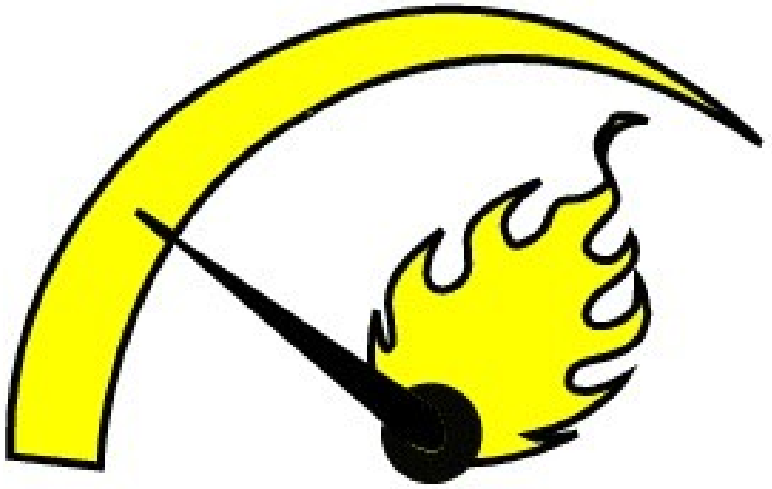
Software Engineering 2

Requirement Analysis and Specification Document

Authors
Paolo Paterna, Lara Premi
Email: paolo.paterna@mail.polimi.it, lara.premi@mail.polimi.it

Politecnico di Milano, Italy

11th Nov 2015



myTaxiService

Scope

- The scope of this system is to manage taxis in a city.
- A town is divided in zone of 2 square kilometers and, for each zone, the system defines a queue, composed by the identifier, which is the vehicle plate, of free taxis in that specific zone.
- A user can require a taxi ride from a zone, but can also book one for another moment, using the web application or the mobile one. About long-term reservations, the user can also, after have created one, modify the date or the hour or both of his/her booking, and he/she can delete it.

Actors

- Users: users can request a taxi ride or book one, using the mobile app or the website.
- Taxi: it is an entity, composed by the taxi driver and the car; it works for a taxi company and it provides the service to the users by transporting them to their destination.
- Taxi not yet registered: taxis that are not yet registered on the system and do not belong to any queue.

Goals

- Booking a Taxi ride.
- Update a long-term reservation.
- Delete a long-term reservation.
- Locate the Users.
- Managing Taxis location.
- Managing Taxis queue.
- Allow Taxi to register in the system.
- Notify the users about their booking changes.
- Notify Taxis about new available rides.

Definitions pt.1

- myTaxiService: the system described in this document.
- User: someone who uses the services offered by myTaxiService, sometimes called also passenger.
- Taxi: entity composed by the taxi driver and the taxi car that transports the users.
- Booking: generic reservation, that is, it can be long-term reservation or a short-term one.
- Short-term reservation: booking, requested from the user, for the current time.

Definitions pt.2

- Long-term reservation: booking, requested from the user, characterized by a time, at least two hours before the requested service, or by a different day from the current one.
- Zone: part of the city, large 2 square kilometers, that is disjoint with reference to the other zones.
- Queue: list of available taxis in a specific zone.
- Busy Taxi: taxi that is doing a ride.
- Available Taxi: taxi not busy.

Product Perspective
Users

- This system is a stand alone system, but it can be extended using the featured API.
- Users can use the web application or the mobile app to access the service site.
- Users can update theirselves about this service and, then book a taxi;
- Users can view their reservations and the system notifications

Product Perspective
Taxi

- Uses the mobile application

- The taxi driver can view some information about the current ride, the notifications from the system, previous rides.

- The taxi driver can send to the system notifications about delays that will be forwarded to the user by the system.

Product Functions
User

- The system allows the users to book a taxi using the website or the mobile application installed on their smart phones.

- They also can reserve a taxi but they must do it at least 2 hours before the chosen hour; in this case the system assigns a unique code for every reservation in order to allow users to modify or delete their reservation.

- The system also sends notification to the users about the status of their bookings and reservation or information about waiting time for their ride.

Product Functions
Taxi

- The system allows taxi to register into it in order to be added to the queue and be given new passengers.

- The system also sends notification to taxis about new jobs. It also allows taxis to notify users the system about delays.

