E-Scooter App

Use Case Report

Author: Author

Revision: 0.2

No Magic

One Allen Center, 700 Central Expressway
South, Suite 110 Allen, Texas 75013 USA

Date: February 09, 2020

REVISION HISTORY

Revision	Date	Reason For Changes	Author
0.1	<date 1>	<draft>	<Author name>
0.2	<date 2>	<release>	<Author name>

TABLE OF CONTENTS

TABLE OF FIGURES

Introduction

Purpose

<This document provides an overview of use case view>

Scope

<Provide a short description of the system being specified and its purpose, including relevant benefits, objectives, and goals>

Overview

<Describe what the document contains and explain how document is organized>

E-Scooter App

Actor Bank

Actor E-Payment Service

1 UseCase Register

New customers need to download the App and register before they can use the E-Scooter rental service.

- Register is performed by Customer
- Register includes Send an email confirmation

Extension Points

- Subscription: Register

Description

Pre-Condition

Customer has downloaded the App.

Post-Condition

Customer has successfully registered and now has an account.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

Normal

Scenarios

Basic Flow

1. Fill in User Information
2. Confirm User Information

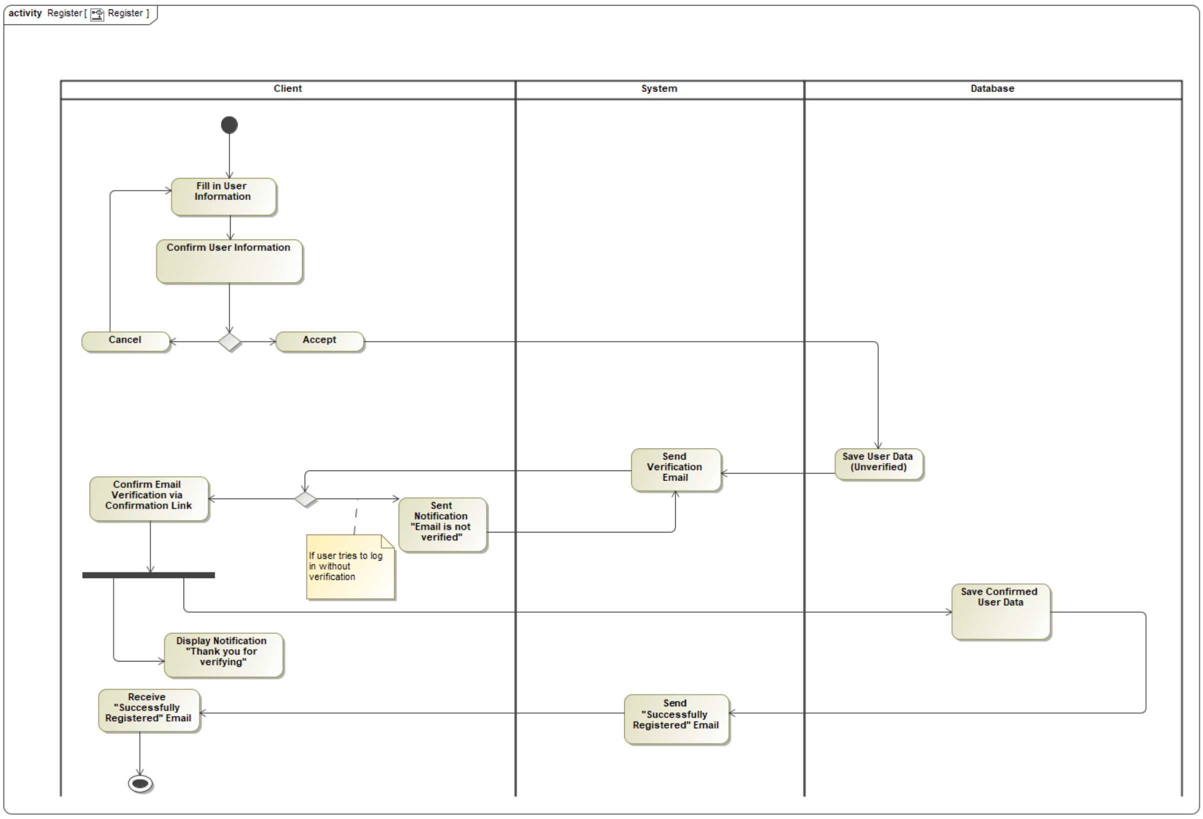
activity Register [Register]
Client
System
Database
Fill in User Information
Confirm User Information
Cancel
Accept
Save User Data (Unverified)
Send Verification Email
Confirm Email Verification via Confirmation Link
Sent Notification "Email is not verified"
If user tries to log in without verification
Save Confirmed User Data
Display Notification "Thank you for verifying"
Receive "Successfully Registered" Email
Send "Successfully Registered" Email

Figure 1. Register

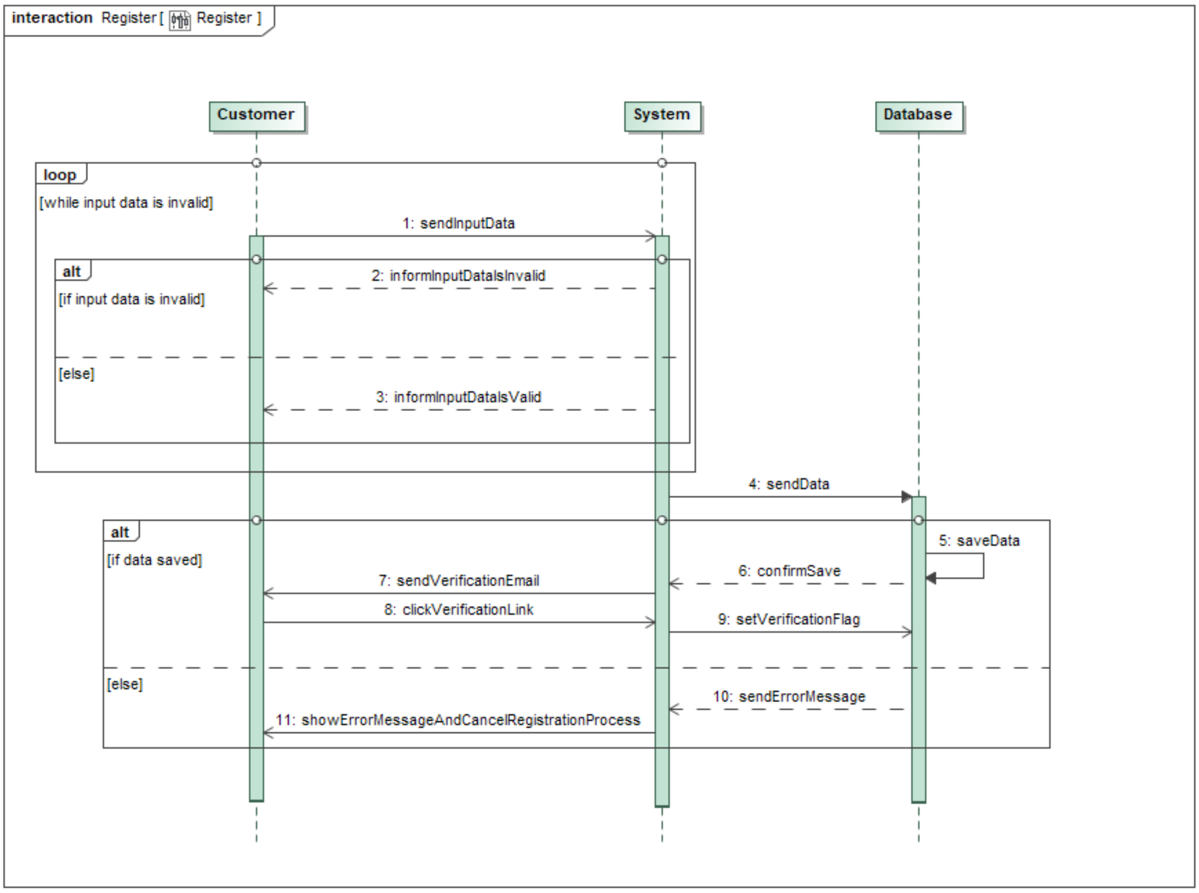

Figure 2. Register

2 UseCase Add E-Scooter to System

If a new E-Scooter arrives, the stock has to be updated and the number of available E-Scooters has to be increased.

- Add E-Scooter to System is performed by Maintenance man

Description

Pre-Condition

New Scooters are shipped to the service/maintenance location.

Post-Condition

New Scooters are registered and ready for its first customer.

Goal

New E-Scooters are properly added to the database and are ready for the first ride.

Complexity

Low Complexity

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

0. Servicer makes a quality check for the fresh E-Scooter.

1. Admin generates new QR-Code for the specific Scooter.

2. Servicer registers the Scooters to the local government registration office, in order to get the plates.

3. Admin registers the E-Scooter to the Database.

4. Servicer places the E-Scooter on the appropriate Scooter stations.

Exceptional Flow

1. Quality check fails.

2. Scooter are returned to the seller.

3. New Scooters are ordered.

4. See the Basic Flow of Events.

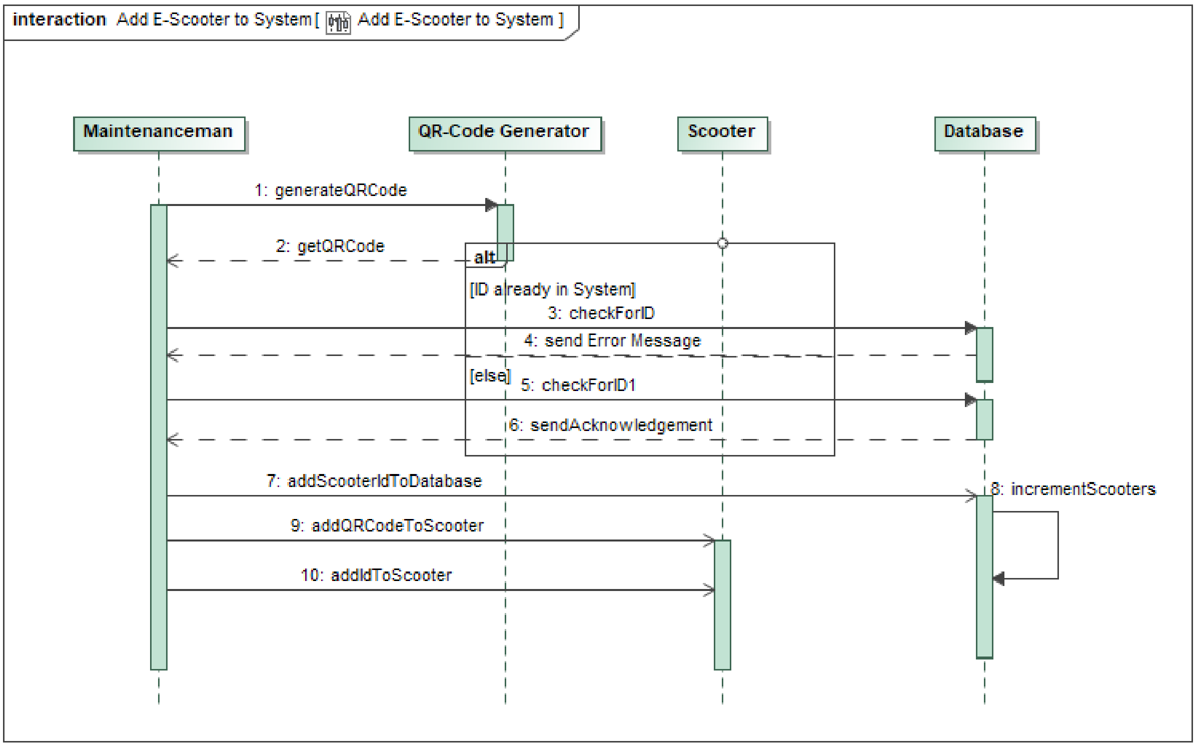

Figure 3. Add E-Scooter to System

3 UseCase Change Bank Credentials

Customers should be able to change their bank credentials.

- Change Bank Credentials extends Manage Account

Description

Pre-Condition

Customer is logged in.

Post-Condition

Customer's credit card credentials are changed.

Goal

Customers can change their credit/debit card information.

Complexity

Average Complexity

Assumption

Credit card information is either already given or it is the first time giving these details.

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Customer logs in.
2. Customer navigates to the "Payment Information/Payment Methods/Credit Debit Card".
3. Customer either enters the information for the first time or she/he changes the credentials.
4. Customer clicks the "Save" button.
5. UI refreshes.

