# 3. SPECIFIC REQUIREMENTS

## 3.1. External Interface Requirements

### 3.1.1 User Interfaces

The user interface of eMSP is an application on smart phones that will be used for users. The most parts of this application are developed for the users.

Also, because CPO can do some works like select a DSO for charging and see the information, she/he uses a website.

### 3.1.2 Hardware Interfaces

All the hardware that these applications require is:

* Users have to use a smartphone which can connect to the internet and can use GPS services, and access to their calendar.
* CPOs have to use a web browser to work with CPMSs.

### 3.1.3 Software Interfaces

The systems use Map as an external interface:

* eMSP provides public API for user to find her/his position and also find the best way for attending to the nearest charging station.
* CPOs can access to all locations of charging stations for check the status of each charging station.

### 3.1.4 Communication Interfaces

* Each eMSP communicates with CPMS of each CPO thorough API. This communicate will happen because data in eMSP put from information that CPMS has.
* Each CPO has a communication with DSOs.
* eMSP application can get access to the calendar of the user and her/his navigation system.

## 3.2. Functional Requirements

Table 9 Functional Requirements

|  |  |
| --- | --- |
| Requirement | Description |
| R1 | The eMSP must allow an unregistered end user to register |
| R2 | After an end user fills all the blanks in registration page correctly, the system must send an email to her/him, in order to confirm her/his email |
| R3 | The eMSP must allow a logged-out end user to login |
| R4 | The CPMS must allow CPO to login |
| R5 | The CPMS must allow CPO to view list of charging stations with their statuses |
| R5 | The CPMS must allow the CPO to insert offers for end users |
| R6 | The eMSP must notify the user when the charging process is finished through notification |
| R7 | The eMSP must access to the GPS of the end user smartphone for suggesting charging stations |
| R8 | The CPMS must allow a logged-out CPO to login |
| R9 | The eMSP must notify the user when there is an error during charging process through notification |
| R10 | The eMSP must allow end user to view her/his information |
| R11 | The eMSP must allow end user to enter her/his information |
| R12 | The CPMS can connect to several eMSPs through API |
| R13 | The CPMS must connect to each charging points |
| R14 | The eMSP must allow end user to pay for the service |
| R15 | The eMSP must create QR code if end user wants to pay by QR code |
| R16 | The eMSP must allow end user to book a charging point |
| R17 | The eMSP must access to the end user’s calendar |
| R18 | The eMSP must suggest charging points depending on the status of the battery |
| R19 | The CPMS must shows external status to the CPO |
| R20 | The CPMS must shows internal status to the CPO |
| R21 | The CPMS must shows locations of charging stations |
| R22 | The CPMS must allow CPO to insert the current price for services |
| R23 | The eMSP must show the receipt after end user paid for the services |
| R24 | When the car is charging, the eMSP must show the exact status of the battery to the end user |
| R25 | When the end user is in the main page, eMSP must shows the list of all charging stations in order |
| R26 | eMSP must shows the prices of each charging stations to the end user in main page |
| R27 | eMSP must allow end user to select the way of payment |
| R28 | eMSP must show the ways of payments to the end user |
| R29 | After an end user inserted all the personal information related to her/him, the eMSP must allow her/him to confirm |
| R30 | eMSP must allow the end user to edit information about her/his credit card |
| R31 | After an end user inserted all the information related to her/him credit card, the eMSP must allow her/him to confirm |
| R32 | eMSP must allow end user to cancel the time she/he booked |
| R33 | The CPMS must shows available DSOs to the CPO |
| R34 | When end user wants to see available charging stations, the eMSP must show special offers that CPO considered for the end user |
| R35 | After an end user books a charging point, the eMSP must send a notification to her/him to inform her/him |
| R36 | eMSP must allow the end user to edit her/his information |
| R37 | eMSP must allow the end user to see the status of her/his car battery |
| R38 | The CPMS must allow CPO to view the current price of energy through DSOs |
| R39 | The eMSP must show available charging stations to the user |
| R40 | The eMSP must show charging stations’ status to the user |

