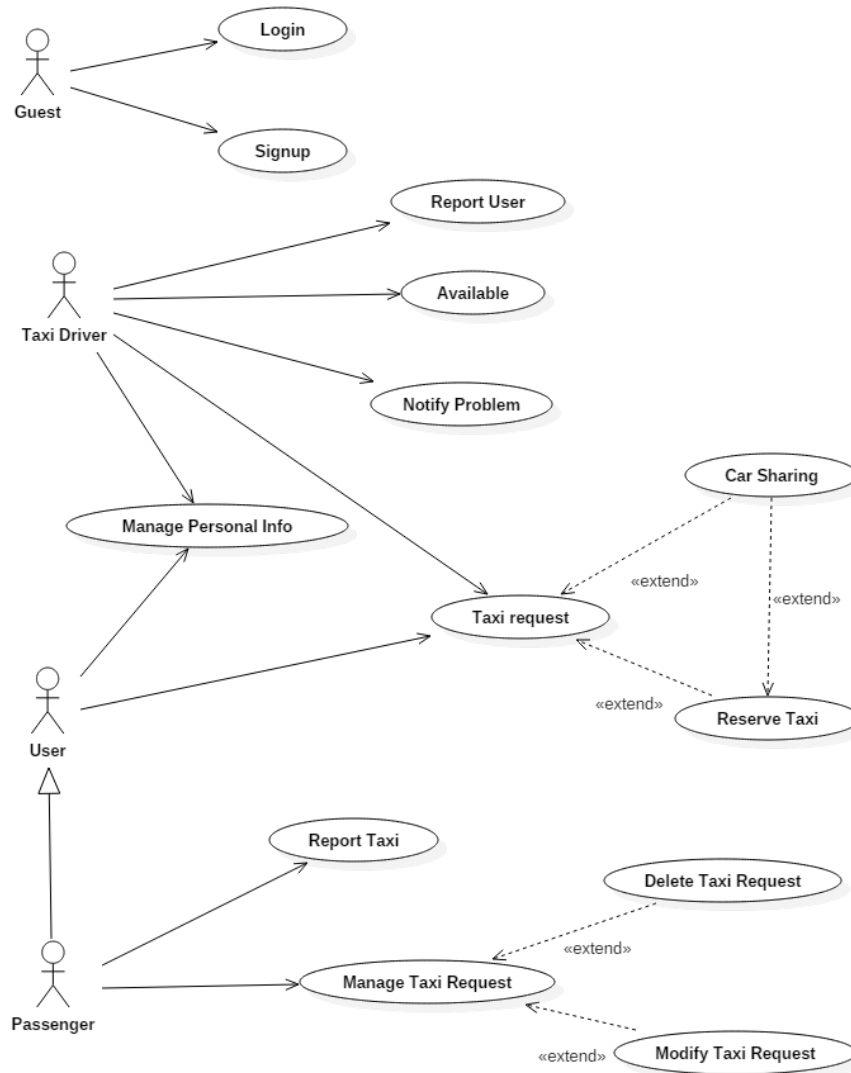


# 1 Specific Requirement

## 1.1 External interfaces

## 1.2 UseCase Diagram



## 1.3 Scenario

*Adam need to take a taxi*

### **1.3.1 Signup**

Adam is new into the System and wants to signup. He requires the Passenger registration page, fills the form and submit the request to the System. If the email and username are unique the System confirms to Adam the successful of the operations and redirect Adam to the Login page, otherwise it display an error message

### **1.3.2 Login**

Adam, now registered, insert the username and password in the login form and click the login button, the System checks the informations and, in case of correct informations, redirect Adam to his own User profile page, otherwise shows an error message

*Hector is a Taxi Driver logged into the System*

### **1.3.3 Available**

Hector, logged into the System, start his working day opening his Taxi Driver profile page and telling his availability to the System. The System change the queue and notify the position (2nd) in the queue to Hector.

### **1.3.4 Taxi Request**

Adam, now logged into the System, book a Taxi for the 5PM (now 11AM) inserting informations in the Taxi Request page (previously required from the system) and submit the data into the System. Now the System check the informations, if are wrongs display to Adam a message with the error, otherwise start asking to the Taxis queue. The System ask the request to the first of the queue, who reject the request and is moved at the end of the queue. The System repeat the operation with the new first of the queue, Hector now confirm the request and take in charge the request. The System now confirm the successful to Adam, and is now a Passenger.

### **1.3.5 Manage Taxi Request**

Adam opens the Manage Taxi Request page to change the book hour to 6PM, change the hour and, because are missing two hours, the System change the information and notify the change to Hector.