4 UseCase Check drive statics (all users)

Admins can check the drive statistics of all/ specific customers.

- Check drive statics (all users) extends Check Statistics (Admin)

Description

Goal

See overall user's habit data pattern.

Complexity

Average Complexity

Non-Functional Requirements

Improve E-Scooter Placement

Component Complexity

Average Complexity

Priority

Low

5 UseCase Check overall revenue

Shows Admins the earnings per day/ month/ quarter/ overall revenues.

- Check overall revenue extends Check Statistics (Admin)

Description

Pre-Condition

At least multiple customers have driven the E-Scooters.

Post-Condition

Administrator knows the current revenue from the overall customer's usage.

Goal

Knowing the current income situation from the E-Scooters.

Complexity

Average Complexity

Non-Functional Requirements

Improve Revenue (Business type non-functional)

Component Complexity

Average Complexity

Priority

Below Normal

Scenarios

Basic Flow

1. Administrator checks the sum of all transactions in the database.

6 UseCase Check Statistics (Admin)

Admins can check statistics like total amount of customers, overall revenue or revenue per customer.

- Check Statistics (Admin) is performed by Administrator
- Check Statistics (Admin) is performed by Administrator

Extension Points

- Check drive statistics (all users) : Check Statistics (Admin)
- Check revenue: Check Statistics (Admin)

Description

Pre-Condition

At least enough statistically significant number of the E-Scooter users.

Post-Condition

Overall pattern of the user's drive habit data is collected and ready for the analysis.

Goal

Know the customer's drive habit in the city in order to improve the placement of the E-Scooters.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

Below Normal

7 UseCase Check usage statistics (Customer)

Customers can check their own statistics, like total price for all rides or total distance covered, etc.

- Check usage statistics (Customer) extends Manage Account

Description

Pre-Condition

Customer has logged in.

Post-Condition

Customer sees her/his drive data on a diagram.

Goal

Presents to the customer his/her drive habit data in the past days/weeks/months/ years

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

Normal

8 UseCase Check-out E-Scooter

Check-out represents the state in which the customer has scanned the QR Code of an E-Scooter in order to ride on it.

- Check-out E-Scooter is performed by Customer
- Check-out E-Scooter is performed by Customer
- Check-out E-Scooter includes Scan QR Code
- Check-out E-Scooter includes Show E-Scooter Battery Status

Description

Pre-Condition

Customer has opened the App and has logged in.

Post-Condition

Customer has successfully checked out his/her E-Scooter.

Complexity

High Complexity

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Open Application
2. Scan QR- Code

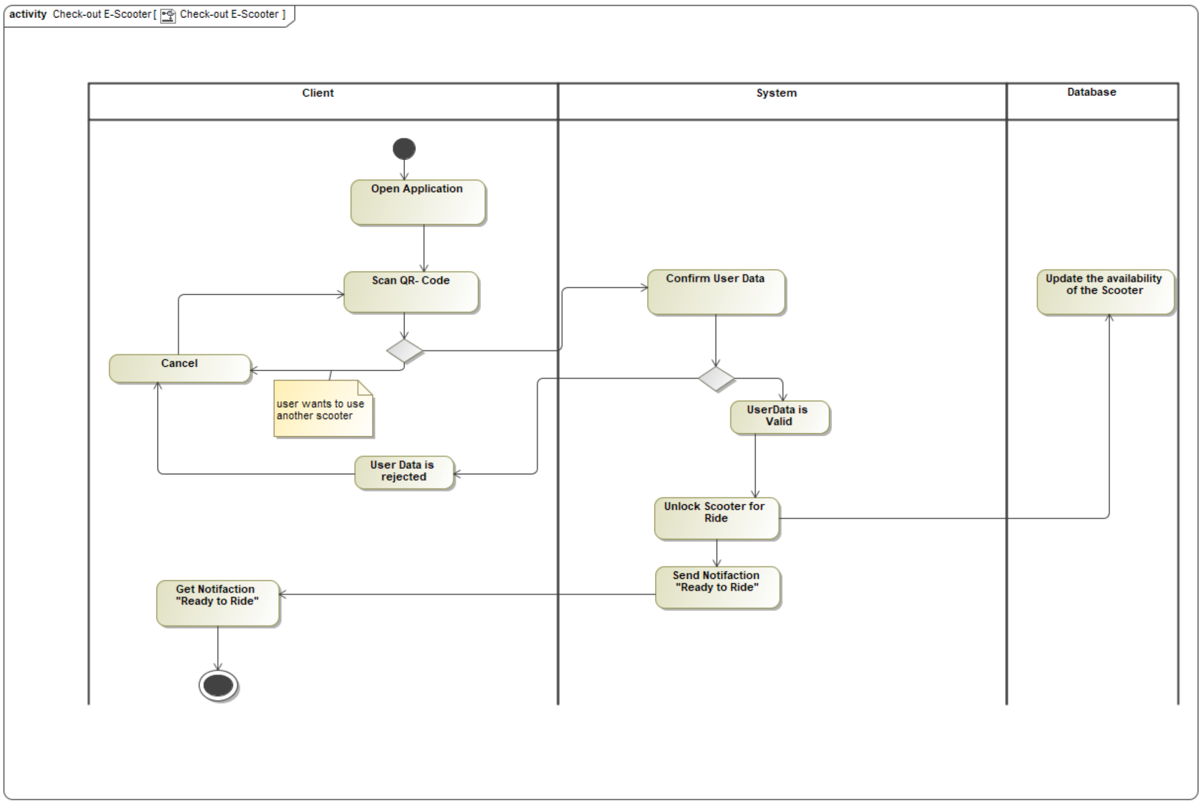
activity Check-out E-Scooter [Check-out E-Scooter]
Client
System
Database
Open Application
Scan QR- Code
Cancel
user wants to use another scooter
User Data is rejected
Get Notifcation "Ready to Ride"
Confirm User Data
UserData is Valid
Unlock Scooter for Ride
Send Notifaction "Ready to Ride"
Update the availability of the Scooter

Figure 4. Check-out E-Scooter

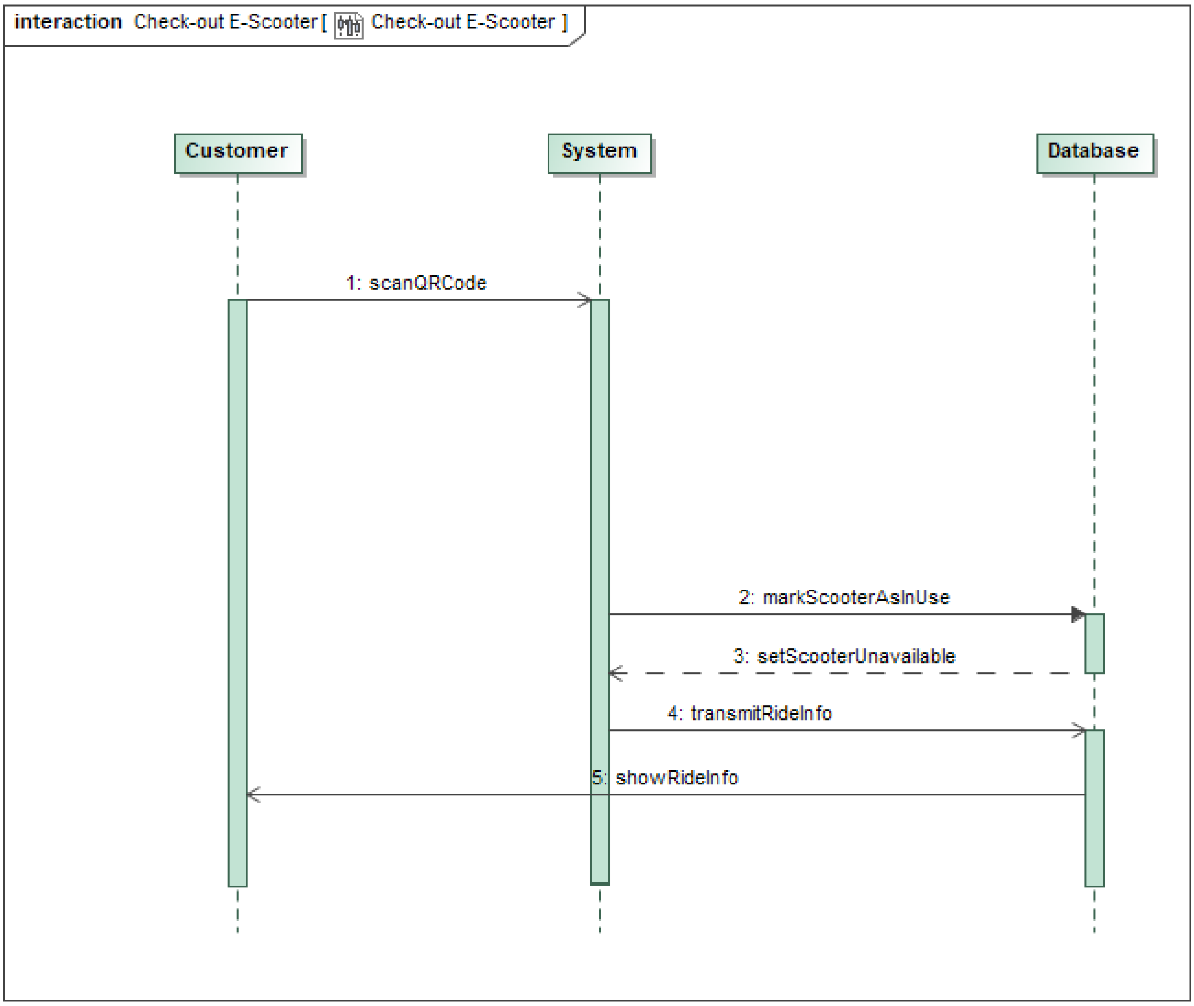

Figure 5. Check-out E-Scooter

9 UseCase Check-in E-Scooter

Check-in represents the state in which the customer has finished the usage of an E-Scooter.

- Check-in E-Scooter is performed by Customer
- Check-in E-Scooter includes Pay the service

Description

Pre-Condition

Customer has already checked out an E-Scooter.

Post-Condition

Customer has successfully terminated the ride and parked the Scooter.

Goal

Make Customer be able to return/park the E-Scooter to an appropriate location within the drive area limits.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

Normal

Scenarios

Basic Flow

1. Display Application Start screen
2. Set Status "Ride Finished"
3. Get Status "Ride Finished"
4. Sent Notification to the user, so it is sure that she/he finished the ride and must pay for it
5. Get Notification to confirm that the user has finished the ride and must pay for it