### 3.2.1. Use Case Diagrams

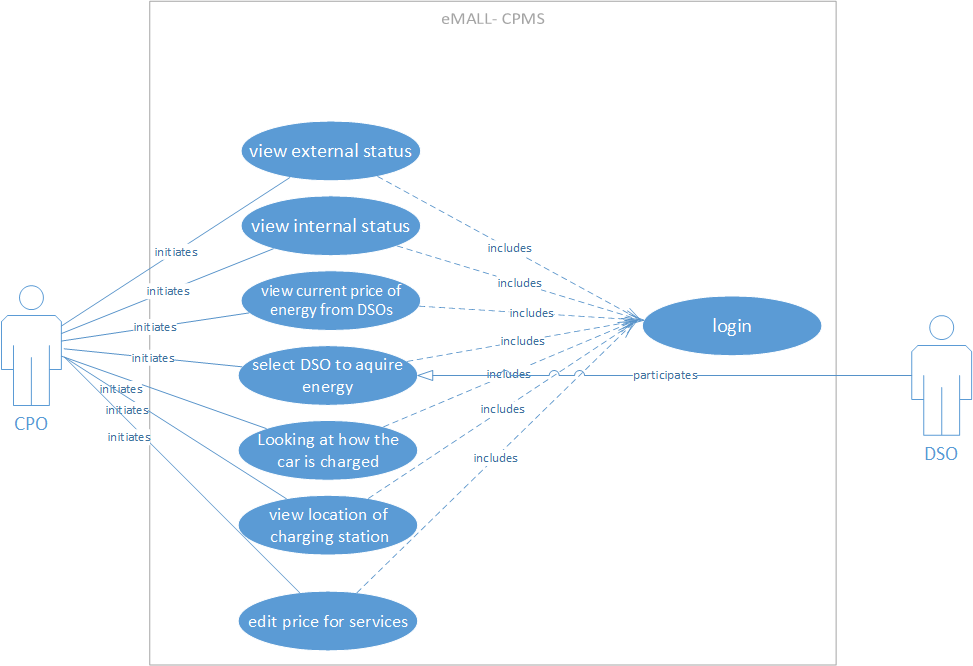


Figure 5 Use Case Diagram for CPO

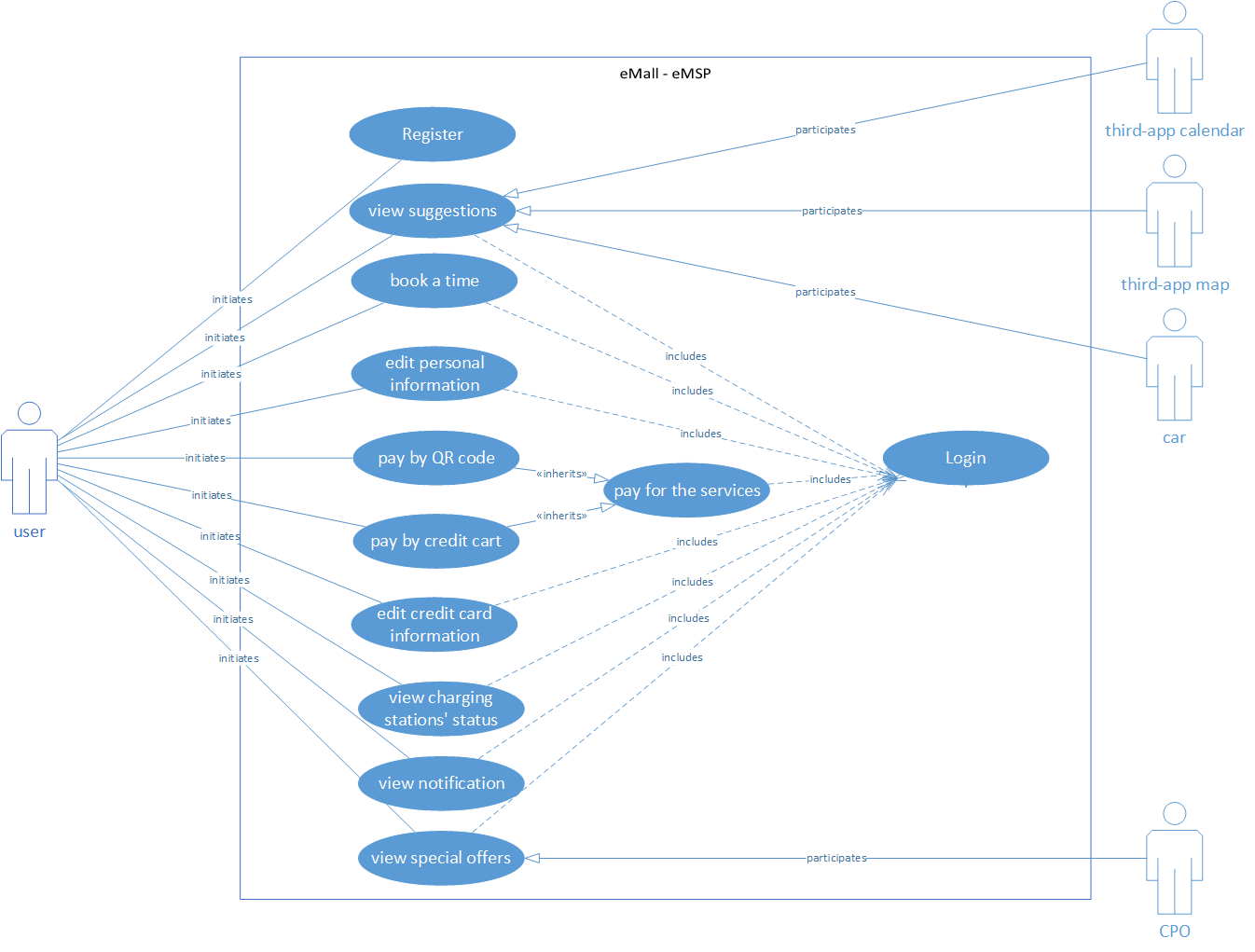


Figure 6 Use Case Diagram for End User

### 3.2.2. Use Cases

* View external status

Table 10 use case for CPO - View external status

|  |  |
| --- | --- |
| Name | View External Status |
| Actor | CPO |
| Entry Conditions | The CPO has already Logged to the CPMS dashboard in successfully. |
| Events Flow | 1. Click on the Charging Station lists on the panel. 2. Dashboard shows a list of this CPO charging stations. 3. CPO select a Charging Stations. 4. Dashboard shows all charging port and their type. 5. CPO can see all external status of charging station in top of the page. 6. CPO can see all external status of the port by clicking on the port. |
| Exit Conditions | Information be presented successfully. |
| Exceptions | * Charging Stations get disconnected from the network and not be able to send the Charging Point status. * The CPMS was down and not able to send the list of Charging Stations to the CPO panel. |

* View internal status

Table 11 use case for CPO - View internal status

|  |  |
| --- | --- |
| Name | View Internal Status |
| Actor | CPO |
| Entry Conditions | The CPO has already Logged to the CPMS dashboard in successfully. |
| Events Flow | 1. Click on the Charging Station lists on the panel. 2. Dashboard shows a list of this CPO charging stations. 3. CPO select a Charging Stations. 4. Dashboard shows all charging port and their type. 5. CPO can see all internal status of charging station in top of the page. 6. CPO can see all internal status of the port by clicking on the port. |
| Exit Conditions | Information be presented successfully. |
| Exceptions | * Charging Stations get disconnected from the network and not be able to send the Charging Points status * The CPMS was down and not able to send the list of Charging Stations to the CPO panel |