Costraints

- The system must maintain the privacy about personal data of users and taxi drivers.
- The system must allow operations to run in parallel.
- The system must work 24 x 7.
- In case of long-term reservation, the user must book a taxi at least two hours before the meeting time.
- In case of long-term reservation modification or elimination, the user must do it at least 15 minutes before the meeting time.

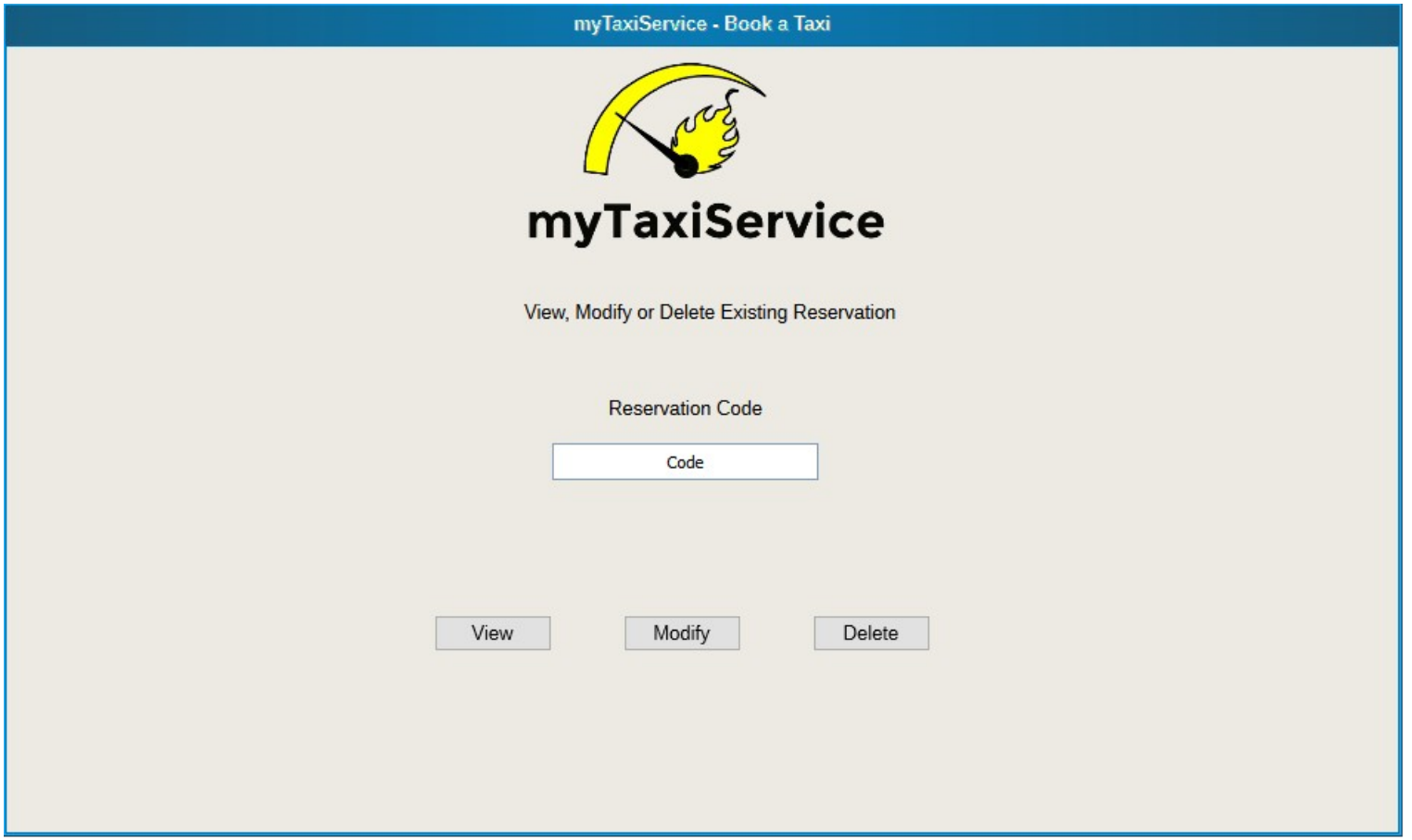
Assumptions

- The terms "code" and "identifier" relate to the same thing, that is the taxi plate.
- The fair queue management is characterized by: at most 10 minutes of wait; a FIFO method to organize the taxis queue.
- The zones in which the city is divided are always the same, don't change position and they never overlap.
- The taxis that are outside of the city or aren't in any defined zone are considered busy until they return in a known zone.
- The taxis that don't answer to the system notifications of new incoming rides are considered as busy.
- The GPS works always, pooled every 2 minutes.

