(CPMS)	The CPMS shall display the information provided by DSOs to its user
R15(CPMS)	The system shall allow Operator profile to set the energy providers from where acquire or extract energy
R16(CPMS)	The system shall allow Operator profile to configure whether store energy in a charging point's batteries
R17(CPMS)	The system shall allow external eMSP to book a charging socket for a time period.
R18(CPMS)	The system shall allow external eMSP to start and stop a charging process on condition satisfied

non-Functional Requirements

Requirement	Description
R17(eMSP)	The system should penalize the driver if the car isn't disconnected from the charging socket at most 15 mins after the booking end time

Constraints

Requirement	Description
R13(eMSP)	The eMSP subsystem must be informed about moment when the car is connected to a charging socket by the CPMS subsystem through an API
R14(eMSP)	The eMSP subsystem must be informed about moment when the car is disconnected from a charging socket by the CPMS subsystem through an API
R15(eMSP)	The eMSP subsystem must be informed about moment when the charging of the corresponding car is finished through an API

R16(eMSP)	The eMSP should be able to interact with multiple CPMS
-----------	--

4.1 Mapping of components on requirements:

Requirement	Components	Description
R1(eMSP)	eCA, eAS, ePIS	To see the info about charging socket driver needs to use mobile application, be logged in, the eMSP needs to request info from CPMSs
R2(eMSP)	eCA, eAS, ePIS, eBM	To see the info about charging socket driver needs to use mobile application, be logged in, the eMSP needs to request info from CPMSs and request booking
R3(eMSP)	eSA, eAS, ePIS, eBM	To pick the payment method for charging driver needs to use mobile application, be logged in, the eMSP needs to verify that the charging station supports the payment method from CPMSs and request booking
R4(eMSP)	eCA, eAS, eBM, eCHS	To start the charging of the certain booking driver needs to use mobile application, be logged in, the eMSP needs to synchronize the charging start of the corresponding booking and request start to CPMS
R5(eMSP)	eCA, eAS, eBM, eCHS	To start the charging of the certain booking driver needs to use mobile application, be logged in, the eMSP needs to synchronize the charging start of the corresponding booking and request start to CPMS. CPMS sends rejection to eCHS if the socket is disconnected
R6(eMSP)	eCA, eAS, ePIS	To see the info about charging stations driver needs to use mobile application, be logged in, the eMSP needs to request info from CPMSs and access the car navigation system
R7(eMSP)	eCA, eAS, ePIS	To see the info about charging stations driver needs to use mobile application, be logged in, the eMSP needs to request info from CPMSs and access the car navigation system
R8(eMSP)	eCA, eAS, ePIS	To see the info about charging stations driver needs to use mobile application, be logged in, the eMSP needs to request info from CPMSs and access the car navigation system
R9(eMSP)	eCA, eAS, eBM, eCHS	To be notified the charging of the certain booking is finished driver needs to use mobile application, be logged in, the eMSP needs to receive charging finish notification from CPMS and synchronize the charging status of the corresponding booking and synchronize it with the DB.
R10(eMSP)	eCA, eAS	To pick the payment method for charging driver needs to use mobile application, be logged in
R11(eMSP)	eCA, eAS, ePIS, eSS, eM, eDB	To receive suggestions driver needs to use mobile application, be logged in, the eMSP needs to request info from CPMSs, have access to the car navigation system and user schedule, use recommendation algorithms and collect driver data to the DB
R12(eMSP)	eCA, eAS, eM, eDB	To register a new profile, driver needs to use mobile application and undergo the procedure of registration. eMSP needs to put new user to its database
R13(eMSP)	eBM	The socket is associated to the booking
R14(eMSP)	eBM	The socket is associated to the booking
R15(eMSP)	eCHS	This component is responsible for receiveing notifications about the charging finish