* Edit current price of energy

Table 12 use case for CPO - Edit current price of energy

|  |  |
| --- | --- |
| Name | Edit current price of energy |
| Actor | CPO |
| Entry Conditions | He/she access to the (admin panel). The CPO has already Logged in successfully. |
| Events Flow | 1. CPO open the panel 2. From dashboard, choose price settings 3. Edit current price and save the settings |
| Exit Conditions | Price gets updated successfully |
| Exceptions | * The price was not valid. E.g., negative value * eMSP was not able to update the database. (Database error, infrastructure problem and so on) |

* Select DSO to acquire energy

Table 13 use case for CPO - Select DSO to acquire energy

|  |  |
| --- | --- |
| Name | Select DSO to acquire energy |
| Actor | CPO |
| Entry Conditions | The CPO has already Logged to the CPMS dashboard in successfully. |
| Events Flow | 1. CPO go to DSO section in the CPMS dashboard. 2. CPO can see all of DSOs name and their prices. 3. CPO can request them by clicking on forward button to send a request. 4. After that DSO accept CPO request, CPMS shows a notification that DSO accept your request. |
| Exit Conditions | DSO accept to provide energy for CPO and the CPO update the eMSP database successfully. |
| Exceptions | 1. DSO is not available 2. DSO does not accept to provide energy for CPO 3. eMSP was not able to update the database. (Database error, infrastructure problem and so on) |

* Select kind of getting energy

Table 14 use case for CPO - Select kind of getting energy

|  |  |
| --- | --- |
| Name | Select kind of getting energy |
| Actor | CPO |
| Entry Conditions | The CPO has already Logged to the CPMS dashboard in successfully. |
| Events Flow | 1. CPO go to list of charging point. 2. Dashboard shows list of CPO all charging stations. 3. CPO by clicking on internal status of specific charging point can see the whole internal status of charging point. 4. CPO can change the power source of one socket from battery to DSOs energy. 5. CPMS can handle this situation when battery are drained and automatically change to DSO energy. 6. CPO can set preferred Energy in top of the page in the dropdown. This causes the fact that CPMS can automatically use DSO after that use batteries or reverse. |
| Exit Conditions | 1. DSO accept to provide energy for CPO if the DSO has been selected. 2. CPO update the eMSP database successfully. |
| Exceptions | 1. DSO is not available. 2. DSO does not accept to provide energy for CPO. 3. eMSP was not able to update the database. |

* View location of charging station

Table 15 use case for CPO - View location of charging station

|  |  |
| --- | --- |
| Name | View location of a charging station |
| Actor | CPO |
| Entry Conditions | The CPO has already Logged to the CPMS dashboard in successfully. |
| Events Flow | 1. CPO go to list of charging point. 2. Dashboard shows list of CPO all charging stations. 3. CPO click on eye icon in front of any item of the list. 4. Dashboard shows the location of the charging station near to Internal Status button External Status button. |
| Exit Conditions | Charging Station information, including the location be shown successfully. |
| Exceptions | 1. Sub systems including CPMS and eMSP not be available. 2. The eMSP database was not responding (for instance, because of the heavy load) |

* Register

Table 16 use case for user - Register

|  |  |
| --- | --- |
| Name | Register |
| Actor | End user |
| Entry Conditions | End user open the application |
| Events Flow | 1. End user selects to register 2. End user fill all the blank fields. 3. End user click on register button. 4. The system sends an email to his/her. 5. End user confirms his/her email. 6. The system allows him/her to login. |
| Exit Conditions | The end user is registered in the app. |
| Exceptions | 1. End user doesn’t check the policies checkbox. 2. End user doesn’t fill all the fields. In this case, application shows warning. 3. the email address already exist in the system. In this case, application warns him/her to login. |

* Login

Table 17 use case for user - Login

|  |  |
| --- | --- |
| Name | Login |
| Actor | End user |
| Entry Conditions | 1. End user open the application. 2. He/She already registered. |
| Events Flow | 1. End user select login button. 2. End user enter email addresses as user name password to login. 3. End user click on login button. |
| Exit Conditions | The system allows the user to login. |
| Exceptions | 1. End user enters the wrong username. 2. End user enters the wrong password.   In both cases, system warns him/her that user with this data not found. |

* View special offers

Table 18 use case for user - View special offers

|  |  |
| --- | --- |
| Name | View special offers |
| Actor | End user |
| Entry Conditions | 1. End user open the application. 2. He/she already signed in. |
| Events Flow | 1. End user clicks on today’s offers button. 2. Application shows a page that contains list of all the offers for the user. 3. End user can click on one offer and accept it. |
| Exit Conditions | Accept an offer or decline all offers. |
| Exceptions | He/she doesn’t login to the application. In this case, user must login to the application to see offers. |

* Book a time

Table 19 use case for user - Book a time

|  |  |
| --- | --- |
| Name | Book a time |
| Actor | End user |
| Entry Conditions | 1. End user has already Logged in successfully. 2. In the application, user go to charging stations section. |
| Events Flow | 1. End user click on Plus icon to save new booking. 2. End user selects time period what he/she want to book. 3. The application show all charging station that have available port for end user car. 4. End user select one charging station that he/she want. 5. The application ask for what time and how long he/she wants To book. |
| Exit Conditions | End user click on confirm/cancel button. |
| Exceptions | End user enter wrong time period. |

* Edit personal information

Table 20 use case for user - Edit personal information

|  |  |
| --- | --- |
| Name | Edit personal information |
| Actor | End user |
| Entry Conditions | 1. End user has already logged in successfully. 2. In the application, user go to edit personal data section. |
| Events Flow | 1. End user click on profile icon. 2. Application show the information about user. 3. End user can change email address, his/her car model, his/her profile photo. |
| Exit Conditions | End user click on save/cancel button. |
| Exceptions | 1. End user enters invalid email address. 2. End user selects image that the volume is more than 5MB. |