User Interfaces (1)

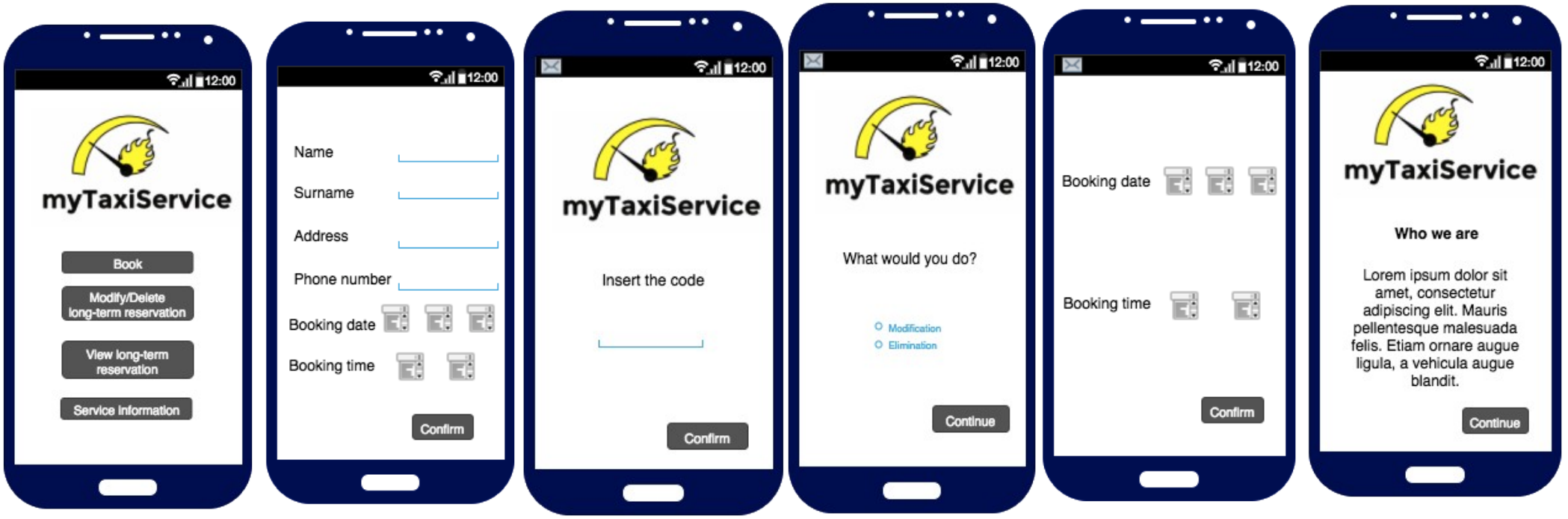
User Interfaces (2)

User Interfaces (3)