Requirement	Components	Description
R16(eMSP)	eCHS, eCA, ePIS, eBM	All these components must be able to interact with multiple CPMSs
R17(eMSP)	eBM, eM, eDB	The booking status depends on the socket connection to the car. Such penalization must be collected in the DB
R1(CPMS)	cpCA, cpAS, cpMODEL, cpDB	Operator and CPO go through ClientApplication to access CPMS Authentication Service in order to login to CPMS, having credentials checked with data stored in DB.
R2(CPMS)	cpCA, cpAS, cpMODEL, cpDB	CPO go through ClientApplication to access CPMS Authentication Service in order to create new operator profile, storing credentials in DB.
R3(CPMS)	cpCA, cpCPMS, cpMODEL, cpDB	CPO go through ClientApplication to access CPMS ChargingPointManagementService in order to create new charging point, storing data in DB.
R4(CPMS)	cpCA, cpCPMS, cpMODEL, cpDB	CPO go through ClientApplication to access CPMS ChargingPointManagementService in order to modify a charging point's information, storing them in DB.
R5(CPMS)	cpCA, cpCPMS, cpCSH	CPO go through ClientApplication to access CPMS ChargingPointManagementService, and ChargingSocketHandler in order to create new charging socket, associate it to a charging point and sensors.
R6(CPMS)	cpCA, cpCPMS, cpCSH	CPO go through ClientApplication to access CPMS ChargingPointManagementService, and ChargingSocketHandler in order to configure a charging socket, associated to a charging point and sensors.
R7(CPMS)	cpCA, cpCPMS, cpBH, cpDIH, cpEAS, cpMODEL, cpDB	CPO go through ClientApplication to access CPMS ChargingPointManagementService, where he could set the condition according to which a charging point decide whether extract energy for charging from DSOs or Batteries.
R8(CPMS)	cpCA, cpCPMS, cpBH, cpDIH, cpEAS, cpMODEL, cpDB	CPO go through ClientApplication to access CPMS ChargingPointManagementService, where he could set the condition according to which a charging point decide when and from which DSO acquire energy storing them to batteries.
R9(CPMS)	cpCA, cpCPMS, cpBH	CPO Operator go through ClientApplication to access CPMS ChargingPointManagementService, which send request to Battery Handler to get the status of batteries present in a charging point from the sensors.
R10(CPMS)	cpCA, cpCPMS, cpCSH	CPO Operator go through ClientApplication to access CPMS ChargingPointManagementService, and ChargingSocketHandler in order to get information of vehicle connected to all charging socket in a charging point.
R11(CPMS)	cpCA, cpCPMS, cpMODEL, cpDB	CPO Operator go through ClientApplication to access CPMS ChargingPointManagementService, in order to extract all booking history of a charging socket present in a charging point from DB.
R12(CPMS)	cpDIH	DSOInformationHandler component is responsible at querying DSOs's uniform API to get their current price of energy.
R13(CPMS)	cpCA, cpCPMS	CPO Operator go through ClientApplication to access CPMS ChargingPointManagementService, in order to switch the energy source control mode of a charging point.
R14(CPMS)	cpCA, cpCPMS, cpDIH	CPO Operator go through ClientApplication to access CPMS ChargingPointManagementService and DSOInformationHandler, in order to visualize the offers made by DSOs.

Requirement	Components	Description
R15(CPMS)	cpCA, cpCPMS, cpEAS, cpBH, cpDIH	CPO Operator go through ClientApplication to access CPMS ChargingPointManagementService, that coordinate with other components in order to provide all informations necessary to user to decide where extract energy from.
R16(CPMS)	cpCA, cpCPMS, cpBH, cpDIH, cpEAS, cpMODEL, cpDB	CPO Operator go through ClientApplication to access CPMS ChargingPointManagementService, that coordinate with other components in order to provide all informations necessary to user to decide whether store energy in batteries.
R17(CPMS)	cpBRH, cpMODEL, cpDB	BookingRequestHandler component is responsible at answering incoming booking request, verifying whether the request is fulfillable in a charging point.
R18(CPMS)	cpCRH, cpCSH, cpMODEL, cpDB	ChargingRequestHandler component is responsible at handling charging request, coordinating the charging socket and verifying booking in DB.

5. Implementation, integration and test plan

In this section first the implementation of the whole system and then its integration & test plan will be described.

5.1 Implementation

Implementation plan here will be divided into several various groups, defined by their role in the whole system.

- **eMSP Client Application**
- Here the the light mobile application will be developed. This is the application which ordinary drivers will use in order to book the charging.
- **CPMS Client Application**
- Desktop application which is not necessary light and easy to use, since it will be used by trained stuff.
- **eMSP Logic**
- The part, where all the logic related components will be developed. It includes all the components from the eMSP Server Logic 'box' from the Component Diagram except for the eMSP Model.
- **CPMS Logic**
- Same logic stuff, but this time for the CPMS Server Logic. As previously, no CPMS Model component included here.
- **eMSP Database**
- In this part the database for eMSP itself and the eMSP Model component to be developed.
- **CPMS Database**
- Similar to the previous part. CPMS Database and CPMS Model component for the interaction with the database.