### **1.3.6 Report Taxi**

Hector takes Adam, during the ride Hector lights a cigarette and Adam open the Report Taxi page to complain this behavior, fill the form with the informations and submit this into the System. The System updates the Taxi Driver informations and confirm the end of the operation to Adam.

### **1.3.7 Report User**

After the ride, Adam annoyed for the behavior of Hector refused to pay the Taxi Driver. Hector open the report User page, fills the informations and submit the data. The System updates the User informations adding a negative report and notify the end of the operation to Hector.

### **1.3.8 Manage Personal Informations**

Adam opens his User Profile page, click on the edit button and change his email, submit the new informations. The System checks the informations, if are right update the Adam profile and notify the success of the operation to Adam.

### **1.3.9 Notify Problem**

Adam opens his User Profile page, click on the edit button and change his email, submit the new informations. The System checks the informations, if are right update the Adam profile and notify the success of the operation to Adam.

*Samuel is a Taxi Driver logged into the System, available in the zone which Hector is now*

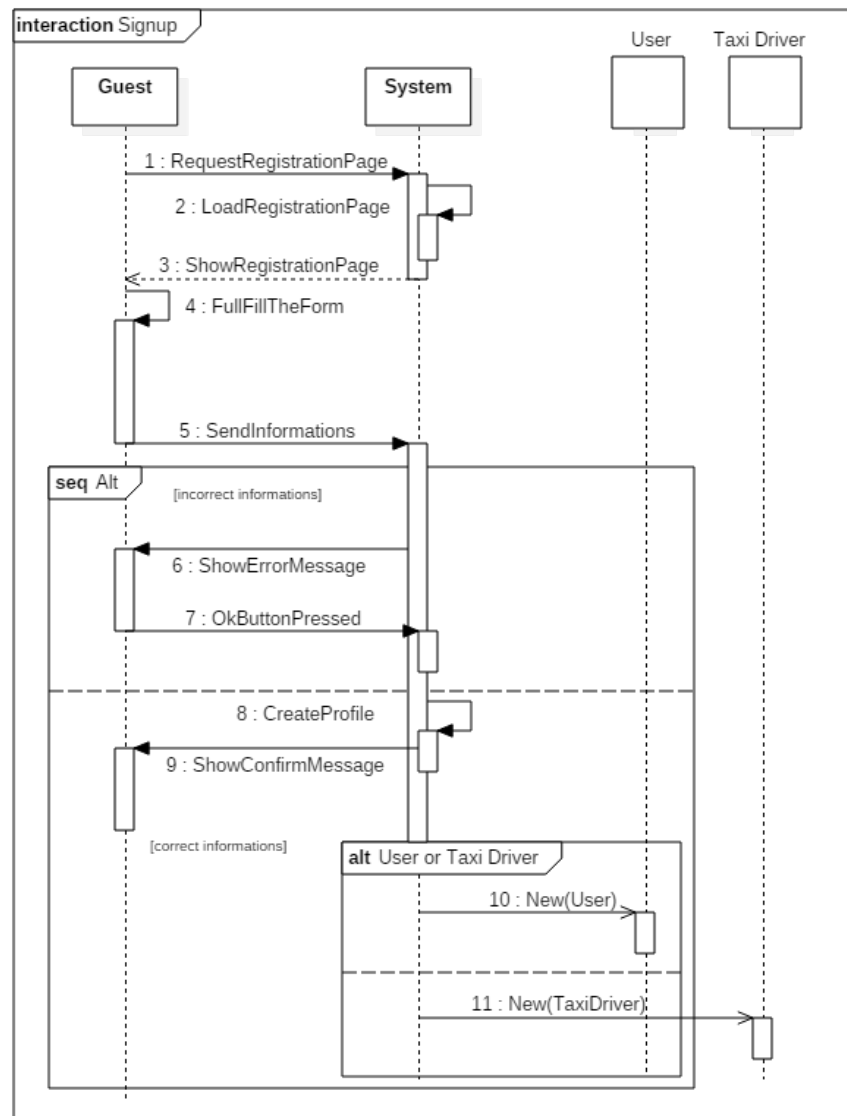
### **1.3.10 Report Problem**

Hector during a ride, has a problem with his taxi motor and thought the Report Problem page report the problem filling the form and submitting. The System insert the problem and ask to Hector if needs a new Taxi. Hector positively answer to the request and the System ask to the first of the queue to reach Hector and his Passenger. Samuel, the first of the queue, accept the request and reach his colleague.