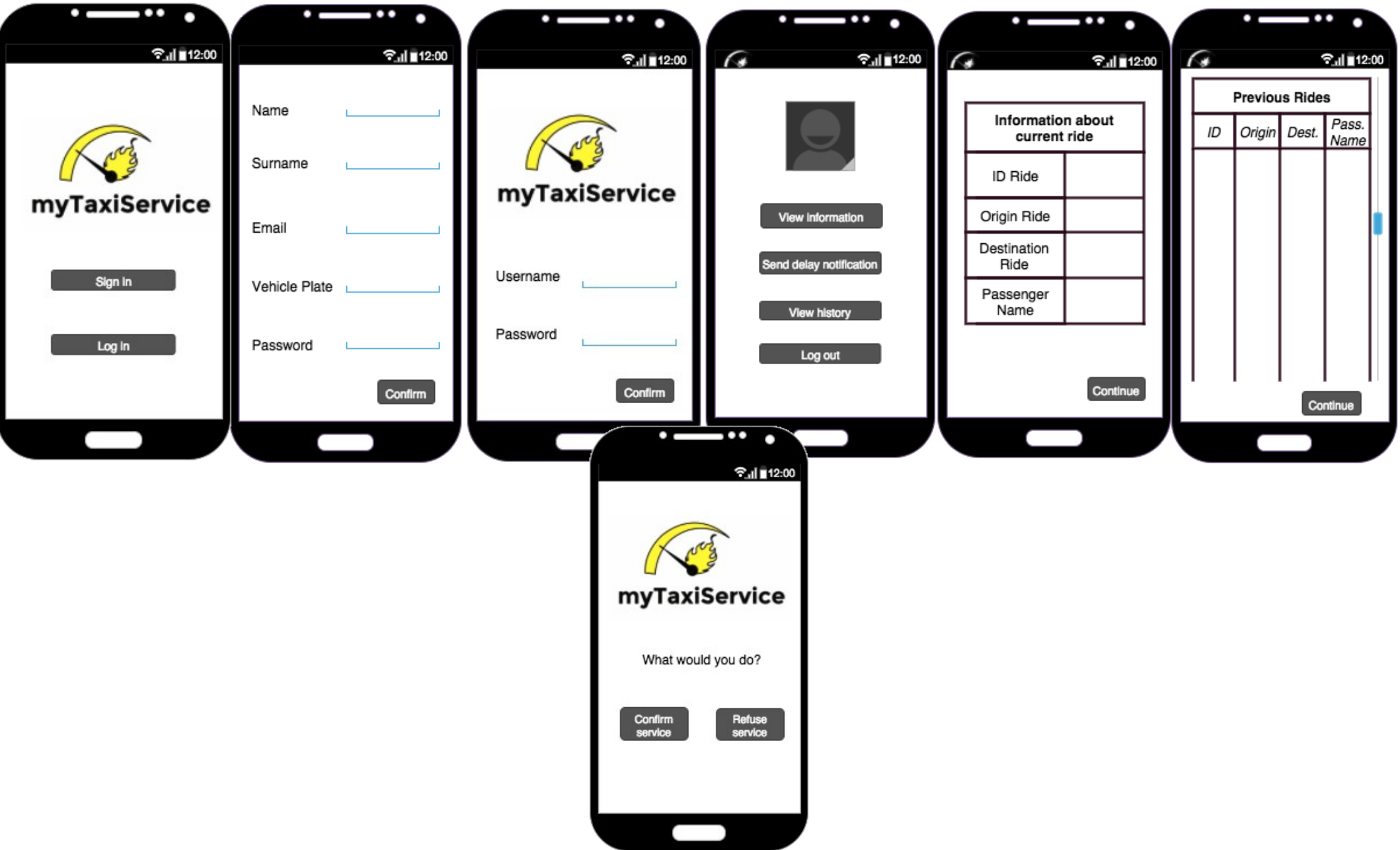


User Interfaces (4)

User Interfaces (5)



Taxi Interfaces



Hardware interfaces

- A machine dedicated to database-server, capable to manage high-volume transactions;
- A machine to run the main software and the web-interface;
- A private redundant gigabit network connecting both machines;
- A public redundant connection to internet for the web-server machine, with adequate speed and scalable traffic quota.