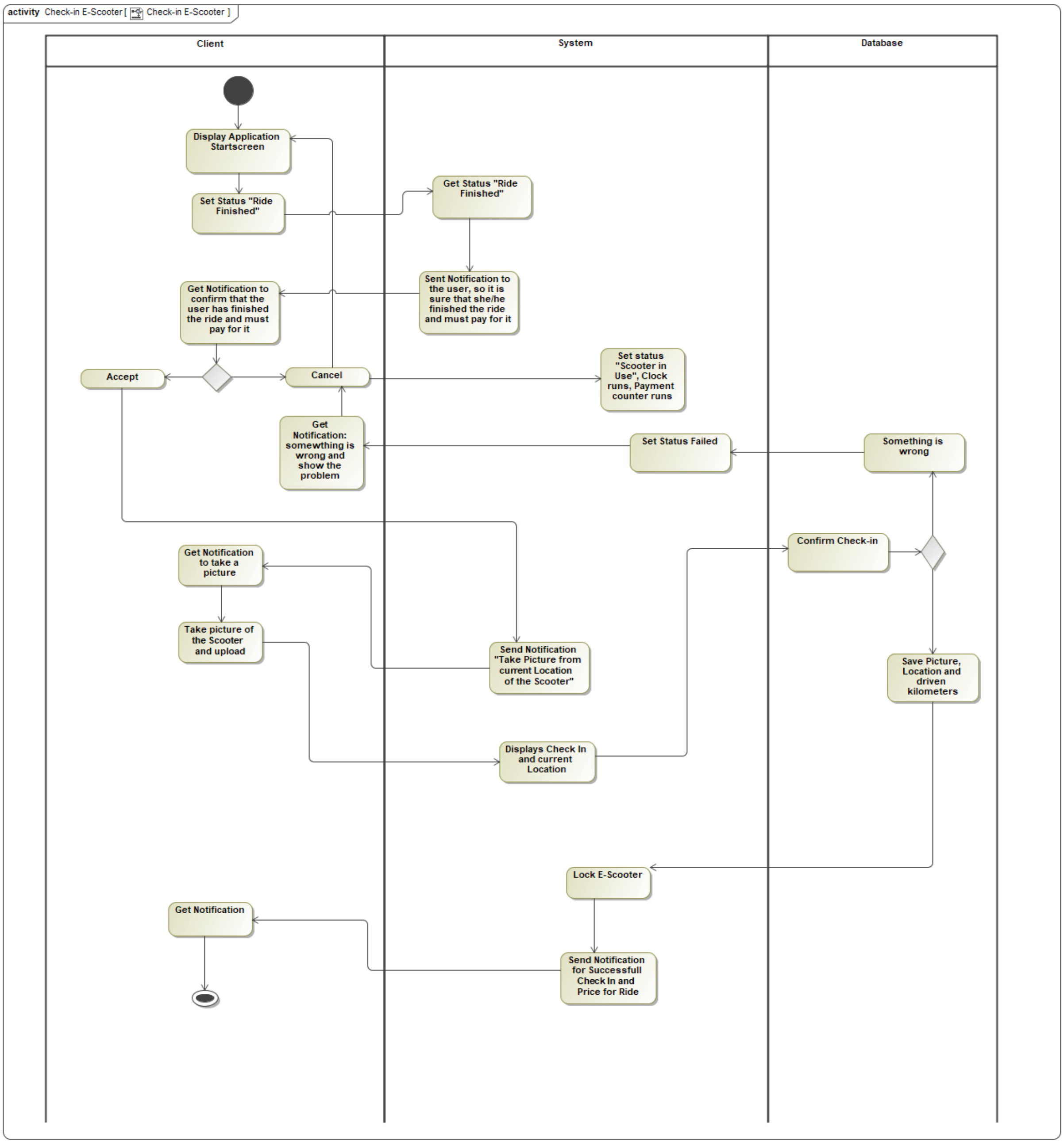
activity Check-in E-Scooter [Check-in E-Scooter]
Client
System
Database
Display Application Startscreen
Set Status "Ride Finished"
Get Status "Ride Finished"
Sent Notification to the user, so it is sure that she/he finished the ride and must pay for it
Get Notification to confirm that the user has finished the ride and must pay for it
Accept
Cancel
Set status "Scooter in Use", Clock runs, Payment counter runs
Get Notification: somewthing is wrong and show the problem
Set Status Failed
Something is wrong
Confirm Check-in
Get Notification to take a picture
Take picture of the Scooter and upload
Send Notification "Take Picture from current Location of the Scooter"
Save Picture, Location and driven kilometers
Displays Check In and current Location
Lock E-Scooter
Get Notification
Send Notification for Successfull Check In and Price for Ride

Figure 6. Check-in E-Scooter

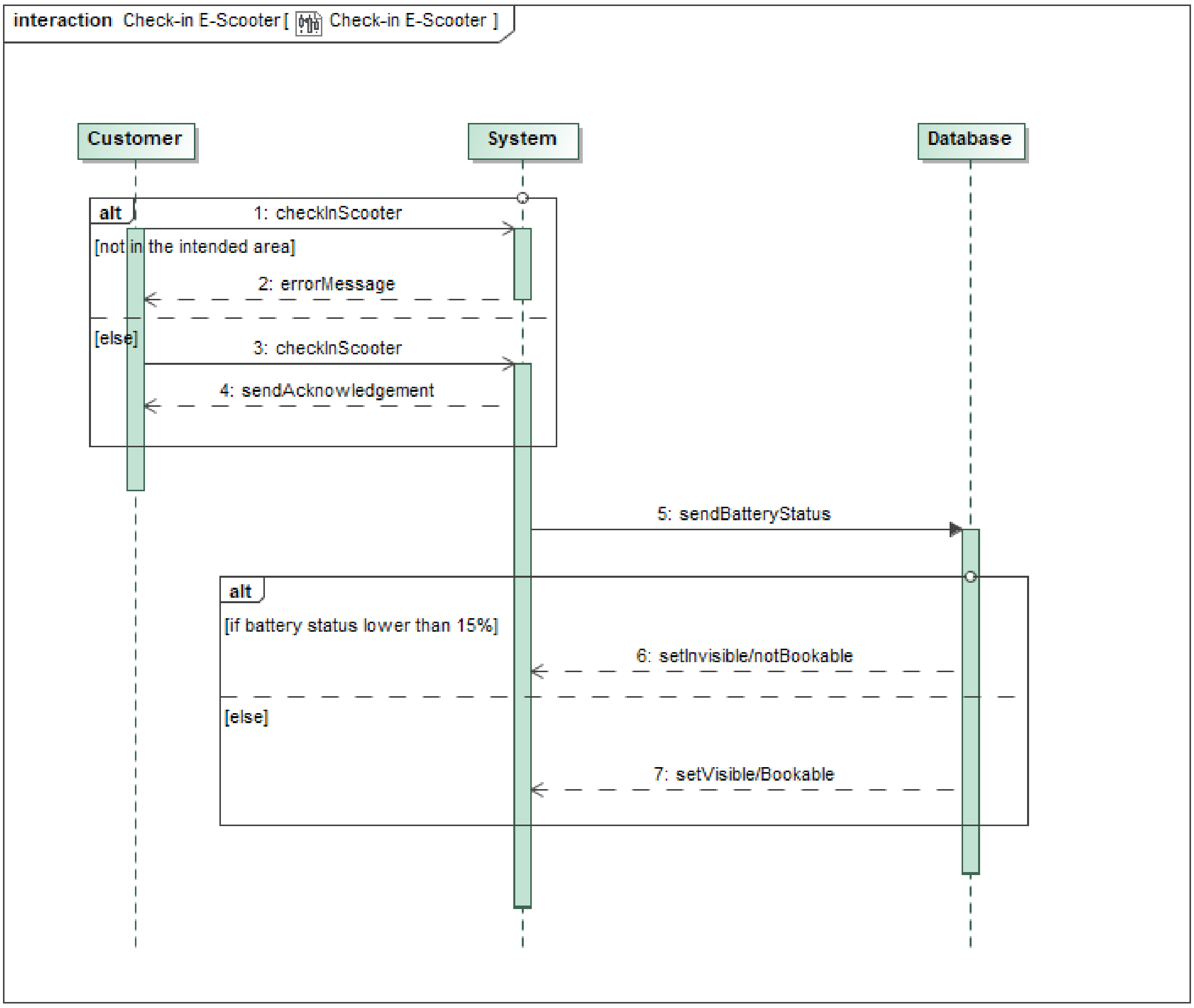

Figure 7. Check-in E-Scooter

10 UseCase Choose subscription plan

There are three membership tiers customers can subscribe: Gold, Silver and Bronze.

- Choose subscription plan includes Pay the service
- Choose subscription plan includes Pay via subscription plan
- Choose subscription plan is included by Subscribe to the service
- Choose subscription plan is included by Subscribe to the service

Extension Points

- Change Subscription Plan: Choose subscription plan

Description

Pre-Condition

Customer opened the app and is logged in.

Post-Condition

Customer changed the subscription plan successfully.

Goal

Give customers the option to choose a subscription plan which fits their needs.

Complexity

Average Complexity

Assumption

Customer did not choose a subscription yet.

Component Complexity

Low Complexity

Priority

High

Scenarios

Basic Flow

1. Customer opens the App and logs in.
2. Customer navigates through the main menu dropdown and finds the "Account Management".
3. Customer clicks on the current subscription status.
4. Customer chooses a plan according to her/his needs.

11 UseCase Change Subscription plan

Customers can change their membership type whenever they want. The new subscription then starts in the following month.

- Change Subscription plan extends Choose subscription plan

Description

Pre-Condition

Customer has logged in.

Post-Condition

Customer has successfully changed the subscription plan.

Goal

Give customers the option to adjust their subscription plan.

Complexity

Average Complexity

Assumption

Customer has already subscribed to the monthly E-Scooter service plan.

Component Complexity

Average Complexity

Priority

Normal

Scenarios

Basic Flow

1. Customer logs in.
2. Customer navigates to the "Account Management".
3. Customer changes her/his subscription plan.
4. UI refreshes.

12 UseCase Log-in

Registered customers must log-in to their account in the App in order to rent an E-Scooter.

- Log-in is performed by Customer
- Log-in is performed by Customer
- Log-in is performed by Customer
- Log-in includes Verify User Credentials

Extension Points

- Log-out: Log-in
- Login Errors: Log-in

Description

Pre-Condition

Customer has been successfully registered and has confirmed the Email address.

Post-Condition

Customer is logged in.

Goal

Customer logs in to ride an E-Scooter.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Enter Email Address
2. Enter Password

3. Get Information
4. Validate Data

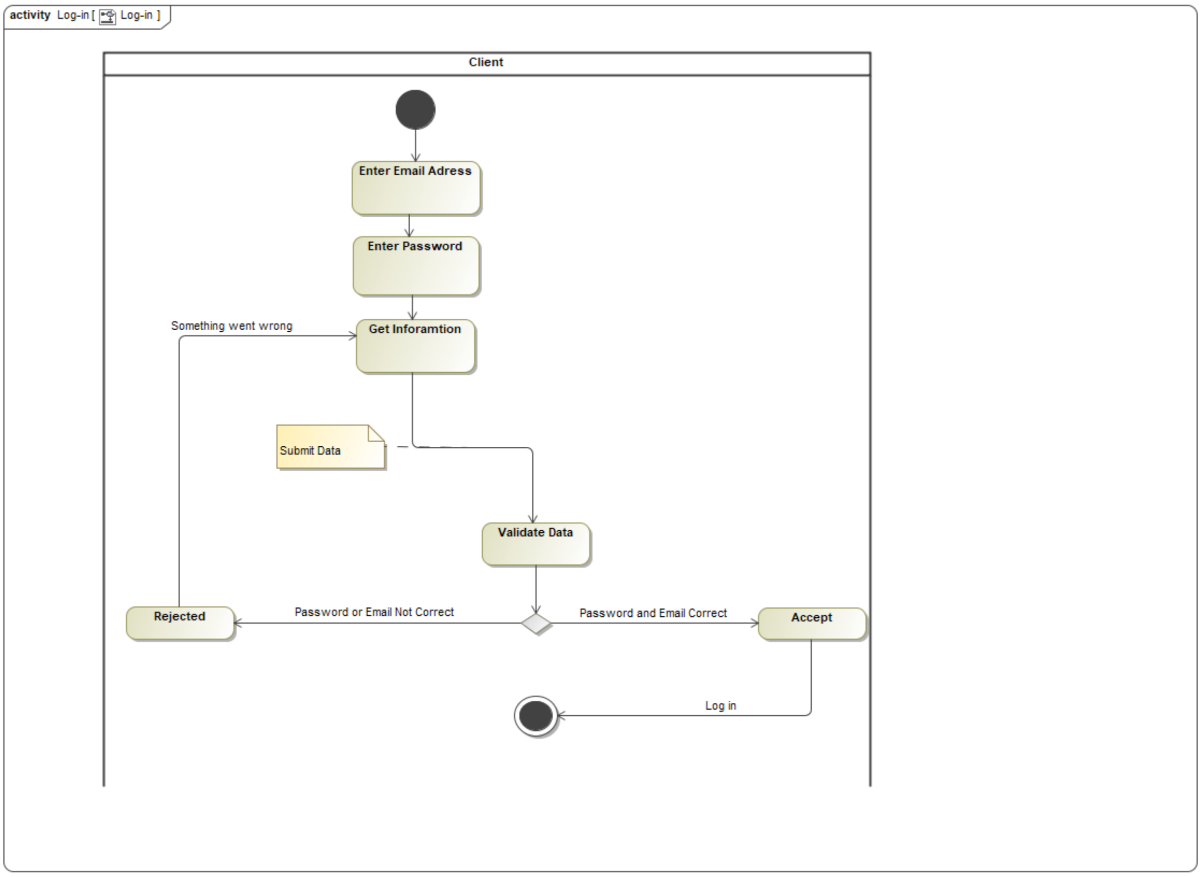

Figure 8. Log-in

13 UseCase Display Log-in Errors

When customers enter a wrong username or password, they get presented a log-in error.

- Display Log-in Errors extends Log-in

Description

Pre-Condition

Customer has been successfully registered.