## 1.4 Functions

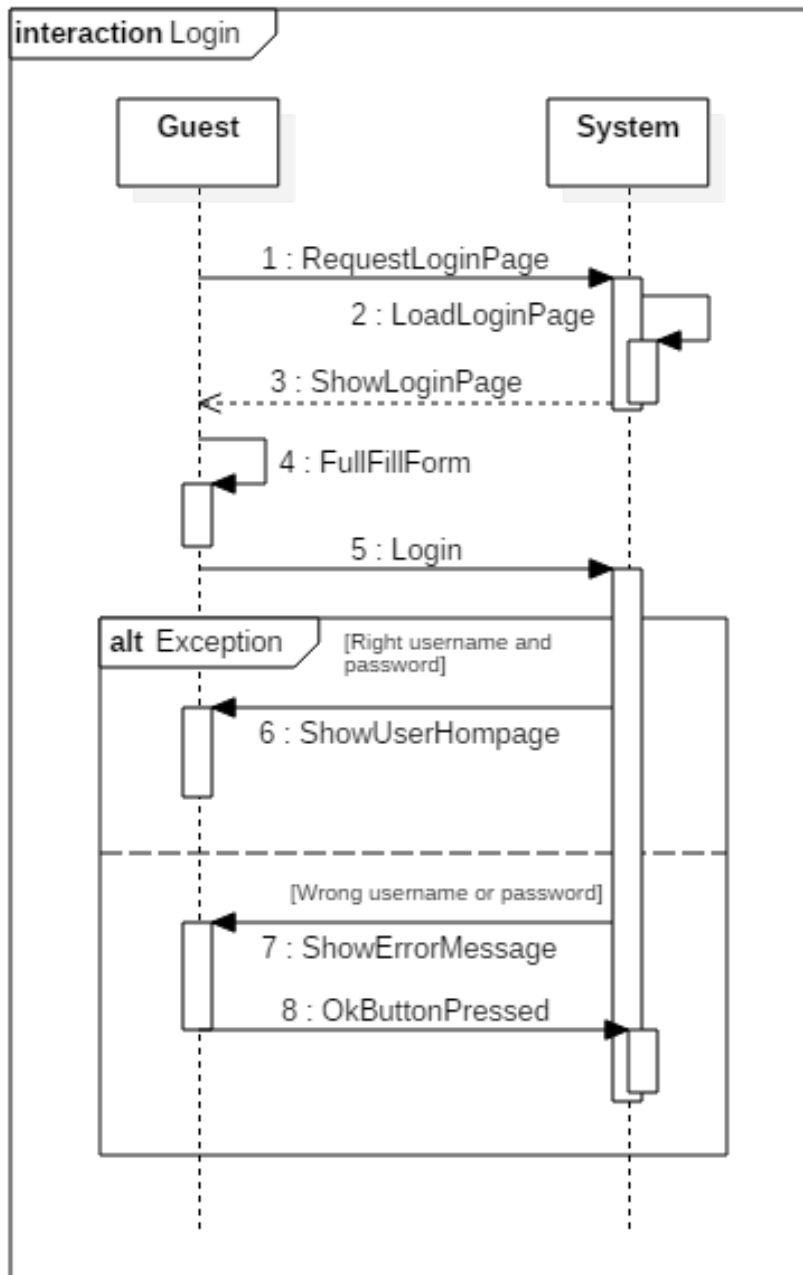
### 1.4.1 Signup

Actors	Guest
Preconditions	The Guest is not registered into the system
Execution Flow	<ol style="list-style-type: none"><li>1. The Guest request the registration page</li><li>2. The System requires the Signup information</li><li>3. The Guest fill the form and submit the request</li><li>4. The System check the uniqueness of the Username and E-Mail</li><li>5. The System create the User (or Taxi drive) profile</li><li>6. The System send the confirm the the Guest</li></ol>
Postconditions	The guest is now a User o a Taxi driver
Exceptions	The E-Maill or username are not unique or, in the case of taxi driver signup, the licenses are not valid