* Pay by QR code

Table 21 use case for user - Pay by QR code

|  |  |
| --- | --- |
| Name | Pay by QR code. |
| Actor | End user |
| Entry Conditions | 1. End user has already Logged in successfully. 2. Application shows a notification that charging is finish. |
| Events Flow | 1. There is one LCD on that port which user put in the car for charging, after the charging finished it shows the QR code. 2. End user scan this QR code with his/her application. 3. Application ask for payment method. 4. End user can select to pay with his/her saved cart, third-party application like google pay or apple pay or pay with money that saved to his/her account. |
| Exit Conditions | Payment is successful. |
| Exceptions | 1. End user select to pay with hid/her account money but there isn’t enough money. 2. End user doesn’t have enough money in his/her bank account. |

* Pay by credit cart

Table 22 use case for user - Pay by credit cart

|  |  |
| --- | --- |
| Name | Pay by credit cart |
| Actor | End user |
| Entry Conditions | 1. End user has already Logged in successfully. 2. Application shows a notification that charging is finish. |
| Events Flow | 1. End user click on notification. 2. Application open in car status page. 3. End user see payment alert about payment in top of application. 4. End user click on alert. 5. Application open in payment method page. 6. End user click on credit cart. 7. Application send request to eMSP. 8. eMSP process the request and send response to mobile application of user about deducting money. |
| Exit Conditions | Payment is successful or error occur in payment. |
| Exceptions | End user doesn’t have enough money in his/her bank account. |

* Edit credit card information

Table 23 use case for user - Edit credit card information

|  |  |
| --- | --- |
| Name | Edit credit cart information |
| Actor | End user |
| Entry Conditions | 1. End user has already Logged in successfully. 2. End user go to edit payment method section in the application. |
| Events Flow | 1. Application shows the list of all payment method such as all credit cart and account balance. 2. End user can enter new credit cart for his/her payment by pressing on Plus icon or edit credit card by click on pen icon. 3. End user can edit his/her credit cart information like cart number or cart name and expire date. |
| Exit Conditions | End user click on Save/Cancel button.  If user click on save, he/she must verify his/her cart information. |
| Exceptions | 1. End user enter invalid cart number, expire date or cart name. 2. End user doesn’t verify his/her cart account. |

* End User Get Some Suggestion

Table 24 use case for user - End User Get Some Suggestion

|  |  |
| --- | --- |
| Name | End User Get Some Suggestion |
| Actor | End User |
| Entry Conditions | 1. End user has already Logged in successfully. 2. End user go to suggestion section. |
| Events Flow | 1. Give location permission to application. 2. Give calendar permission to application. 3. Enable phones GPS. 4. Enable phones Bluetooth. 5. After a few seconds, Application show some suggestion to user. |
| Exit Conditions | End user confirm or decline one/all suggestion. |
| Exceptions | End user doesn’t give permission to application. |