Post-Condition

Customer is being presented with an alert warning her/him about the login errors.

Goal

When an incorrect password or username is entered the user should be presented with log in errors.

Complexity

Average Complexity

Component Complexity

Low Complexity

Priority

Above Normal

14 UseCase Verify User Credentials

The entered log-in data are compared and verified with those in the database.

- Verify User Credentials is included by Log-in

Description

Pre-Condition

Customer has successfully registered her/his account.

Post-Condition

Customer has entered the log-in credentials which have been verified.

Goal

Verify the log-in credentials entered by the customer.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Customer enters the username and password.
2. Customer clicks log-in.
3. Log-in credentials are being compared with the log-in data in the database.

15 UseCase Log-out

When the customer has finished the ride and does not need the E-Scooter service any further, she/he can log-out of her/his account.

- Log-out is performed by Customer
- Log-out extends Log-in
- Log-out extends Log-in

Description

Pre-Condition

Customer is logged in.

Post-Condition

Customer is logged out.

Goal

Customer should be able to log out from the App.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Open Menu
2. Tap "Logout" Button

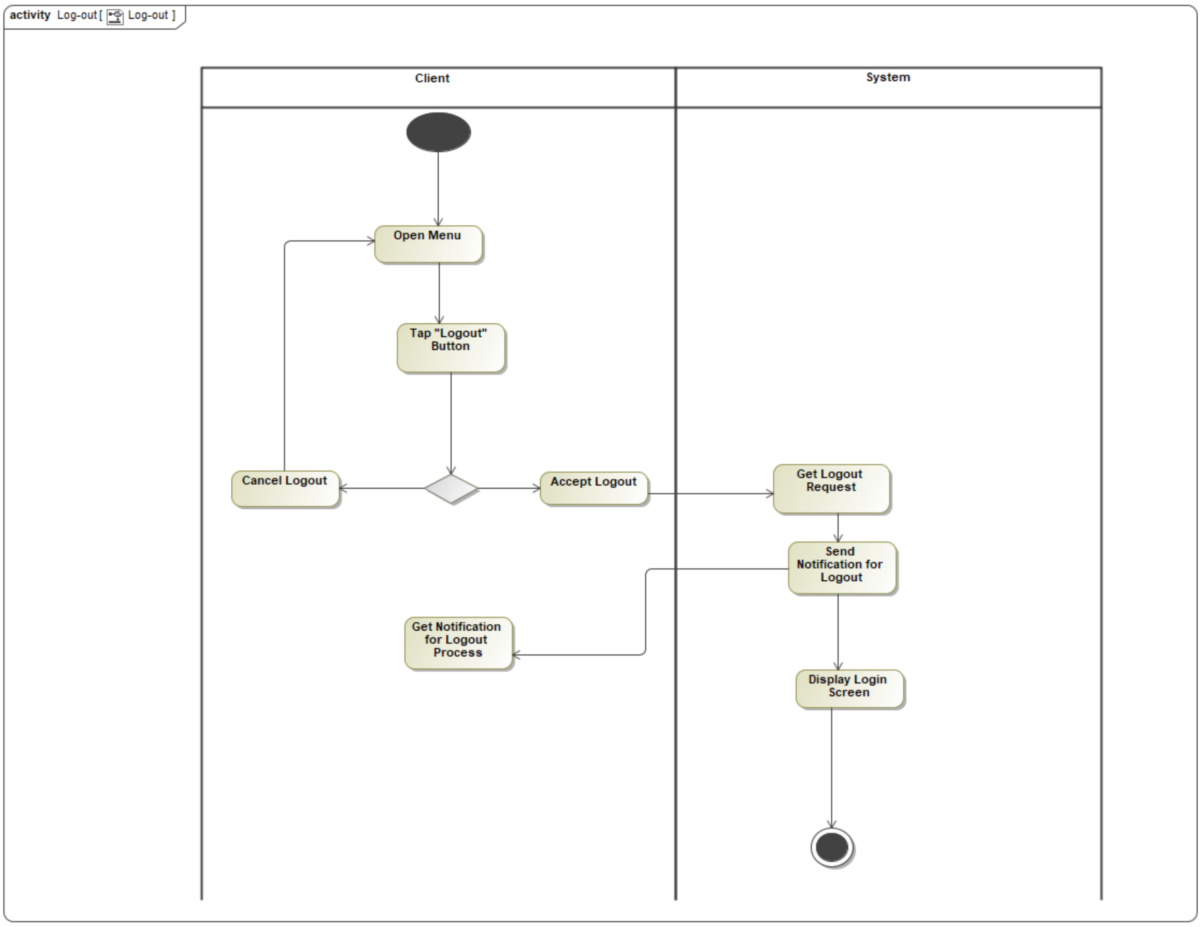

Figure 9. Log-out

16 UseCase Scan QR Code

Customers must scan the QR Code of an E-Scooter before the ride can start.

- Scan QR Code is included by Check-out E-Scooter

Description

Pre-Condition

Customer is logged in.

Post-Condition

QR-Code has been scanned.

Goal

Scan the Scooter's QR-Code in the Check-out process.

Complexity

Average Complexity

Assumption

Customer's phone has a camera.

Implementation Issue

No camera on the Customer's device.

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Customer opens the App.
2. Customer gives permission to the App to use the camera.
3. Camera opens, and the QR-Code is being tracked by a red box.
4. QR-Code is scanned.
5. Availability attribute for the scanned Scooter is changed in the database.

17 UseCase Manage Account

Customers can change account details like bank credentials, E-Mail address, address, etc.

- Manage Account is performed by Customer
- Manage Account is performed by Customer

Extension Points

- Change Bank Credentials: Manage Account
- Check usage stats: Manage Account
- Delete Account: Manage Account

Description

Pre-Condition

Customer is logged in.

Post-Condition

Customer has saved changes to her/his account.

Goal

Customers can change their Personal details.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Start Application
2. Open Account Management
3. Edit Account

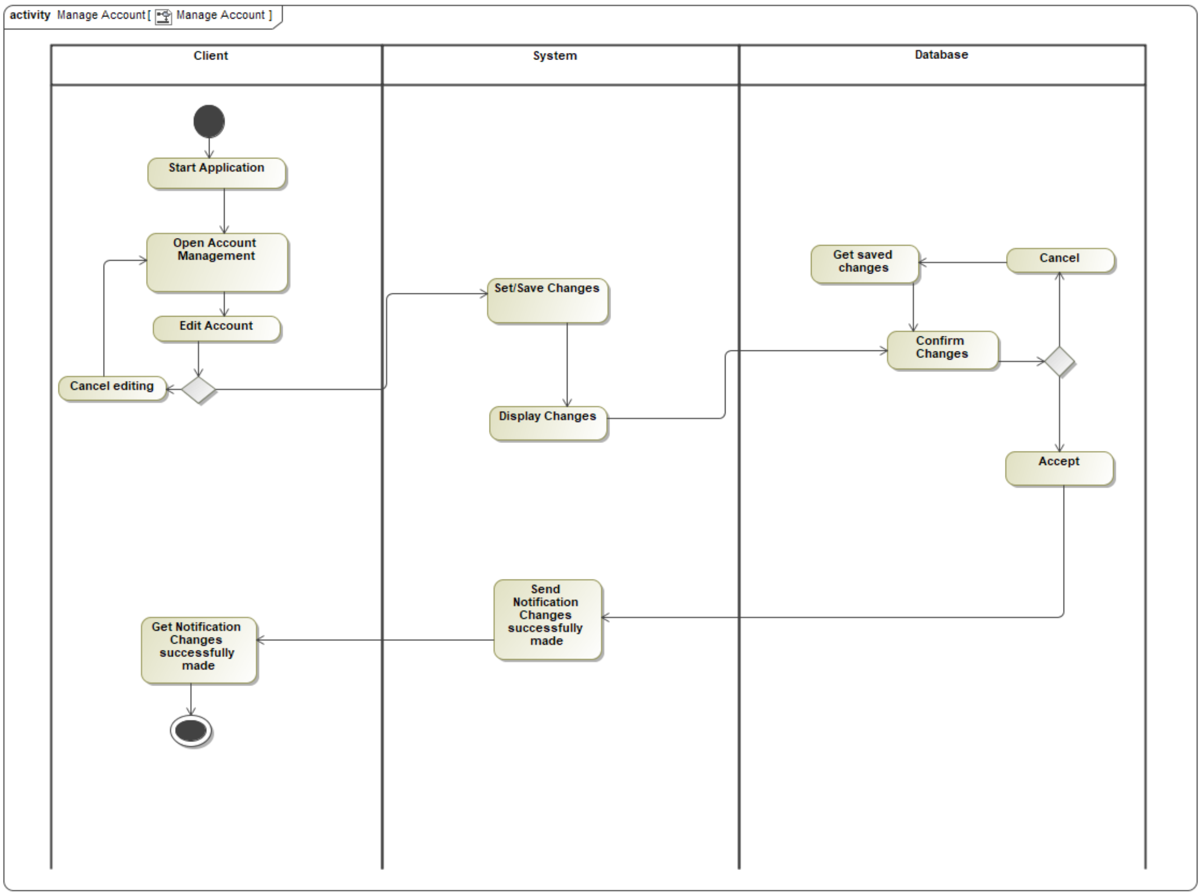

Figure 10. Manage Account

18 UseCase Delete Account

A customer can delete her/ his account, if she/ he does not want to use the service anymore.

Admins also can delete a customer's account, if she/ he violates the terms and conditions.

- Delete Account is performed by Customer
- Delete Account extends Manage Account

Description

Pre-Condition

Customer is logged in.

Post-Condition

Customer no longer possesses an account in the E-Scooter App.

Goal

Removes the customer's account from the database.

Complexity

Low Complexity

Assumption

Customer has created an account.

Component Complexity

Low Complexity

Priority

Above Normal

Scenarios

Basic Flow

1. Customer opens the App.
2. Customer navigates to the "Account Management".
3. Customer deletes the account.
4. Customer's information is removed from the App's database.

19 UseCase Find an E-Scooter on the map

Displays all nearby available E-Scooters on the map using a built-in map and GPS locator.

- Find an E-Scooter on the map is performed by Customer
- Find an E-Scooter on the map is performed by Customer
- Find an E-Scooter on the map is performed by Maintenance man
- Find an E-Scooter on the map includes Get GPS Location of the Customer
- Find an E-Scooter on the map includes Get GPS Location of the E-Scooter

Extension Points

- Download maps : Find an E-Scooter on the map
- Get GPS Location of the customer : Find an E-Scooter on the map
- Reserve an E-Scooter : Find an E-Scooter on the map

Description

Pre-Condition

App installed and Customer has made an account. GPS-Location has been allowed to share with the App.

Post-Condition

View of the map with nearby available Scooters on it.

Goal

Show the available scooters on the map for the customer's potential use.

Complexity

Average Complexity

Outstanding Issue

No available Scooters in the area.