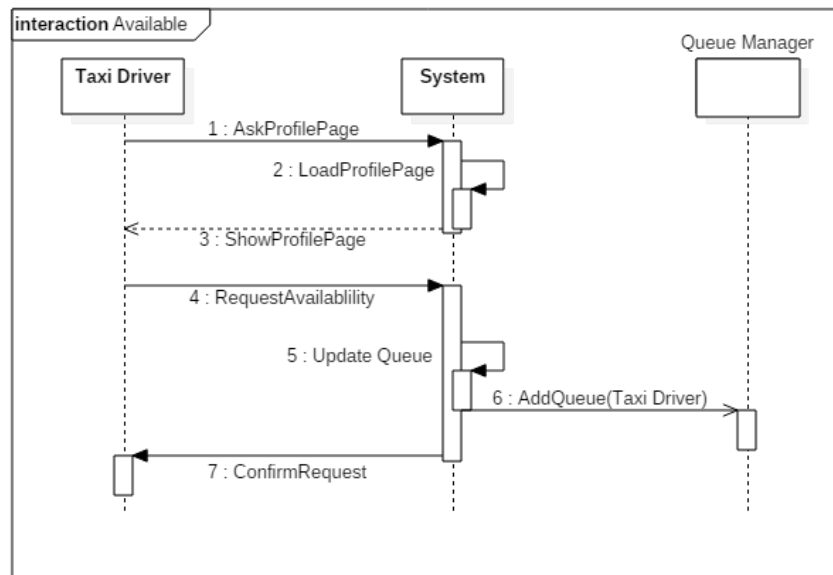
### 1.4.2 Login

Actors	Guest
Preconditions	The Guest has already a profile into the System
Execution Flow	<ol style="list-style-type: none"><li>1. The Guest request the login page</li><li>2. The System requires the Login information (Username, Password)</li><li>3. The Guest fill the form and submit the request</li><li>4. The System check the Username and Password</li><li>5. The System send the confirmation of Login</li><li>6. The Guest is logged into the System</li><li>7. The Guest is redirected to the User profile Page</li></ol>
Postconditions	The guest is now a User or Taxi Driver
Exceptions	The Username, password couple is incorrect, so the Guest cannot log in



### 1.4.3 Available

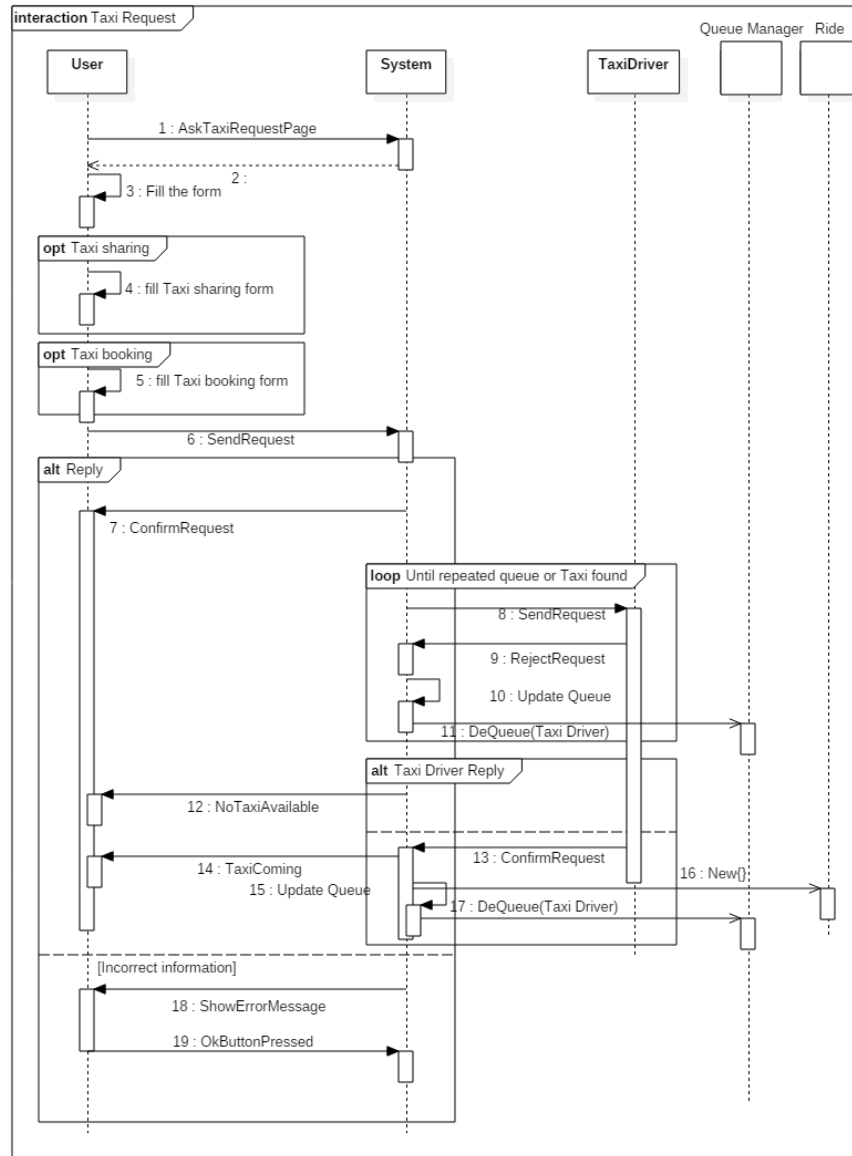
Actors	Taxi Driver
Preconditions	
Execution Flow	<ol style="list-style-type: none"> <li>1. The Taxi Driver request the Taxi profile page</li> <li>2. The System returns the personal informations of the Taxi</li> <li>3. The Taxi Driver notify the availability to the System</li> <li>4. The System update the local queue</li> <li>5. The System confirm the request to the Taxi Driver</li> </ol>
Postconditions	The Taxi Driver is now Available
Exceptions	<ul style="list-style-type: none"> <li>• The Taxi Driver is located in a invalid zone</li> <li>• The Taxi Driver is carrying a Passenger</li> <li>• The Taxi Driver is yet available</li> </ul>