5.2 Integration and test plan

Obviously, unit tests are to be done for each component. Only after the success of the unit testing we will be able to make the integration test of the components. Regarding the testing strategy, the top-down style was chosen, thus we are testing first the low-level components (The interaction with the DBMS, logic stuff) and only then we include the presentation layer (eMSP Client Application and CPMS Client Application). The reason for such way is that we are expecting the bugs to appear

mostly in the logic layer, so using the top-down strategy we may fix them first. However, there was some exception made. Considering, that the eMSP and CPMS are two independent systems, their integration test with each other are to be done only after the integration tests of both CPMS and eMSP with their corresponding client applications. It is also important to note that we are not testing the Databases themselves, but the interaction with them. The Database systems are to be checked by company, which provides it (MySQL AB in our case). Same thing is applied to the external API regarding the DSOs and the third part payment system. The order in which the components are going to be testes is shown in diagrams below:

1. First, we have to test the eMSP Model

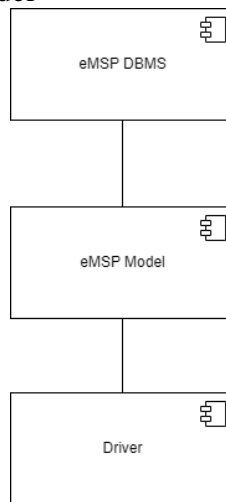


Fig. 5.1

2. Then we test 3 components which have the direct relation to the eMSP Model

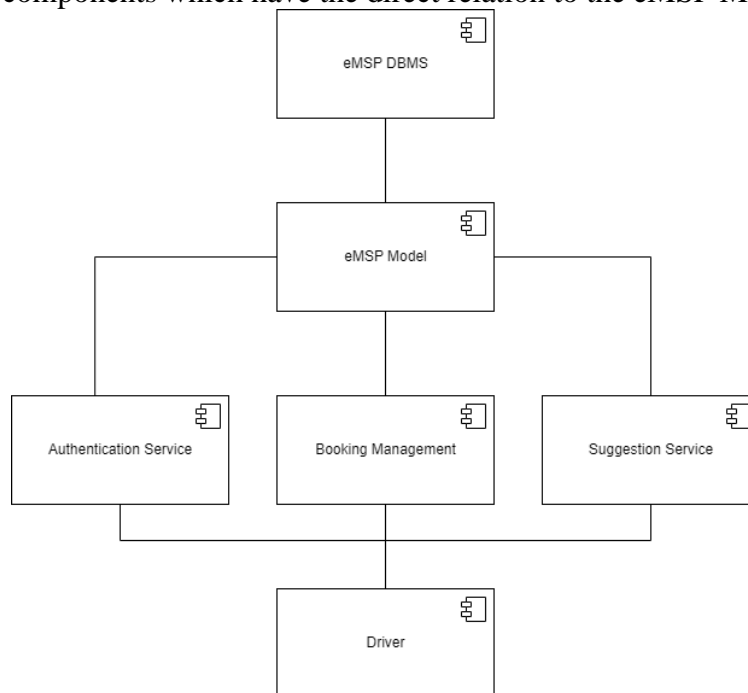


Fig. 5.2

3. Now time for ProvideInfo Service

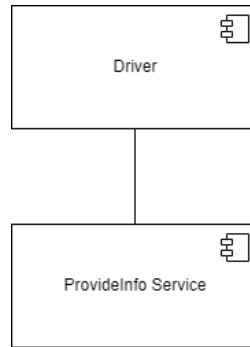


Fig. 5.3

4. Adding ChargingHandler Service

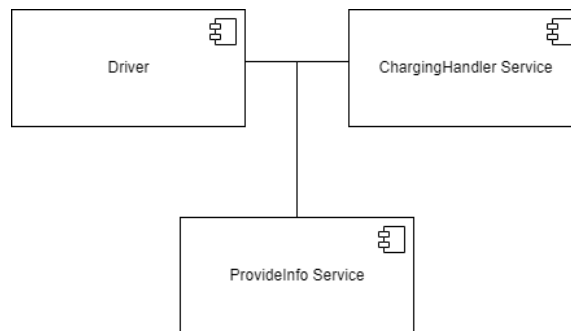


Fig. 5.4

5. Connecting other components and adding the eMSP Client Application

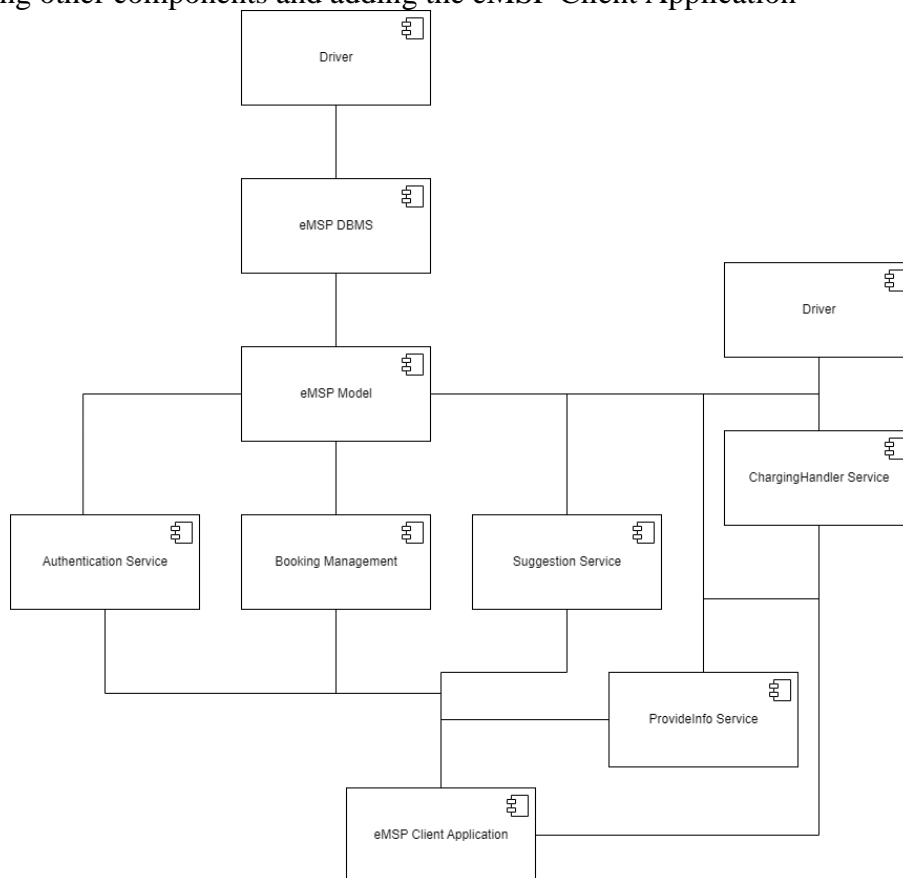


Fig. 5.5