### 3.2.3. Sequence Diagrams

* CPO views location of charging station

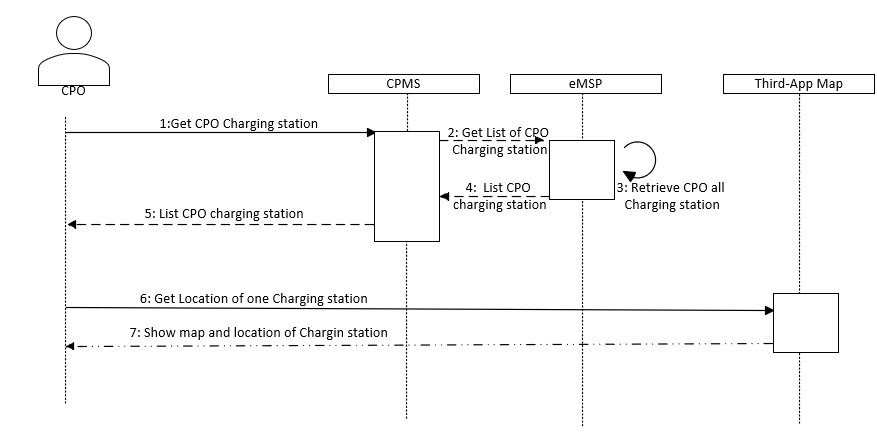


Figure 7 CPO views location of charging station

* Select DSO to aquire energy

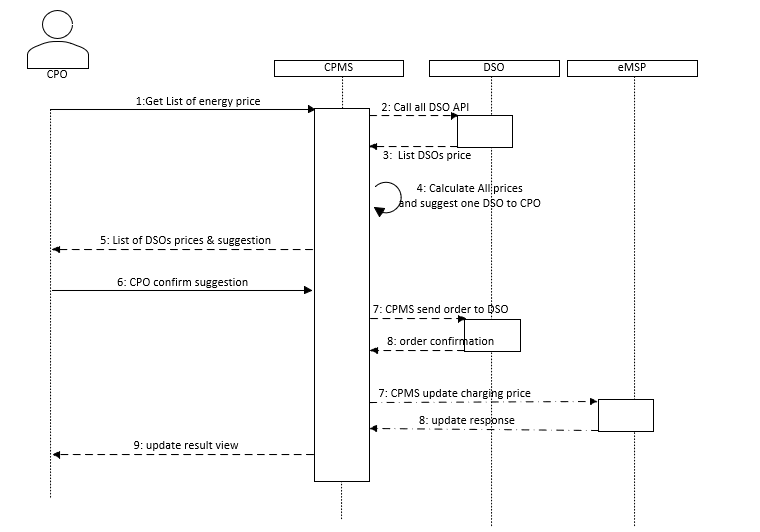


Figure 8 Select DSO to acquire energy

* CPO edit price of service

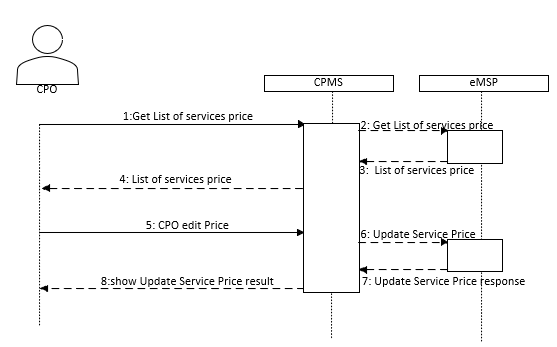


Figure 9 CPO edit price of service

* CPO views external status

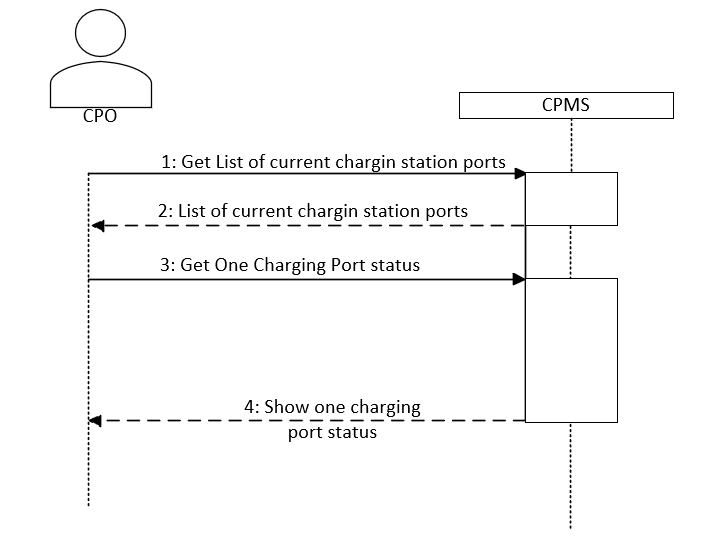


Figure 10 CPO views external status

* CPO views Internal status

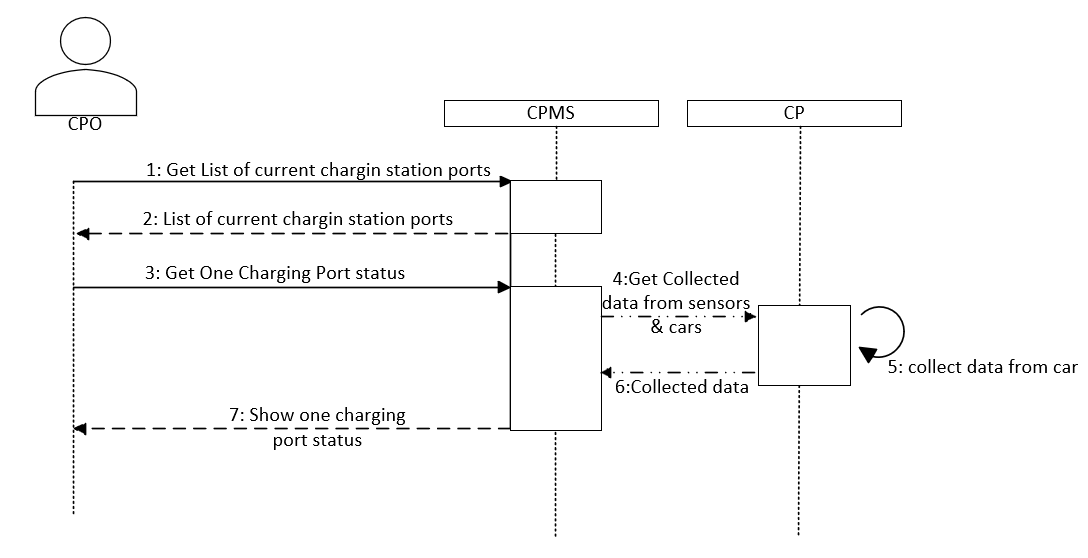


Figure 11 CPO views Internal status

* End User Get Seggestion

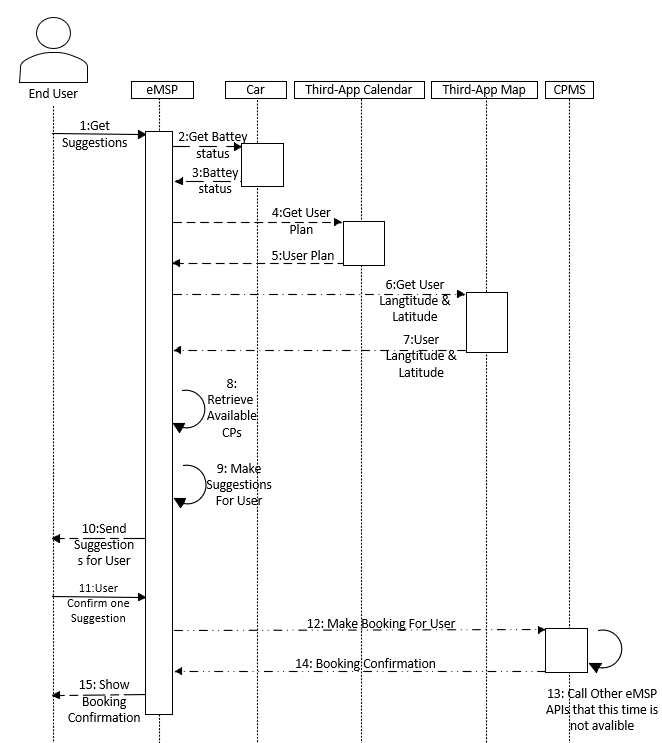


Figure 12 End User Get Seggestion

* End User Pay by Account Balance

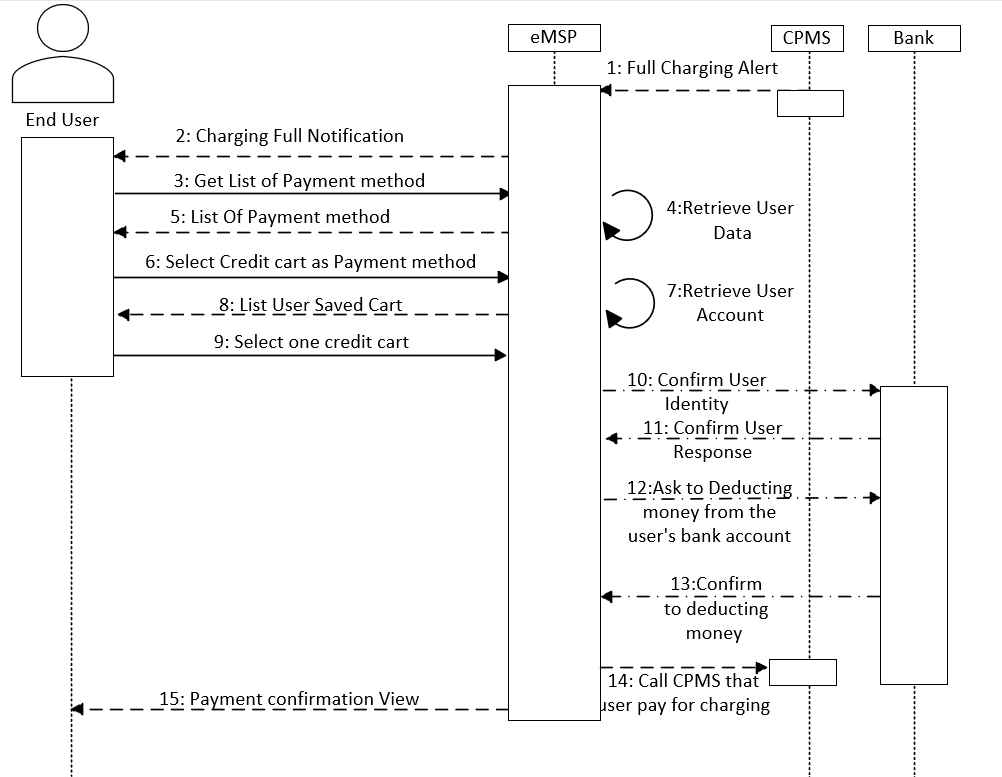


Figure 13 End User Pay by Account Balance

* CPO Save Offers

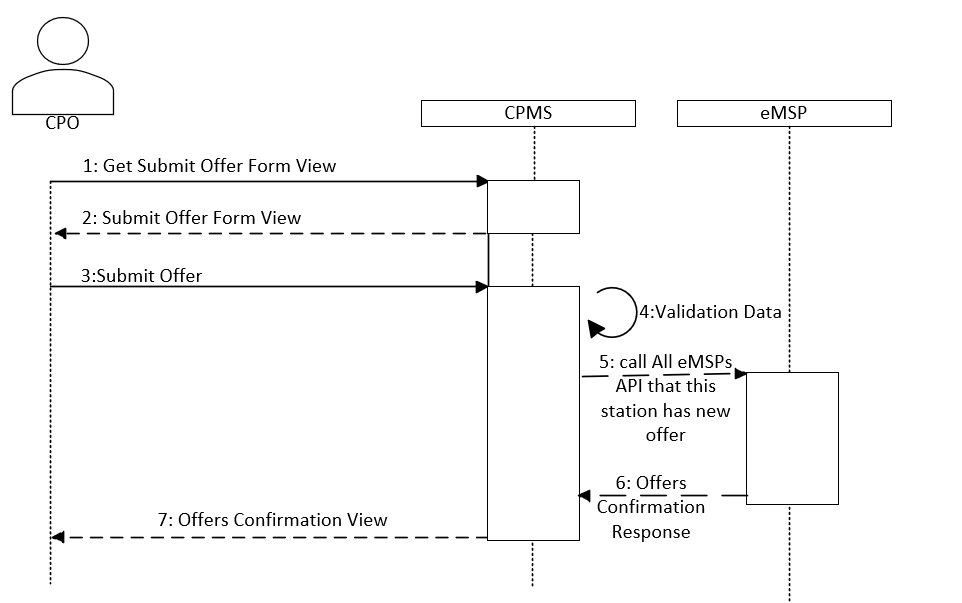


Figure 14 CPO Save Offers

* End User Pay by Account Balance

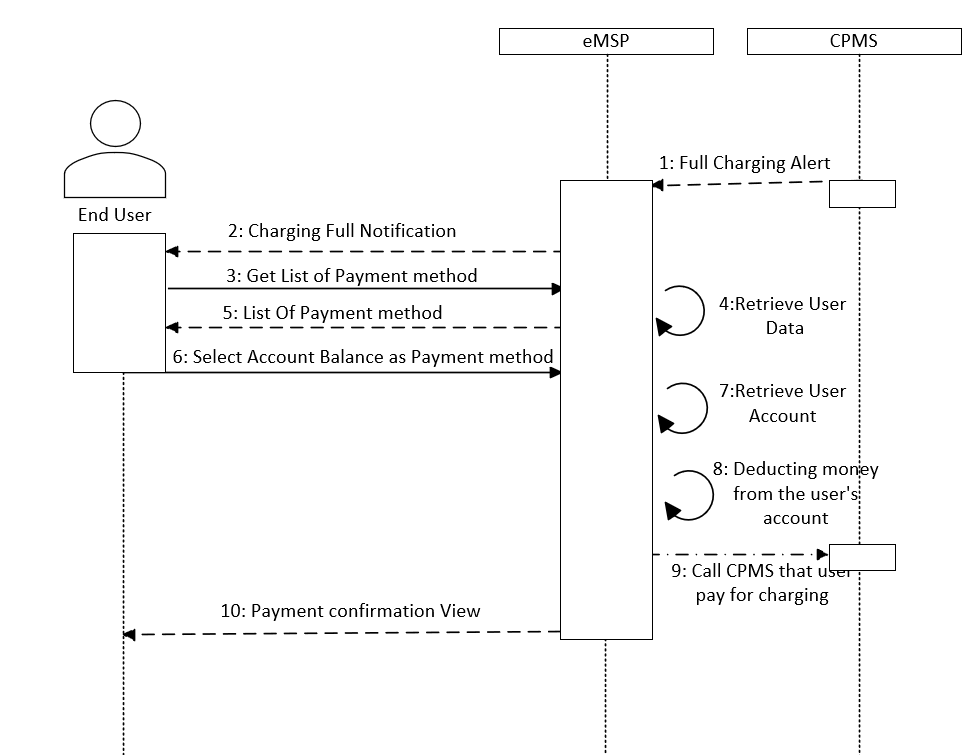


Figure 15 End User Pay by Account Balance

* End User Booking

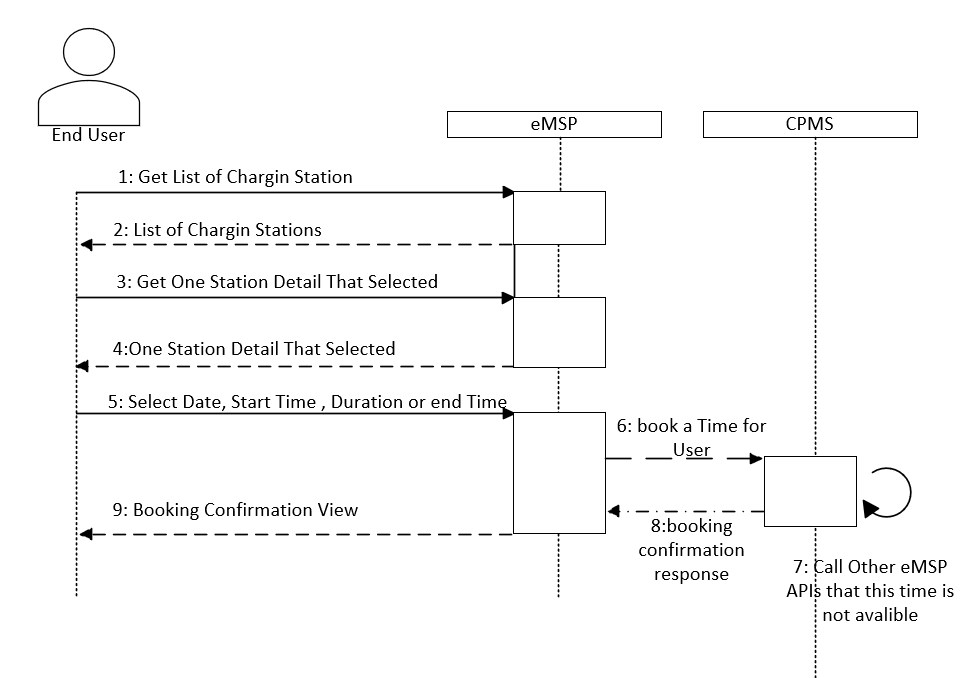


Figure 16 End User Booking

* End User login

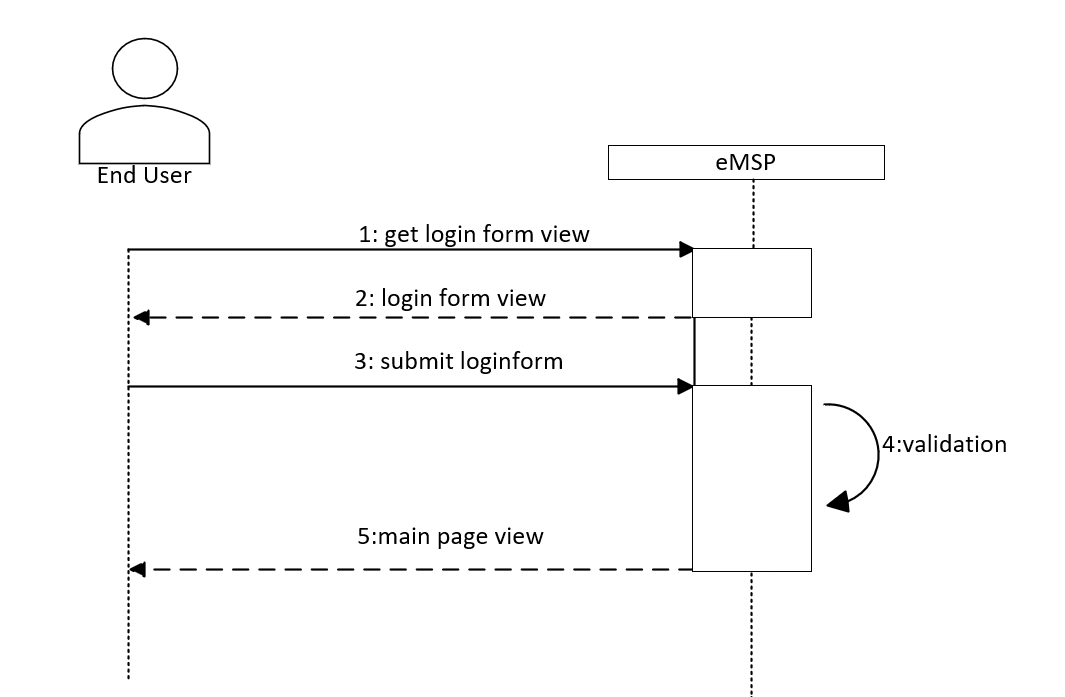


Figure 17 End User login

* End User registration

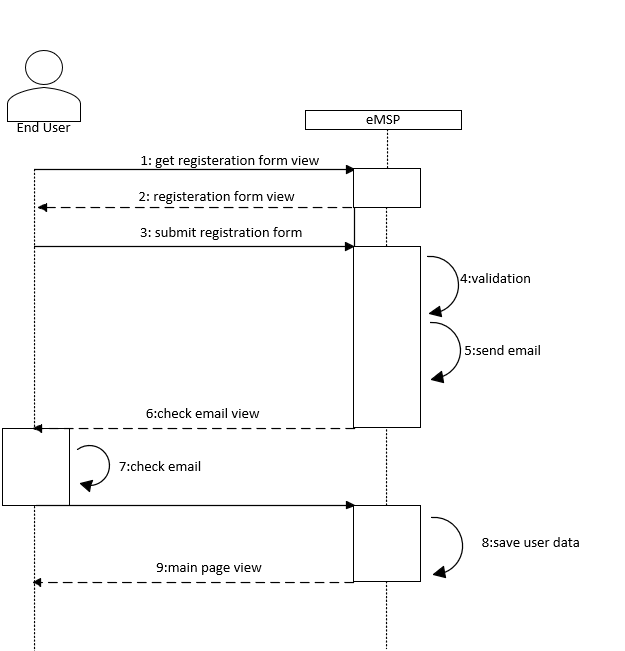


Figure 18 End User registration