Component Complexity

High Complexity

Priority

Above Normal

Scenarios

Basic Flow

1. Customer opens the App.
2. Customer gets presented with the map.

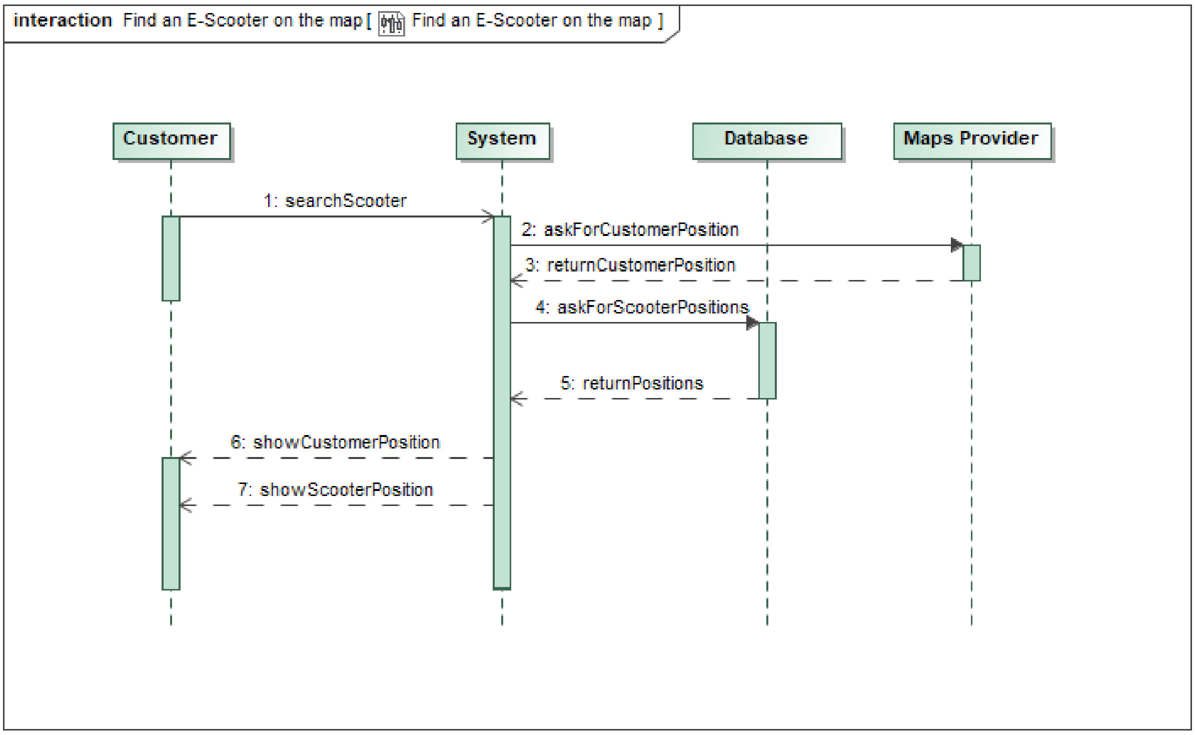
interaction Find an E-Scooter on the map [Find an E-Scooter on the map]
Customer
System
Database
Maps Provider
1: searchScooter
2: askForCustomerPosition
3: returnCustomerPosition
4: askForScooterPositions
5: returnPositions
6: showCustomerPosition
7: showScooterPosition

Figure 11. Find an E-Scooter on the map

20 UseCase Get GPS Location of the E-Scooter

Displays all nearby available E-Scooters on the map using a built-in map and GPS locator.

- Get GPS Location of the E-Scooter is performed by Administrator
- Get GPS Location of the E-Scooter is performed by Maintenance man
- Get GPS Location of the E-Scooter is performed by Maintenance man
- Get GPS Location of the E-Scooter is performed by Maps Service Provider
- Get GPS Location of the E-Scooter is included by Find an E-Scooter on the map

Description

Pre-Condition

Customer has opened the Application and turned on her/his GPS-Location service.

Post-Condition

Map updates and user gets the view of all nearby available Scooters.

Complexity

Average Complexity

Component Complexity

Low Complexity

Priority

Normal

Scenarios

Basic Flow

1. Customer opens the E-Scooter Service App.

2. Customer clicks "Allow sharing GPS-Location to this App".

3. Customer gets the view of all available Scooter nearby (relative to his/her position).

3.1 Maintenance man/Servicer gets presented with faulty or battery deficient Scooters.

Alternative Flow

1. User does not accept to share his GPS coordinates.

2. User gets presented with larger view of the City center that the entered as her/his city.

21 UseCase Get GPS Location of the Customer

Displays the customer's position on the map using GPS so she/ he is able to see their relative position to the nearby available E-Scooters.

- Get GPS Location of the Customer is performed by Maps Service Provider
- Get GPS Location of the Customer is included by Find an E-Scooter on the map
- Get GPS Location of the Customer extends Find an E-Scooter on the map

Description

Pre-Condition

Customer is logged in.

Post-Condition

Map updates showing the customer's position.

Goal

Map needs the current position of the customer in order to find E-Scooters on the map relative to her/his position.

Complexity

Average Complexity

Assumption

GPS Service is turned on the customer's end.

Component Complexity

Low Complexity

Priority

High

Scenarios

Basic Flow

1. Customer turns on the GPS Location service on the phone.

2. Map updates showing the customer's position.

Alternative Flow

1. GPS Location on the Customer's end is not turned on

2. Customer's in-app map gets updated showing the city center (taken from the Customer's account info)

22 UseCase Download Offline Maps

Customers can download an offline map of their area.

- Download Offline Maps is performed by Maps Service Provider
- Download Offline Maps is performed by Maps Service Provider
- Download Offline Maps extends Find an E-Scooter on the map
- Download Offline Maps extends Find an E-Scooter on the map

Description

Pre-Condition

Customer is logged in.

Post-Condition

Maps of the customer's surroundings in a n perimeter have been downloaded.

Goal

The customer should be able to download the maps from the Map Service Provides current surroundings in order to use it for an offline use.

Complexity

Average Complexity

Assumption

Customer has an Internet connection for the maps to be downloaded.

Component Complexity

Low Complexity

Priority

Low

23 UseCase Subscribe to the service

There are three membership tiers customers can subscribe: Gold, Silver and Bronze. Customers without a membership get their prices calculated per km or min/ h while the customers with a membership pay more affordable prices on a monthly/ yearly basis.

- Subscribe to the service is performed by Customer
- Subscribe to the service is performed by Customer
- Subscribe to the service includes Choose subscription plan
- Subscribe to the service includes Choose subscription plan
- Subscribe to the service extends Register

Extension Points

- Cancel Subscription: Subscribe to the service

Description

Pre-Condition

Customer is registered.

Customer is logged in.

Post-Condition

Customer chooses between many subscriptions plans available.

Goal

The option to subscribe and pay monthly instead after every ride should be allowed to the customer.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

Above Normal

24 UseCase Cancel Subscription

If a Customer knows she/ he will not use the service for a certain period of time, she/ he can cancel the membership without deleting the whole account.

- Cancel Subscription extends Subscribe to the service

Description

Pre-Condition

Customer has opened his/her app and is logged in.

Post-Condition

Customer's subscription record is removed from the database as the paying is switched to the wallet payment.

Goal

Delete a record in the database containing the customer's subscription details when the customer wants to cancel the subscription.

Complexity

Average Complexity

Component Complexity

Low Complexity

Priority

Above Normal

Scenarios

Basic Flow

1. Customer opens the app and is logged in.
2. Customer has navigated to the "Account Management/Change Account Details".
3. Customer has clicked his/her "Subscription button".
4. Customer clicks, confirms and cancels his/her subscription.

25 UseCase Pay the service

There are multiple ways for customers to pay for the service: Membership Subscription and Wallet.

- Pay the service is included by Check-in E-Scooter
- Pay the service is included by Choose subscription plan

Extension Points

- Subscription Plan Payment: Pay the service
- Wallet Payment: Pay the service

Description

Pre-Condition

Customer has checked-in the chosen E-Scooter.

Post-Condition

A payment transaction to the App's account has been made.

Goal

Make transactions for the given service between the Customer and the E-Scooter Service.

Complexity

Average Complexity

Assumption

Customer has at least n Euros on her/his in-app wallet.

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Check-out an E-scooter
2. Check-in an E-Scooter
3. Confirm Check-in
4. Get Confirmation of Check-in
5. Display Drive Stats and Price
6. Request Info about Subscription
7. Retrieve Information about Subscription

Alternative Flow

7.1. Alternative Condition

7.1.1. Choose Payment method

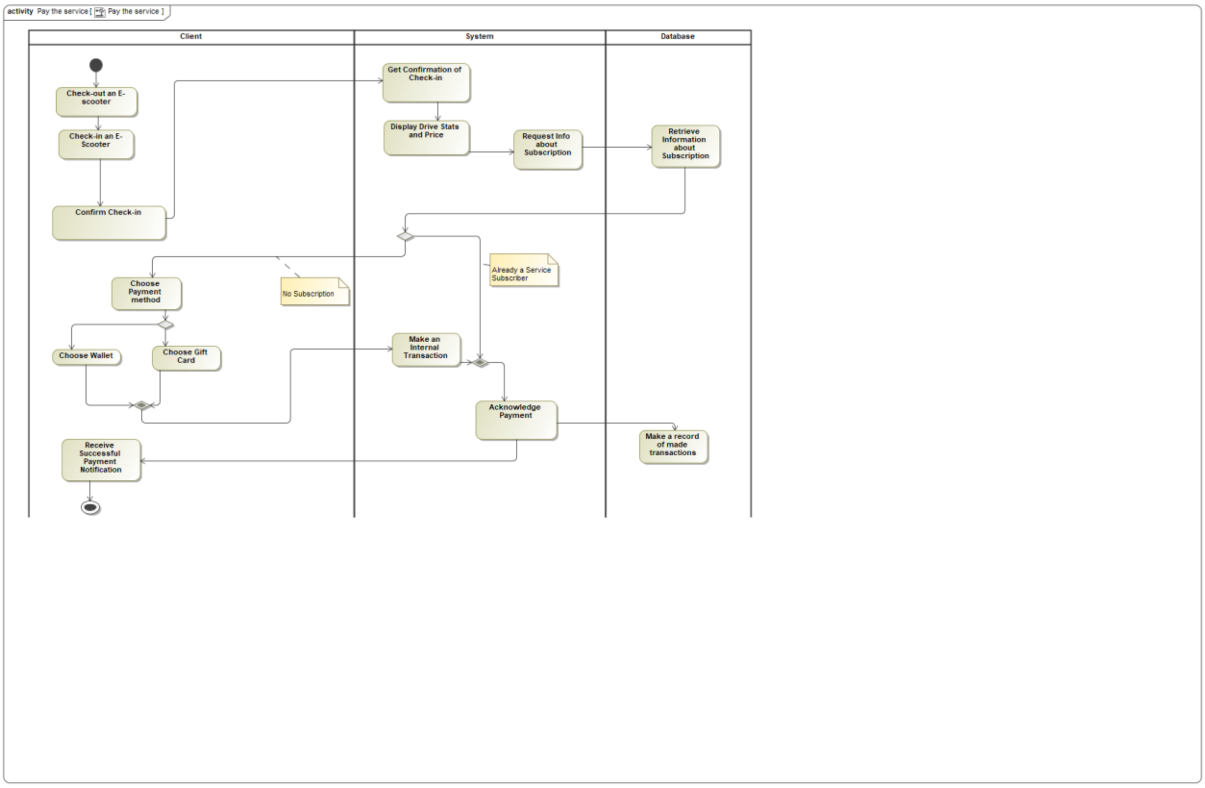
activity Pay the service [Pay the service]
Client
System
Database
Check-out an E-scooter
Check-in an E-Scooter
Confirm Check-in
Get Confirmation of Check-in
Display Drive Stats and Price
Request Info about Subscription
Retrieve Information about Subscription
Already a Service Subscriber
No Subscription
Choose Payment method
Choose Wallet
Choose Gift Card
Make an Internal Transaction
Acknowledge Payment
Make a record of made transactions
Receive Successful Payment Notification