6. Getting to the CPMS part now

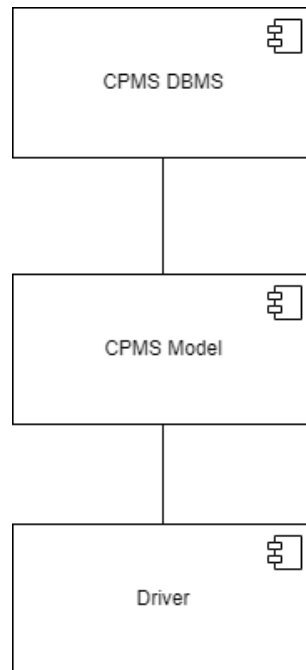


Fig. 5.6

7. Adding 4 components to CPMS Model



Fig. 5.7

8. Testing the external API and components connected to it

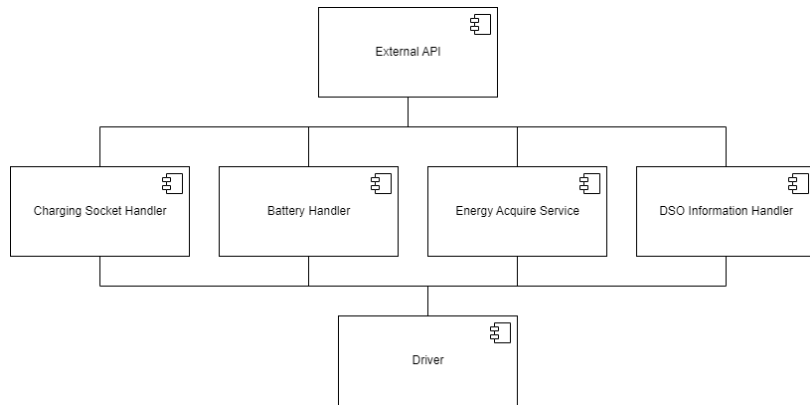


Fig. 5.8

9. Connecting ChargingRequest Handler to both of the diagrams shown previously

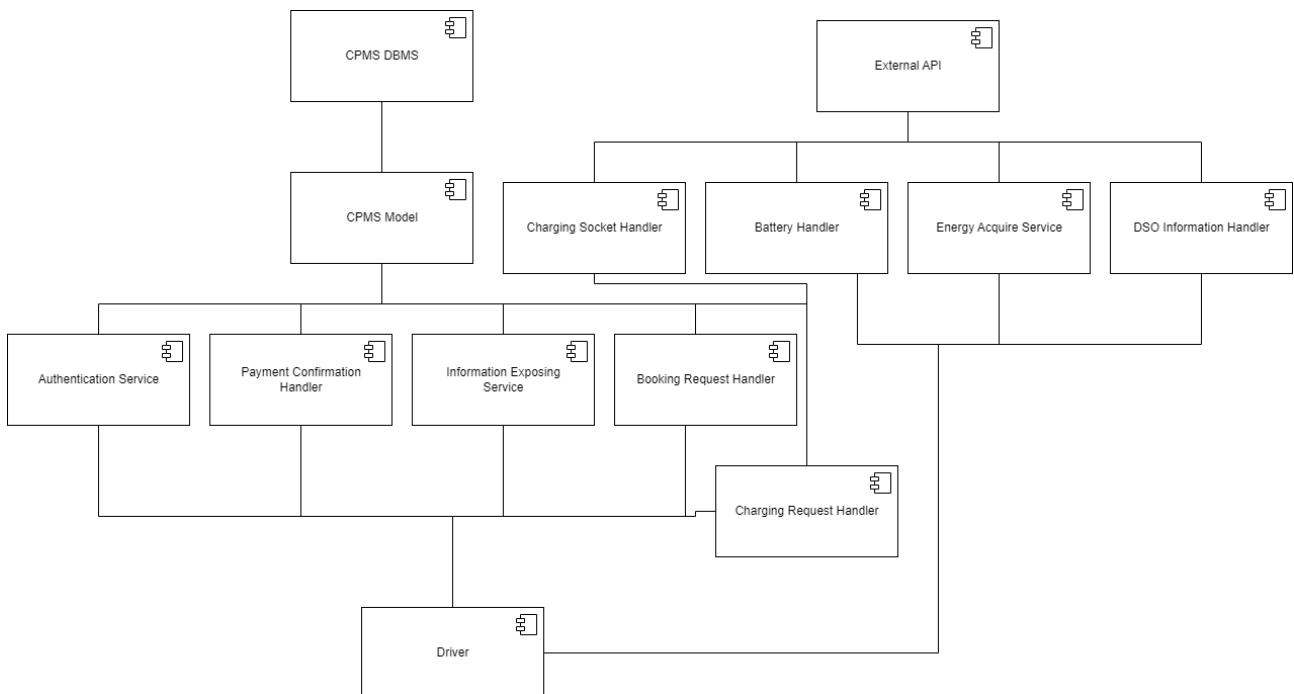


Fig. 5.9

10. Adding Charging Point Management Service

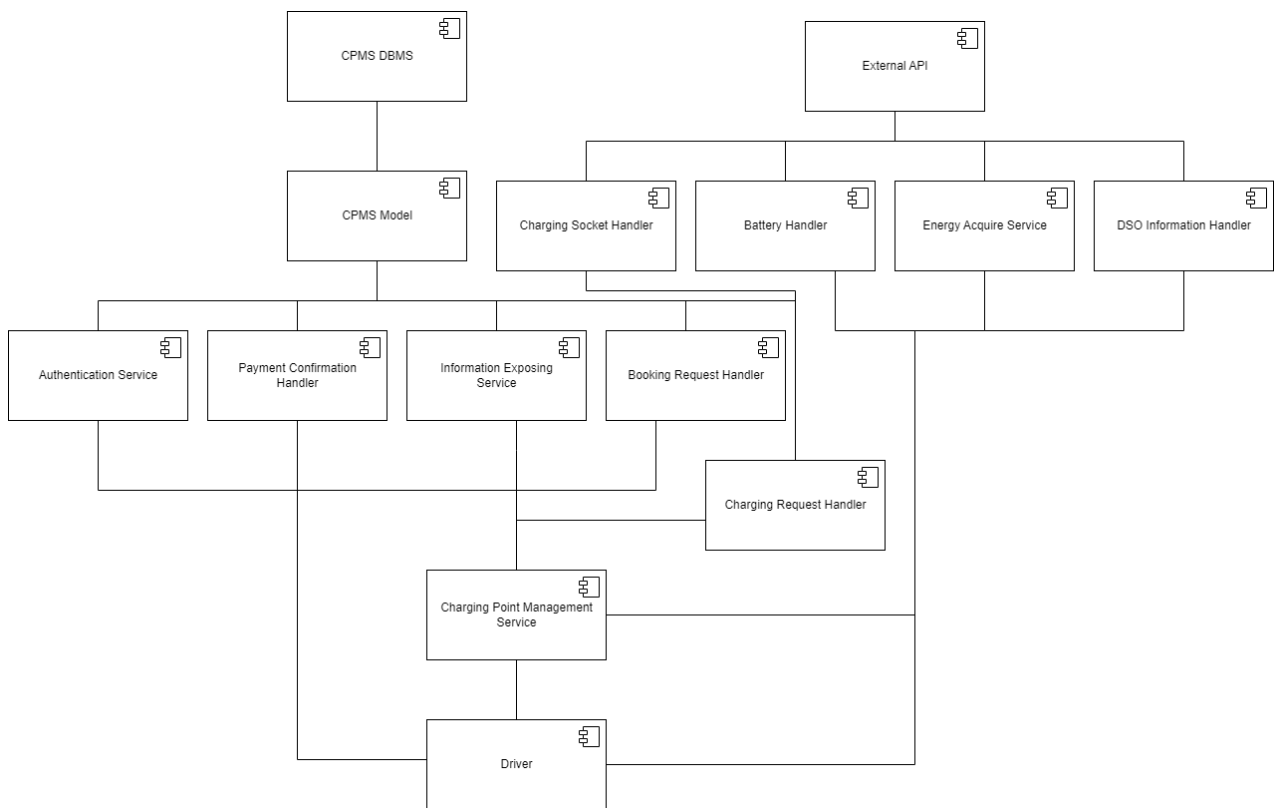


Fig. 5.10

11. Testing the whole system with the CPMS Client Application (The cross signs mean that the crossing lines are not connected)