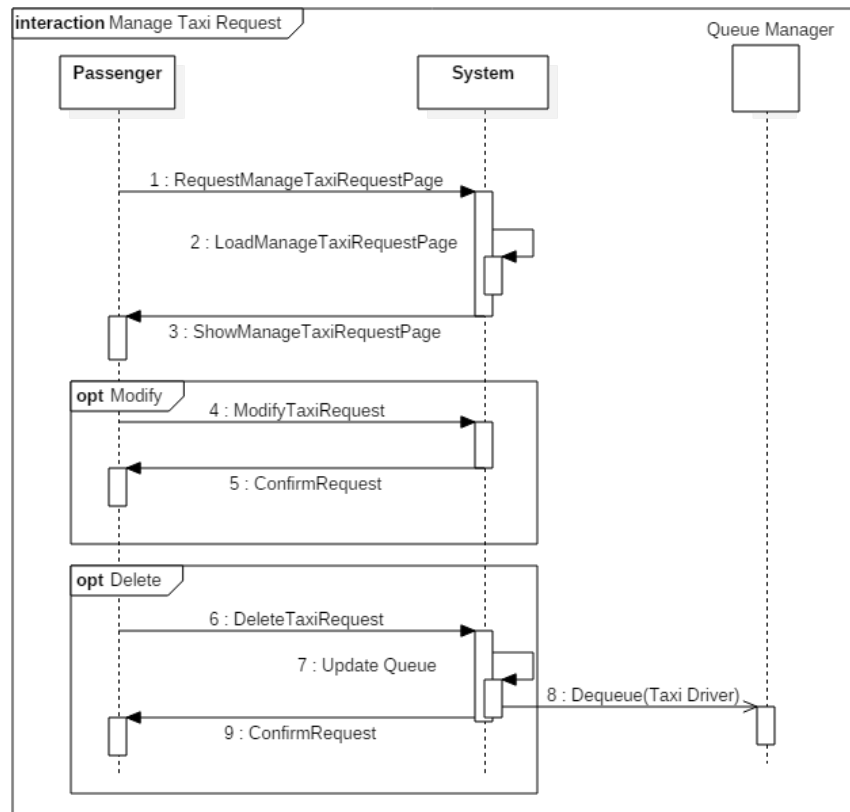
#### 1.4.4 Taxi Request

Actors	User, Taxi Driver
Preconditions	The User should not be banned
Execution Flow	<ol style="list-style-type: none"><li>1. The Guest request the Taxi Request page</li><li>2. The System requires the type of request the User wants to perform</li><li>3. The User provide the choice to the System</li><li>4. The System ask for the information to perform the choose request</li><li>5. The User fill the form and send the information to the System</li><li>6. The System forward the request to the first Taxi Driver of the local queue</li><li>7. If the Taxi Driver answer positively to the request he take in charge the User, otherwise he deny the request</li><li>8. If the Taxi Driver answered positively the System notify to the User the incoming taxi and change the availability of the Taxi Driver, otherwise the System update the queue and forward the request to the new first of queue until the queue is not repeated</li><li>9. If there are no Taxis available the System notify to the User the unavailability</li></ol>
Postconditions	If the request has positively response the User is now a Passenger
Exceptions	<ul style="list-style-type: none"><li>• The User provides incorrect informations in the request form</li><li>• The User is not into a valid position (<i>e.g. outside the city</i>)</li></ul>