Figure 12. Pay the service

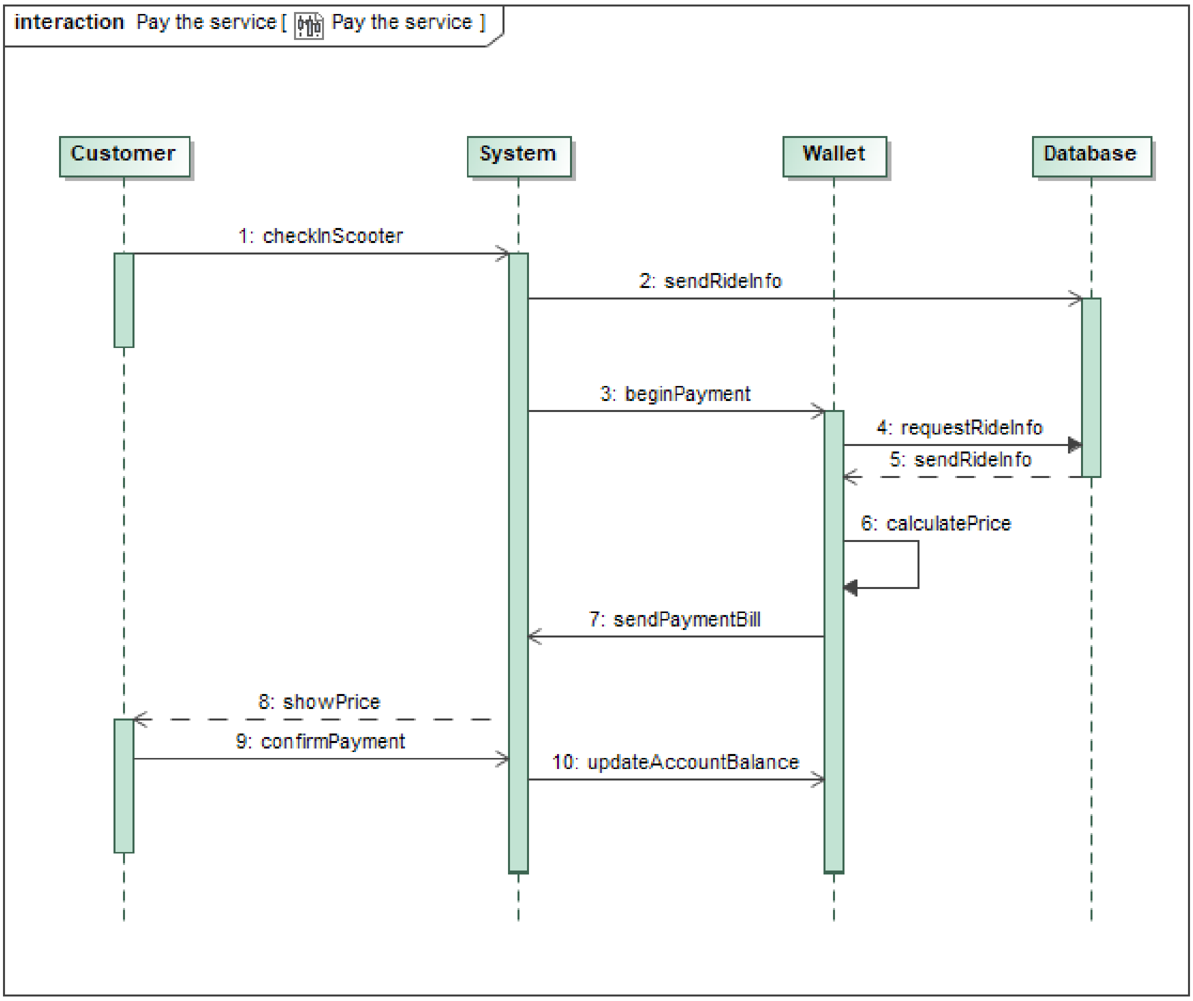

Figure 13. Pay the service

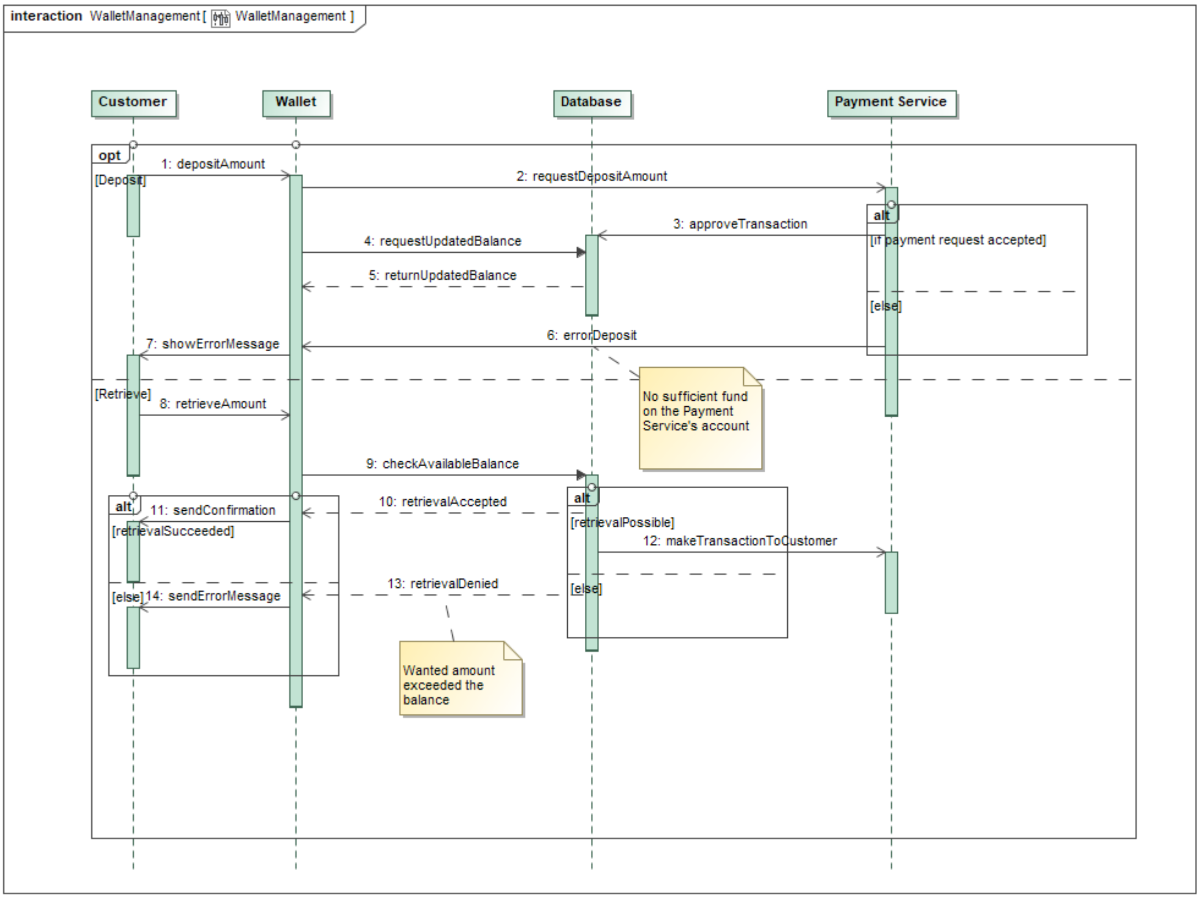

Figure 14. Wallet Management

26 UseCase Pay via subscription plan

Customers are able to choose to pay on a monthly or yearly basis by choosing among the three membership tiers: Gold, Silver and Bronze. The payment is made via a deposited credit/ debit card.

- Pay via subscription plan is performed by Payment Service
- Pay via subscription plan is included by Choose subscription plan
- Pay via subscription plan extends Pay the service

Description

Pre-Condition

Customer has checked-in her/his E-Scooter.

Post-Condition

Customer gets notified with the amount of km driven and the rest of her/his monthly subscription plan.

Goal

Customer has an option to pay a monthly fixed fee for a certain amount of km driven.

Complexity

Average Complexity

Assumption

Customer is subscribed to the E-Scooter service.

Customer has already entered the credit/debit card credentials.

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Customer checks-in her/his E-Scooter.
2. System checks if the customer is a subscriber of the E-Scooter service.
3. Customer receives a notification of her/his ride details.

27 UseCase Pay via Wallet

The Wallet is an in-App virtual money storage system designed for the convenient use by the customer. It is needed for managing the customer's money balance for the ride.

- Pay via Wallet is performed by Payment Service
- Pay via Wallet extends Pay the service
- Pay via Wallet extends Pay the service

Description

Pre-Condition

Customer is logged in.

Customer has entered bank information.

Customer has deposited a minimum of n Euros to his/her in-App wallet.

Post-Condition

Customer's rides are being charged via Wallet.

Goal

Customer should be able to deposit and then later retrieve the money in order to make virtual wallet from which the rides will be charged.

Complexity

Average Complexity

Assumption

At least n Euros in the Wallet.

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

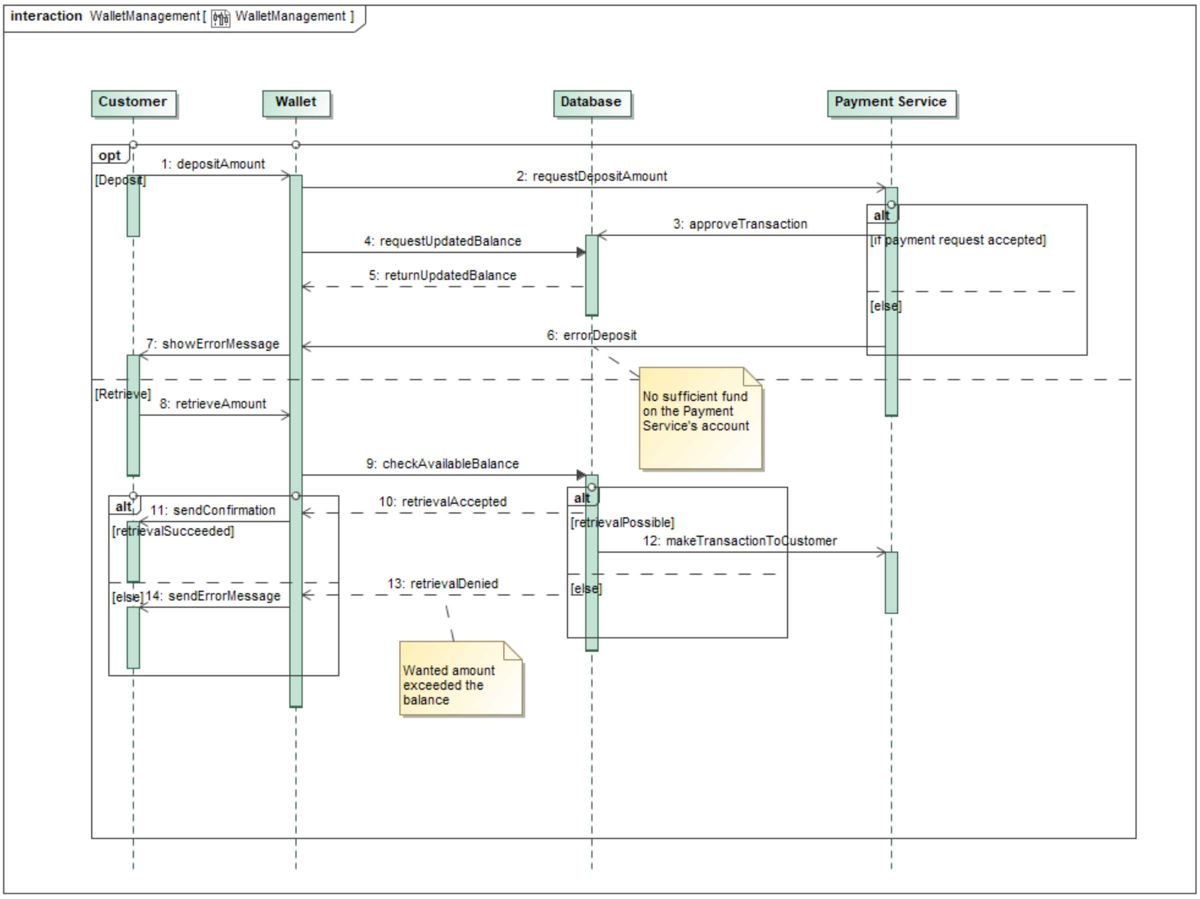

Figure 15. Basic flow of Pay via Wallet

28 UseCase Show E-Scooter Battery Status

Shows Admins/ Customers the battery status of each E-Scooter. On the hand, this helps the customers to decide, if the Scooter has enough battery for the ride, on the other hand, the Admins see the Scooters, which need to be recharged.

- Show E-Scooter Battery Status is performed by Administrator
- Show E-Scooter Battery Status is performed by Maintenance man
- Show E-Scooter Battery Status is included by Check-out E-Scooter

Description

Goal

Show battery status of the E-Scooters on the map to any user type.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

High

29 UseCase Reserve an E-Scooter

Customers have the possibility to pre-book an E-Scooter for later or for another day.

- Reserve an E-Scooter is performed by Customer
- Reserve an E-Scooter extends Find an E-Scooter on the map

Description

Pre-Condition

Customer is logged in.

Customer has found an E-Scooter that he/she wants to reserve.

Post-Condition

A chosen E-Scooter has been reserved.

Goal

Customer should be able to make a reservation for an E-Scooter found on the map.

Complexity

Average Complexity

Assumption

Scooter to be reserved is available.