* CPO Get Price form DSOs

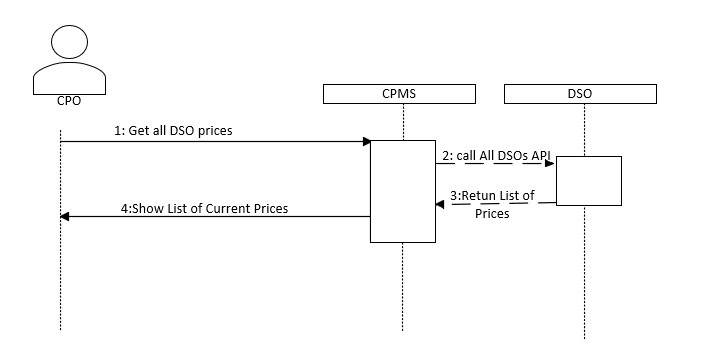


Figure 19 CPO Get Price form DSOs

### 3.2.4. Mapping on Requirements

Table 25 Mapping on Requirements

|  |  |
| --- | --- |
| Use case | Requirements |
| view external status | R4, R5, R8, R19 |
| view internal status | R4, R5, R8, R20 |
| Login (CPO) | R4 |
| Edit current price of energy | R22, R38 |
| select DSO to acquire energy | R33 |
| select kind of getting energy | R5, R33 |
| view location of charging station | R21 |
| Register | R1, R2, R30, R31, R36 |
| Login (user) | R3, R4, R8 |
| view special offers | R7, R18, R34 |
| book a time | R16, R32, R35 |
| edit personal information | R29, R30, R36 |
| pay by QR code | R14, R15, R23, R27, R28 |
| pay by credit cart | R14, R23, R27, R28, R30, R31 |
| edit credit card information | R30, R31 |
| view charging station’s status | R39, R40 |
| view notification | R6, R9, R35 |

## 3.3 Performance Requirements

* The system must be able to serve a great number of users simultaneously.
* The system must guarantee correct responses.
* The system must be able to send a response to a query less than 3 seconds since it has been received.
* The system must be available 99% of the time.