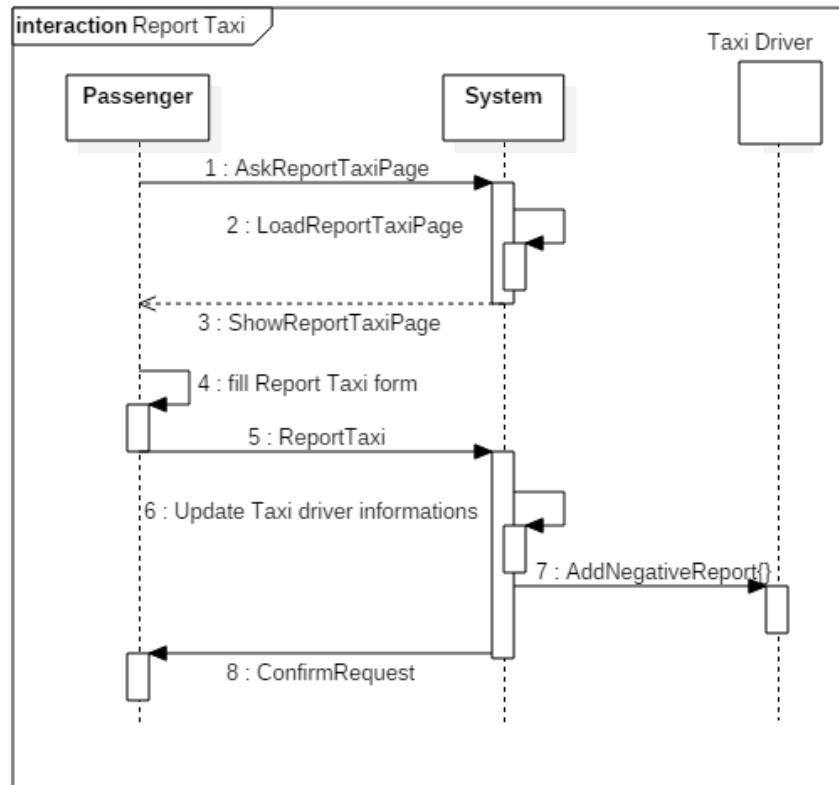
#### 1.4.5 Manage Taxi Request

Actors	Passenger
Preconditions	
Execution Flow	<ol style="list-style-type: none"><li>1. The Passenger request the Taxi Request Management page</li><li>2. The System load the page and return it to the Passenger</li><li>3. The Passenger can modify the Request full filling the modify Request</li><li>4. The System modify the request and return a confirm to the Passenger</li><li>5. The Passenger can delete the request, submitting the operation to the System</li><li>6. The System update the queue and return a confirmation to the Passenger</li></ol>
Postconditions	<ul style="list-style-type: none"><li>• If the Passenger choose to modify the Request, the request is updated</li><li>• If the Passenger choose to delete the Request, the request is canceled and the taxi is now in queue</li></ul>
Exceptions	<ul style="list-style-type: none"><li>• The Passenger provides incorrect informations in the modify request form</li><li>• The Passenger cancel the request too late</li></ul>



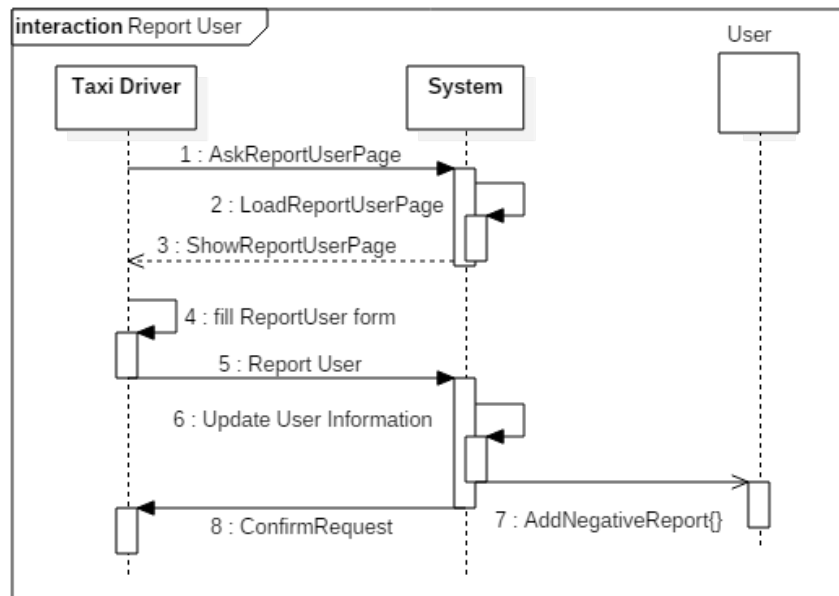
#### 1.4.6 Report Taxi

Actors	Passenger
Preconditions	
Execution Flow	<ol style="list-style-type: none"><li>1. The Passenger request the Report Taxi page</li><li>2. The System elaborate the request and ask to the Passenger the report information</li><li>3. The Passenger fill the form and submit the report</li><li>4. The System check the data obtained</li><li>5. The System update the Taxi Driver informations</li><li>6. The System notify to the Passenger the successful of the operation</li></ol>
Postconditions	The Taxi Driver is reported by the current Passenger
Exceptions	The Passenger provides wrong informations in the report form