Component Complexity

Average Complexity

Priority

Normal

Scenarios

Basic Flow

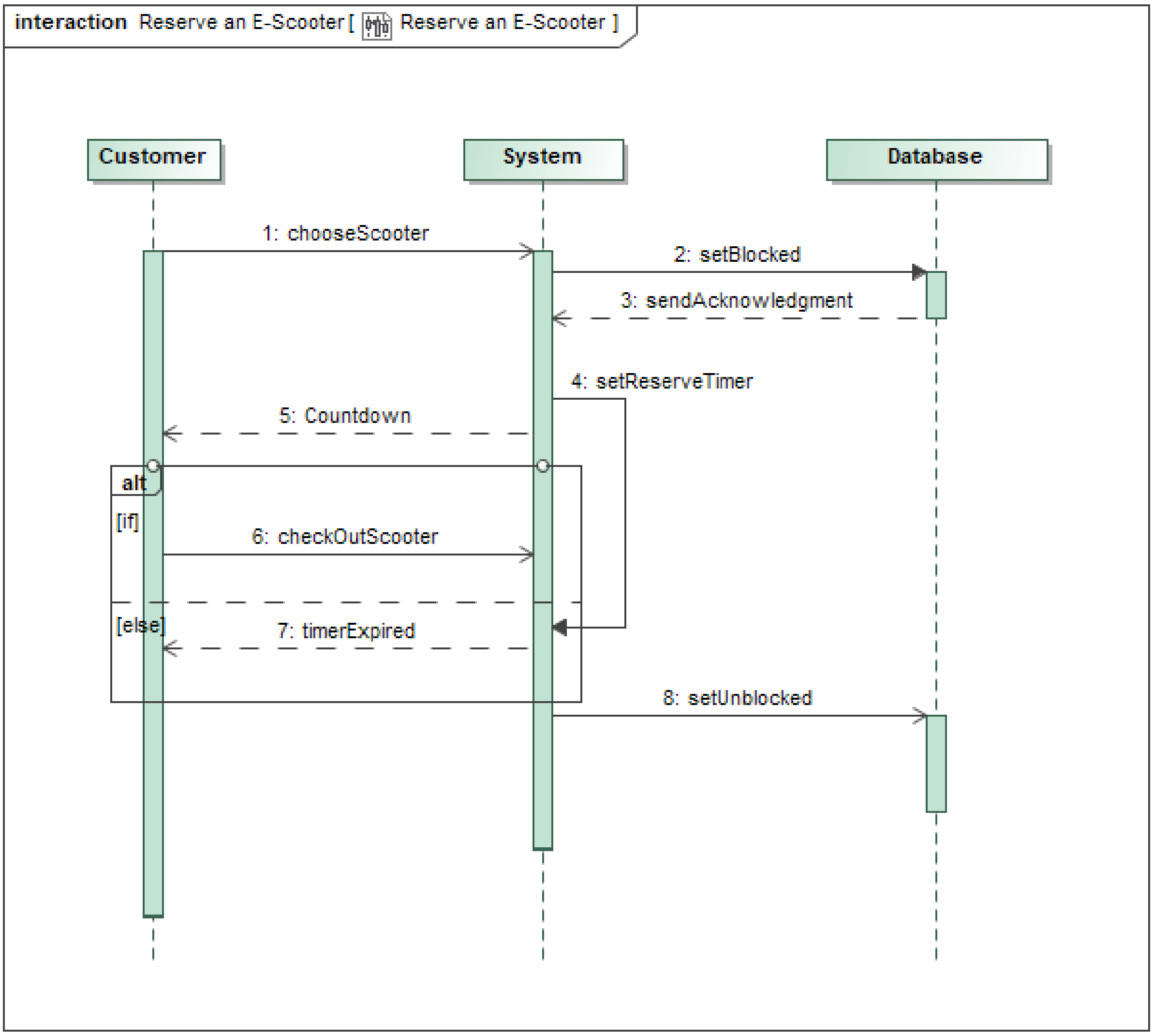

Figure 16. Reserve an E-Scooter

30 UseCase Report Problems with Scooter

Customers can send information about specific problems with an E-Scooter.

- Report Problems with Scooter is performed by Customer
- Report Problems with Scooter is performed by Customer

Description

Pre-Condition

Customer/Maintenance man is logged in.

Post-Condition

A report has been filled detailing a problem with(-out) pictures attached to it.

Goal

Make reports about the potential and current problems with an E-Scooter.

Complexity

Average Complexity

Assumption

There is a problem with an E-Scooter

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Open the App and find "Report a Problem" option in the Menu
2. File A Basic Report and/or attach a Photo
3. Get the Problem Report
4. Store the Reports
5. Analyze Reports

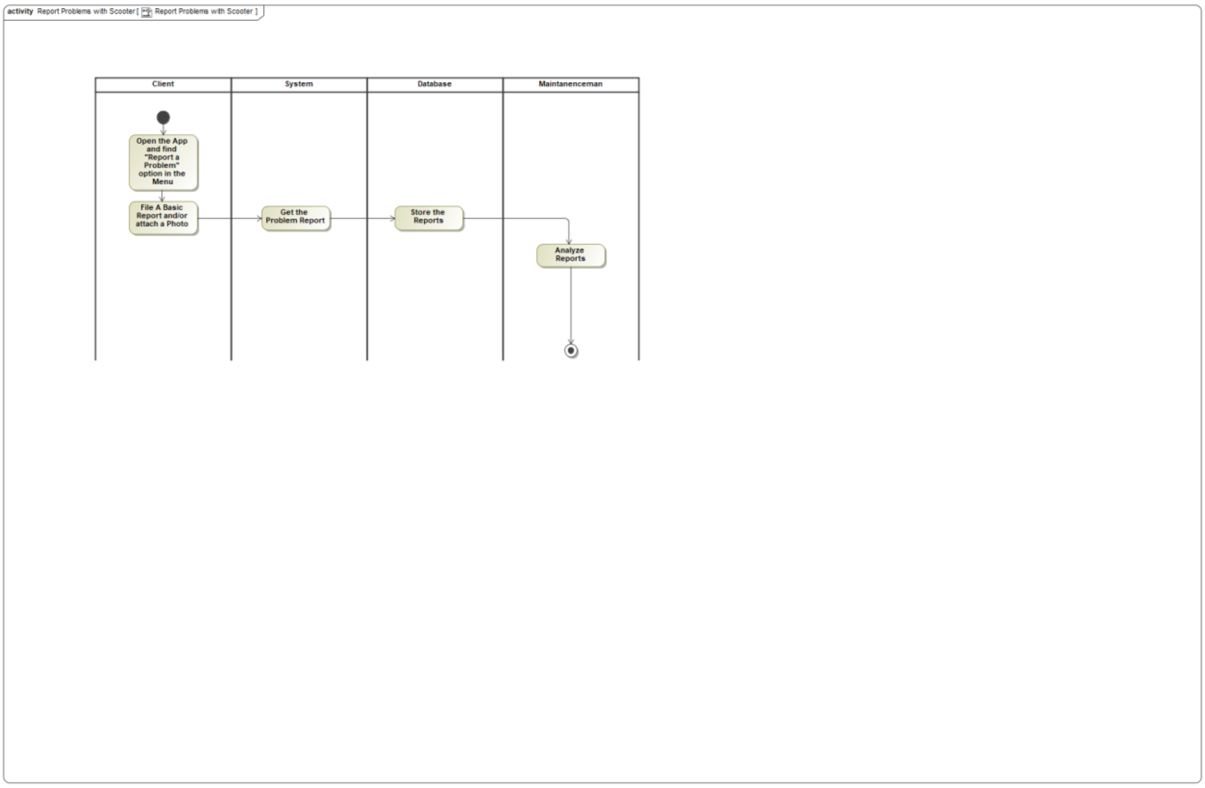
activity Report Problems with Scooter [Report Problems with Scooter]
Client
System
Database
Maintanenceman
Open the App and find "Report a Problem" option in the Menu
File A Basic Report and/or attach a Photo
Get the Problem Report
Store the Reports
Analyze Reports

Figure 17. Report Problems with Scooter

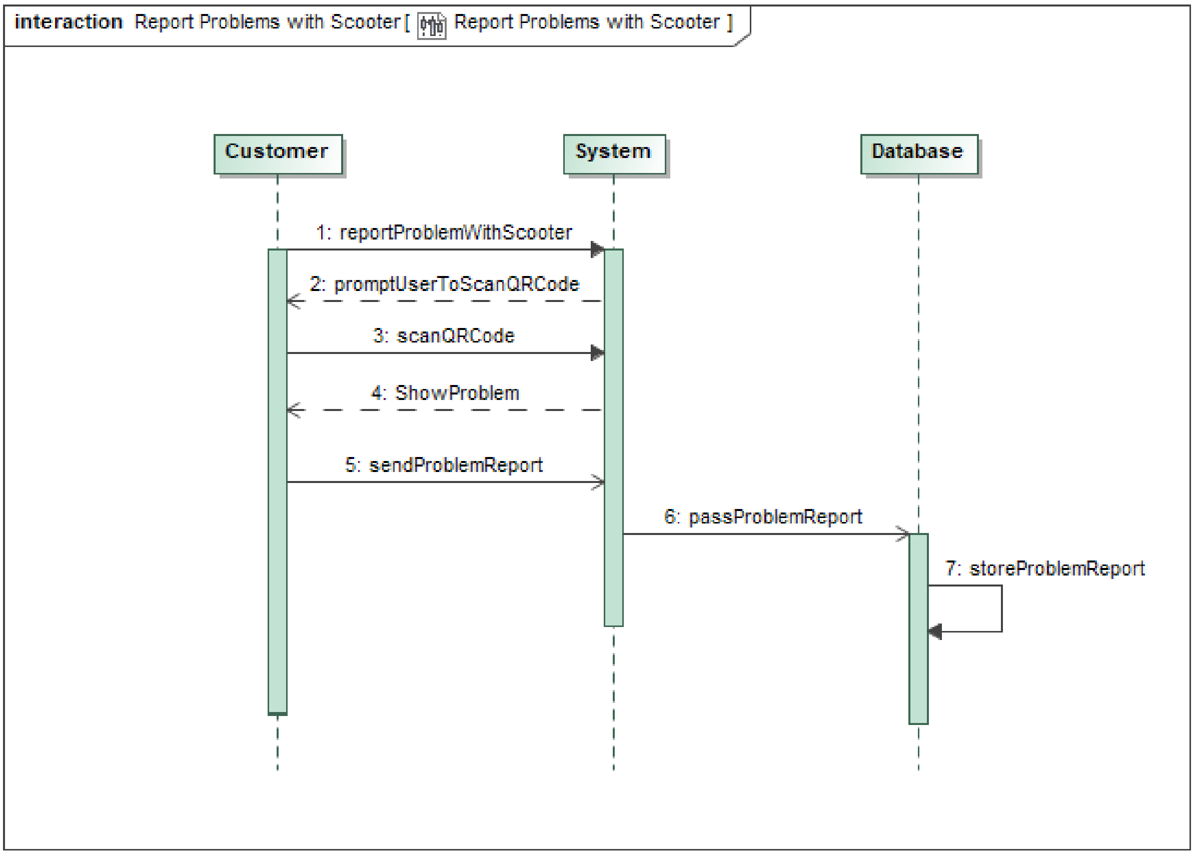
interaction Report Problems with Scooter [Report Problems with Scooter]
Customer
System
Database
1: reportProblemWithScooter
2: promptUserToScanQRCode
3: scanQRCode
4: ShowProblem
5: sendProblemReport
6: passProblemReport
7: storeProblemReport

Figure 18. Report Problems with Scooter

31 UseCase Give Feedback

If customers have any suggestions to improve the service, they can give a feedback in the App.

- Give Feedback is performed by Customer
- Give Feedback is performed by Customer
- Give Feedback includes Analyze Customer Feedback

Description

Pre-Condition

Customer is logged in.

Post-Condition

Customer has left a feedback on the current situation of the app in form of a comment.