## 3.4. Design Constraints

### 3.4.1 Standards compliance

* The eMSP must manage the data retrieved from the users in respect with the privacy laws.
* The eMSP must require the customer the permission to retrieve data regarding the position.
* The CPMS must manage data in respect with the privacy laws.

### 3.4.2 Hardware limitations

* The web browser or the smartphone which the user and the CPO are using must has the ability of connecting to the internet and using GPS services.
* The smartphone which end user has must have an ability to scan QR code.
* All systems must have enough memory to run the application.

### 3.4.3 Any other constraint

* The information which is related to battery status of the user's car must be accurate 99% of the time.
* All battery charging prices must be updated periodically and displayed to the user through the eMSP.

## 3.5. Software System Attributes

### 3.5.1 Reliability

The system must be able to run continuously without any interrupts. Reliability of the system depends on the services of the system, and should be up for a 99% of time. This means the MTTR or downtime should be 365 days per years. In order to guarantee this time of downtime the system must have an appropriate infrastructure with a fully backup system located in different office that replicates the core services for covering general failure of the main system.

### 3.5.2 Availability

Like other kinds of charging stations that are available 24 hours a day, these kinds of charging stations must be like them. So, our system needs high availability. It should be noted that CPOs work in two shifts, morning and night and system is always available for end users.

### 3.5.3 Security

The provided information by CPOs is not sensitive, thus we don’t need high security for the CPMS as a software. But regarding eMSP, because our system keeps information about credit card of user, and account balances, we need high security for eMSP.

Generally, for preventing some problems, we need to encrypt stored data and also the password of the users hashed before stored.

### 3.5.4 Maintainability

The software must be written in python and codes must be written with good standards. The occurred problems must be detected easily and solved simply. The problem in one component must have not interrupted other components.

### 3.5.5 Portability

The eMSP must be designed simply and implemented on different smartphones platforms. This software runs in different platforms must support Android and iOS operating systems for mobile devices. Also, the CPMS must be runs in web application through all browsers like Chrome, Microsoft edge, and Firefax.