#### 1.4.7 Report User

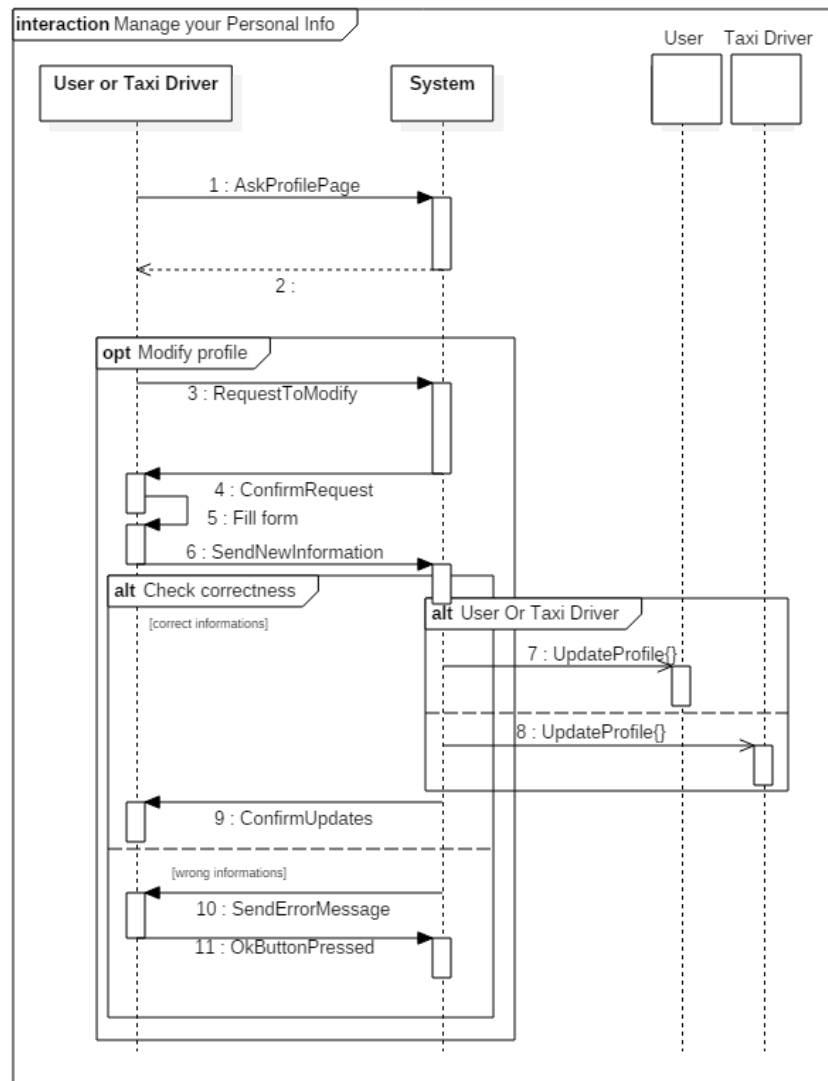
Actors	Taxi Driver
Preconditions	The Taxi Driver should had carried the User past 24 hours
Execution Flow	<ol style="list-style-type: none"> <li>1. The Taxi Driver request the Report User page</li> <li>2. The System elaborate the request and ask to the Taxi Driver the report information</li> <li>3. The Taxi Driver fill the form and submit the report</li> <li>4. The System check the data obtained</li> <li>5. The System update the User informations</li> <li>6. The System notify to the Taxi Driver the successful of the operation</li> </ol>
Postconditions	The User is reported by the Taxi Driver
Exceptions	The Taxi Driver provides wrong informations in the report form