Goal

Customer can leave a feedback in the app at any time.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

High

Scenarios

Basic Flow

1. Leave a Feedback on the App
2. Store Customer Feedback
3. Retrieve Stored Feedbacks
4. Analyze Customer Feedback

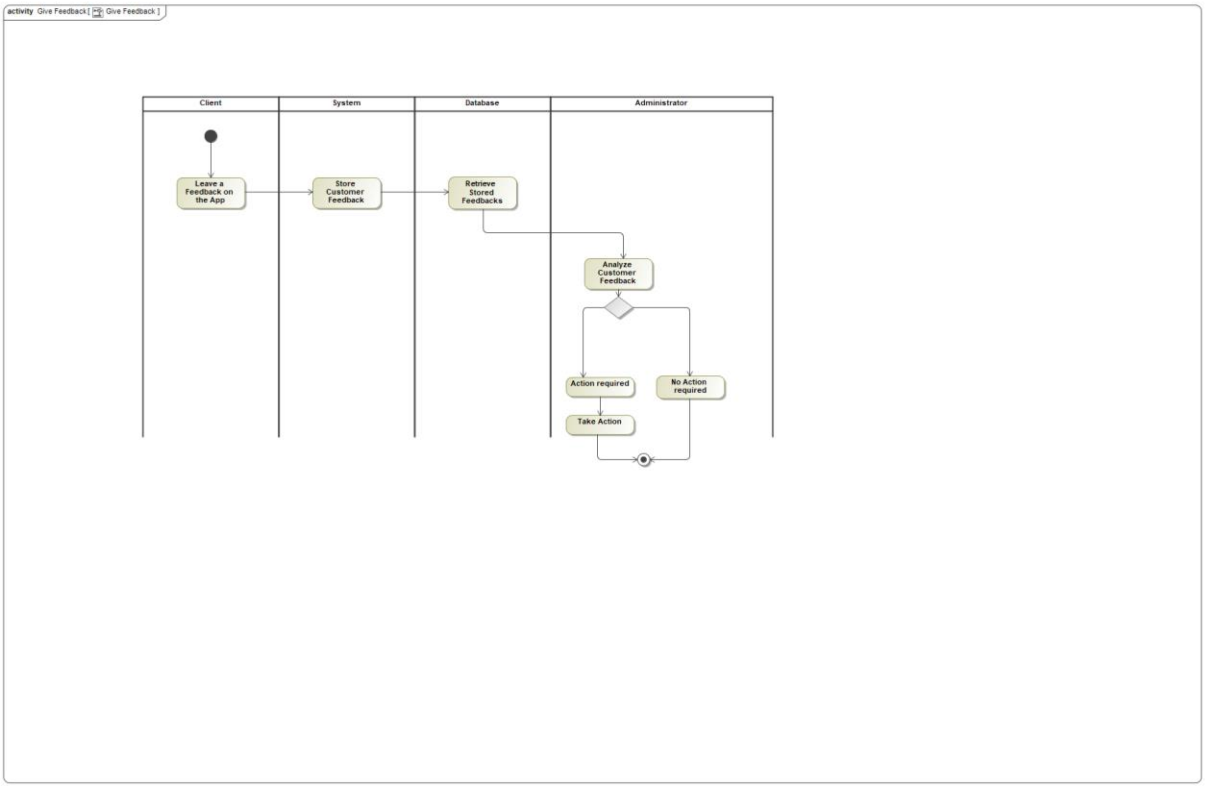
activity Give Feedback [Give Feedback]
Client
System
Database
Administrator
Leave a Feedback on the App
Store Customer Feedback
Retrieve Stored Feedbacks
Analyze Customer Feedback
Action required
No Action required
Take Action

Figure 19. Give Feedback

32 UseCase Ask for feedback

- Ask for feedback is performed by Administrator
- Ask for feedback extends Get Notifications

Description

Pre-Condition

Customer has opened the app.

Post-Condition

Customer gets presented with a panel containing the option to leave a feedback on the app/service.

Goal

Get feedback through a panel in the app UI containing a star scale and the possibility to add a comment for the Customer.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

Below Normal

33 UseCase Analyze Customer Feedback

By analyzing customer feedbacks, the company can improve the service and keep the customer satisfaction on a high level.

- Analyze Customer Feedback is performed by Administrator
- Analyze Customer Feedback is included by Give Feedback

Description

Pre-Condition

Admin has received feedbacks

Post-Condition

According to feedbacks with comments admin computes the result of the most wanted customer changes/adjustments to the service/app.

Goal

Analyze the current situation of the app.

Complexity

Low Complexity

Component Complexity

Average Complexity

Priority

Normal

34 UseCase Send warning

There is a well-defined "legal" area, where the E-Scooters can be driven and checked-in.

If the customer is leaving this "legal" area, she/ he is getting a push notification on her/ his device with a warning, that she/ he is leaving the allowed area and that the Scooter will be locked, if she/ he does not drive back to the "legal" area.

- Send warning extends Get Notifications

Description

Pre-Condition

Customer has checked-out her/his Scooter

Customer drove out of bounds.

Post-Condition

Customer gets a warning notification telling him/her that the Scooter will be locked if the Customer does not drive back into allowed limits.

Goal

Send a warning message telling the user that he/she drove out of the allowed limits or that the user is driving inappropriately.

Complexity

Average Complexity

Assumption

Driving area limit is well defined and possibly presented on the map.

Component Complexity

Average Complexity

Priority

Above Normal

35 UseCase Get Notifications

Customers get a push notification on their devices via the App.

- Get Notifications is performed by Customer

Extension Points

- Push notifications: Get Notifications
- Warnings: Get Notifications

Description

Pre-Condition

Customer is registered.

Post-Condition

Customer receives notifications from the App.

Goal

Customer receives push notifications/newsletter/warnings from the App.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

Below Normal

36 UseCase Send a push notification

Send push notifications to the customer's device via the App.

Customers get notifications about:

- Driving rules: all new customers are shown the basic driving rules before their first ride with an E-Scooter.

- Safety risks: customers are showed the safety risks of driving an E-Scooter.

- Legal information: if a customer tries to misuse the E-Scooter, she/ he gets presented with the legal constraints and risks.

- Newsletter: Customers receive an E-Mail with news about the App, special offers, etc.

- Send a push notification is performed by Administrator
- Send a push notification extends Get Notifications
- Send a push notification extends Get Notifications

Description

Goal

Customer should be able to receive push notifications.

Complexity

Average Complexity

Component Complexity

Average Complexity

Priority

Normal

37 UseCase Send an email confirmation

After a customer has entered all relevant registration information, she/ he gets an Email confirmation to complete the registration.

- Send an email confirmation is included by Register

Description

Pre-Condition

Customer has entered all relevant registration details.

Post-Condition

An Email confirmation of the registration has been sent to the customer's Email address.

Goal

Customer's Email address should be confirmed upon registration.

Complexity

Average Complexity

Assumption

Customer has entered a right Email address belonging to him/her.

Component Complexity

Average Complexity

Priority

Below Normal

E-scooter App

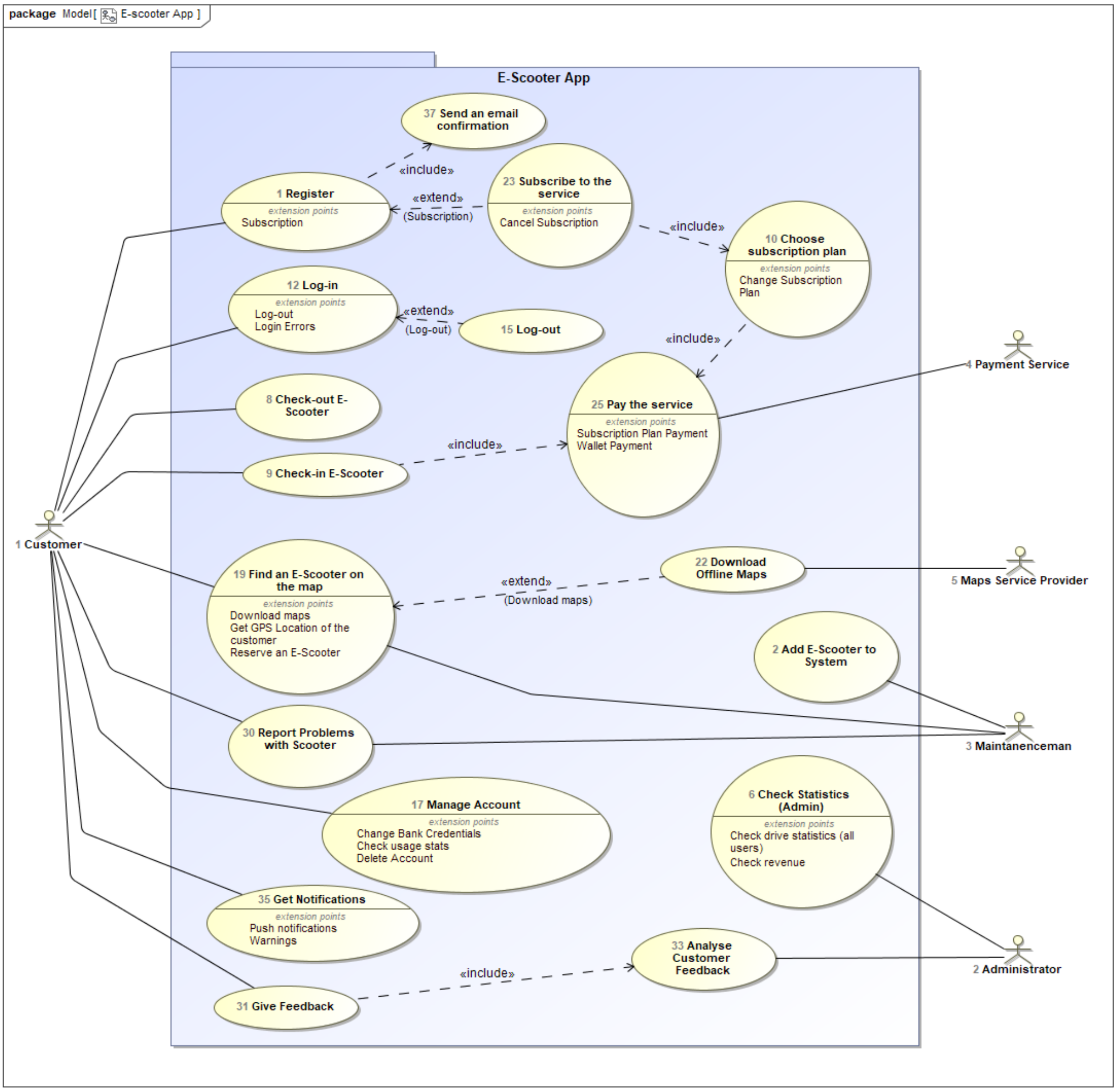

Figure 20. E-scooter App

Actor Administrator

- Administrator performs Analyze Customer Feedback
- Administrator performs Ask for feedback
- Administrator performs Check Statistics (Admin)
- Administrator performs Check Statistics (Admin)
- Administrator performs Get GPS Location of the E-Scooter
- Administrator performs Send a push notification
- Administrator performs Show E-Scooter Battery Status

Actor Customer

- Customer performs Check usage statistics (Customer)
- Customer performs Check usage statistics (Customer)
- Customer performs Check-in E-Scooter
- Customer performs Check-out E-Scooter
- Customer performs Check-out E-Scooter
- Customer performs Delete Account
- Customer performs Delete Account
- Customer performs Find an E-Scooter on the map
- Customer performs Find an E-Scooter on the map
- Customer performs Get Notifications
- Customer performs Get Notifications
- Customer performs Give Feedback
- Customer performs Give Feedback
- Customer performs Log-in
- Customer performs Log-in
- Customer performs Log-in
- Customer performs Log-out
- Customer performs Manage Account
- Customer performs Manage Account
- Customer performs Pay the service
- Customer performs Register
- Customer performs Report Problems with Scooter
- Customer performs Report Problems with Scooter

- Customer performs Reserve an E-Scooter
- Customer performs Subscribe to the service
- Customer performs Subscribe to the service

Actor Maintenance man

- Maintenance man performs Add E-Scooter to System
- Maintenance man performs Find an E-Scooter on the map
- Maintenance man performs Get GPS Location of the E-Scooter
- Maintenance man performs Get GPS Location of the E-Scooter
- Maintenance man performs Report Problems with Scooter
- Maintenance man performs Show E-Scooter Battery Status
- Maintenance man performs Show E-Scooter Battery Status

Actor Maps Service Provider

- Maps Service Provider performs Download Offline Maps
- Maps Service Provider performs Download Offline Maps
- Maps Service Provider performs Get GPS Location of the Customer
- Maps Service Provider performs Get GPS Location of the E-Scooter

Actor Payment Service

- Payment Service performs Pay the service
- Payment Service performs Pay via Wallet
- Payment Service performs Pay via Wallet
- Payment Service performs Pay via subscription plan

Known other usecases

UseCase Register, UseCase Add E-Scooter to System, UseCase Check Statistics (Admin), UseCase Check-out E-Scooter, UseCase Check-in E-Scooter, UseCase Choose subscription plan, UseCase Log-in, UseCase Log-out, UseCase Manage Account, UseCase Find an E-Scooter on the map, UseCase Download Offline Maps, UseCase Subscribe to the service, UseCase Pay the service, UseCase Report Problems with Scooter, UseCase Give Feedback, UseCase Analyze Customer Feedback, UseCase Get Notifications, UseCase Send an email confirmation