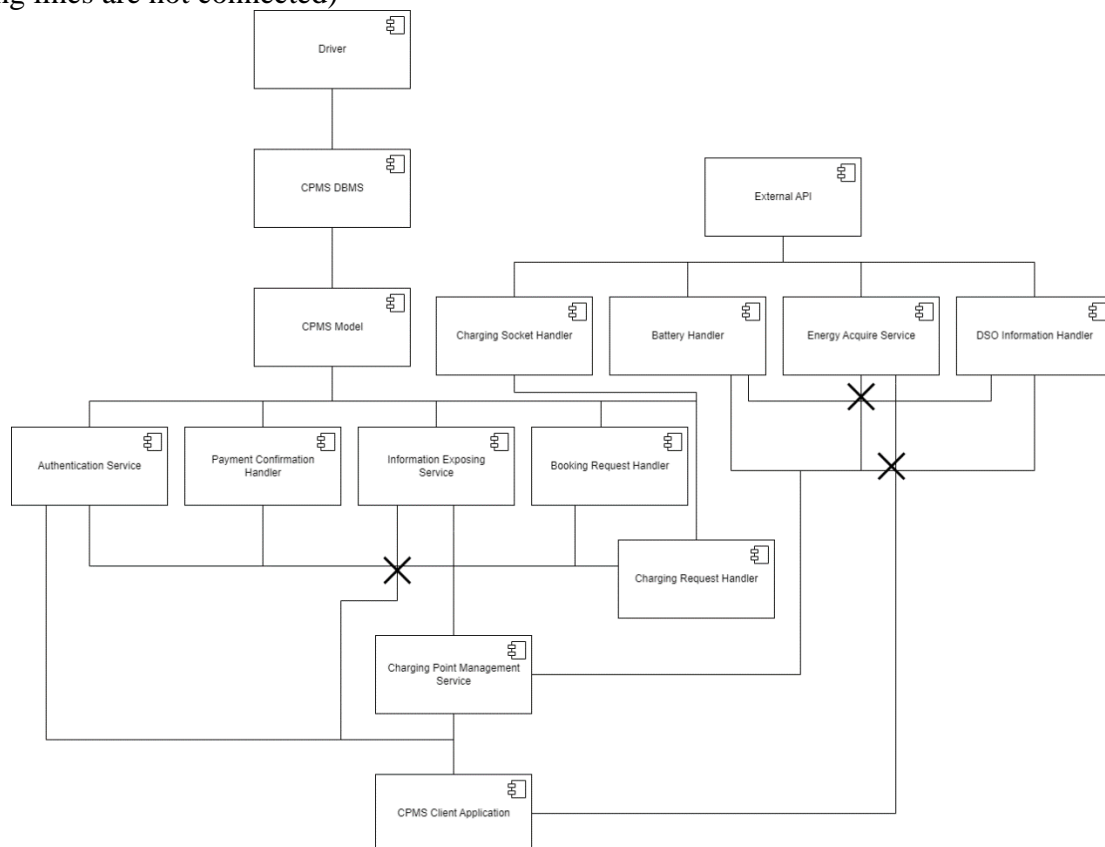


Fig. 5.11

12. The final test of the 2 systems

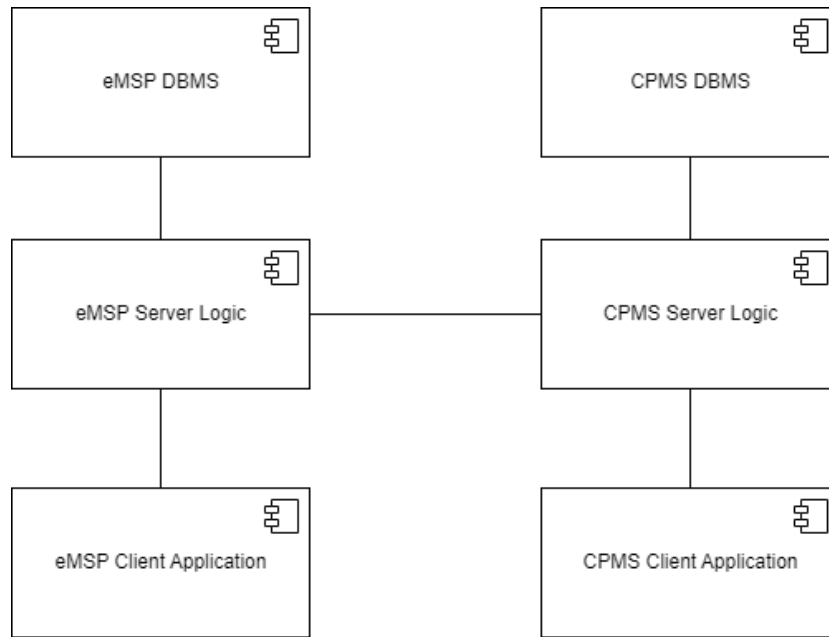


Fig. 5.12

6. Effort spent

6.1 Beliakov Maksim

Task	Time spent (h)
Brainstorming and discussion	10
Introduction	4
Architectural design	4
User interface design	15
Requirements traceability	3
Implementation, integration and test plan	1
Document refinement	1
Total	37

6.2 Aliyev Rustam

Task	Time spent
Brainstorming and discussion	10
Introduction	1
Architectural design	12
User interface design	2
Requirements traceability	6
Implementation, integration and test plan	4
Document refinement	1
Total	36

6.3 Yizhou Wu

Task	Time spent
Brainstorming and discussion	10
Introduction	0
Architectural design	8
User interface design	8
Requirements traceability	2
Implementation, integration and test plan	1
Document refinement	6
Total	35

7. References

ⁱ (<https://www.ibm.com/topics/three-tier-architecture>) IBM: What is three-tier architecture?