#### 1.4.8 Manage Personal Informations

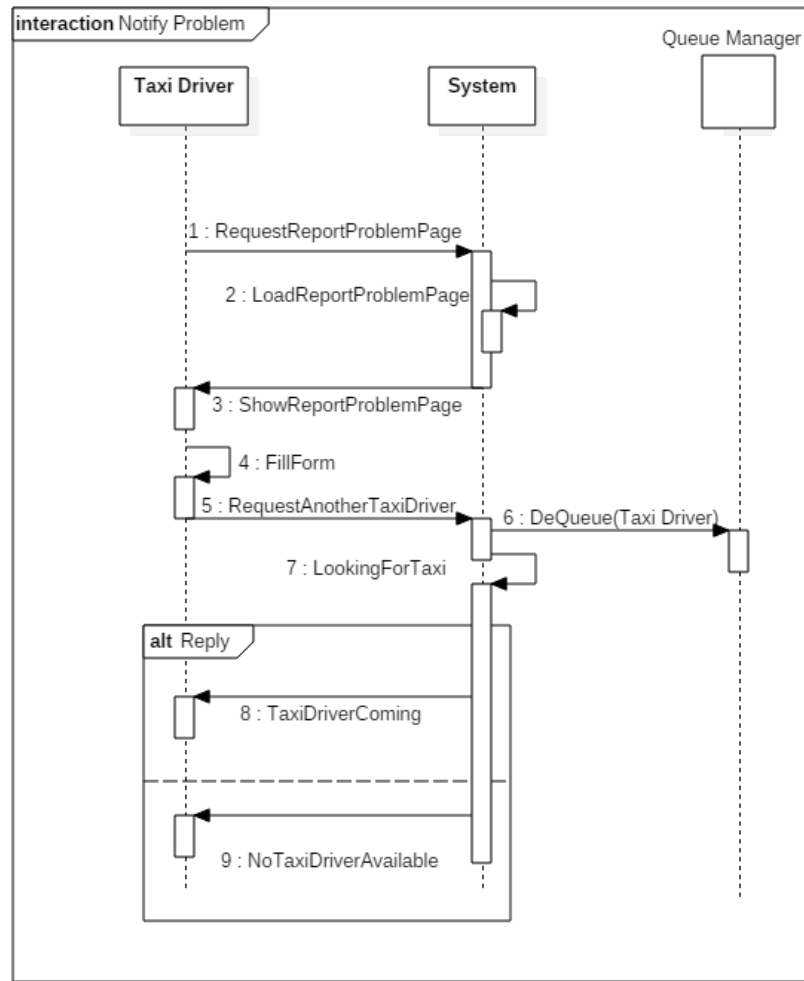
Actors	User or Taxi Driver
Preconditions	
Execution Flow	<ol style="list-style-type: none"><li>1. The User (or Taxi Driver) request the User profile page</li><li>2. The System returns the personal informations of the User (or Taxi Driver)</li><li>3. The User (or Taxi Driver) can request the edit of the profile</li><li>4. The System return the editable informations of the profile</li><li>5. The User (or Taxi Driver) can edit those data and send the changes to the System</li><li>6. The System check the correctness of the new Informations</li><li>7. If the informations are correct send the Change Confirm to the User (or Taxi Driver), otherwise send back an error message</li></ol>
Postconditions	If the User request a modify profile and submit correct informations the Profile is changed
Exceptions	The User provides wrong informations





#### 1.4.9 Notify Problem

Actors	Taxi Driver
Preconditions	
Execution Flow	<ol style="list-style-type: none"><li>1. The Taxi Driver request the Report Problem page</li><li>2. The System returns the page of the Report requiring problem informations</li><li>3. The Taxi Driver fill the form and send the information</li><li>4. The Taxi Driver can request another Taxi for the User who is carrying (if is carrying one)</li><li>5. If the Taxi Driver request another Taxi the System looks for a available Taxi Driver</li><li>6. If exist an available Taxi change the carrier of the User with the new Taxi Driver and send him, otherwise notify the unavailability</li><li>7. The System return the success of the operation</li></ol>
Postconditions	The problem is submitted into the System
Exceptions	<ul style="list-style-type: none"><li>• The Taxi Driver is located in a invalid zone</li><li>• The Taxi Driver fill the form with wrong informations</li></ul>



## 1.5 Performance requirements

- The System should support at least all the Taxi Driver registered into the City Database
- The System should elaborate User incoming informations
- The System should provide the faster path for each ride
- The System should calculate the final price of the ride with a maximum error of 10%
- The System should provide to a Passenger the arrival time (maximum error 20%) and change the path in case of traffic

## **1.6 Logical database requirements**

## **1.7 Design constraints**

- GPS precision limitations: average 3m error
- Internet congestion

## **1.8 Standards compliance**

## **1.9 Software system attributes**