Software interfaces

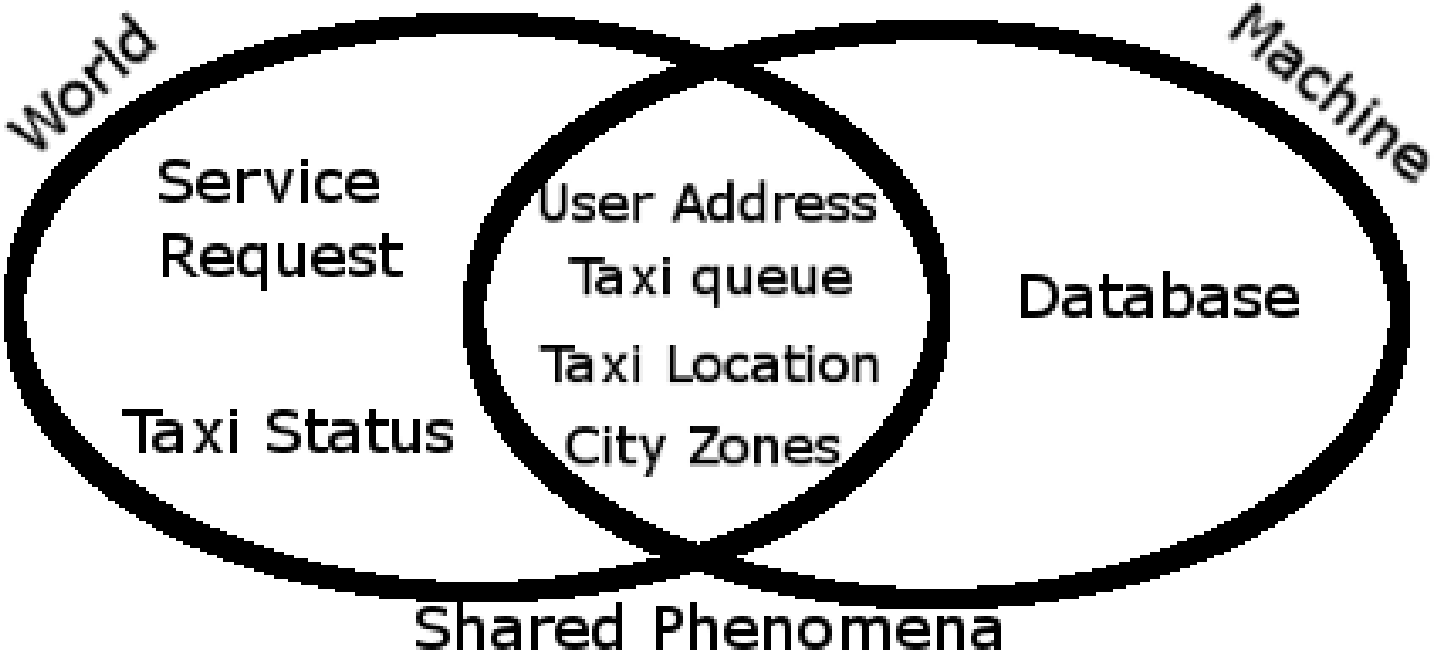
- A web server to provide information to browsers and mobile apps;
- The webserver must be capable to run PHP scripts;
- A DBMS, compatible with PHP, to store all the data managed by the system.

Functional Requirements

- Analyzing the goals we can find the functional requirements of the system

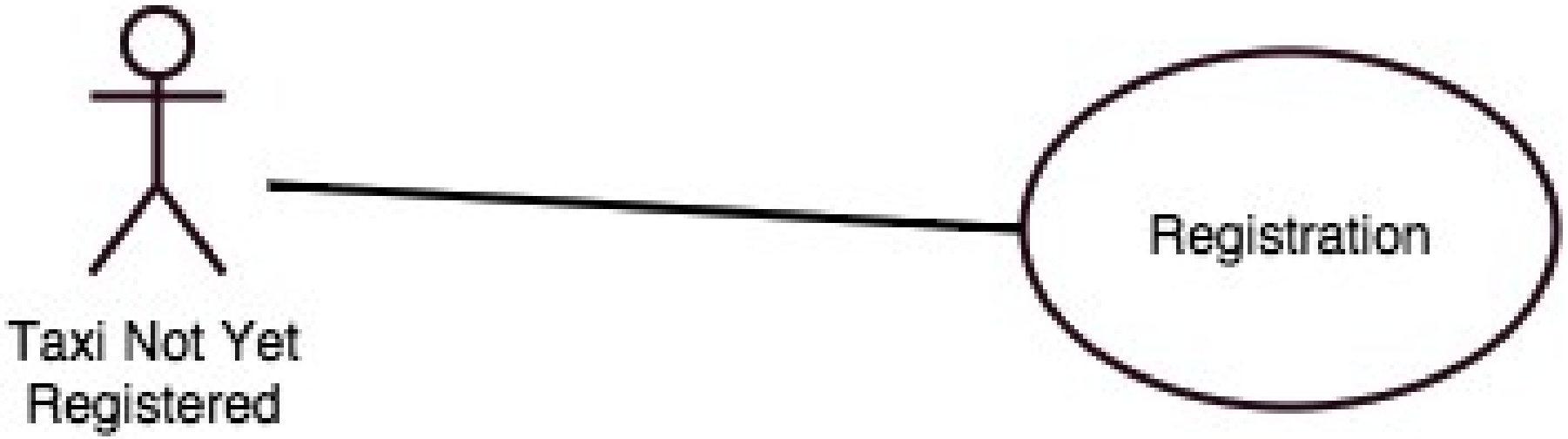
- For example **Managing Taxi queue:**
- Only in the case that a taxi is available, the system inserts its identifier in the zone queue, that is unique.
- If there is a new ride, the system chooses the first taxi in the queue, so, using a FIFO policy.
- If there is a new ride, but the interested zone queue is empty, the system checks the queues of the four adjacent zones; the system continues in this way as long as it finds an available taxi.

The World and the Machine

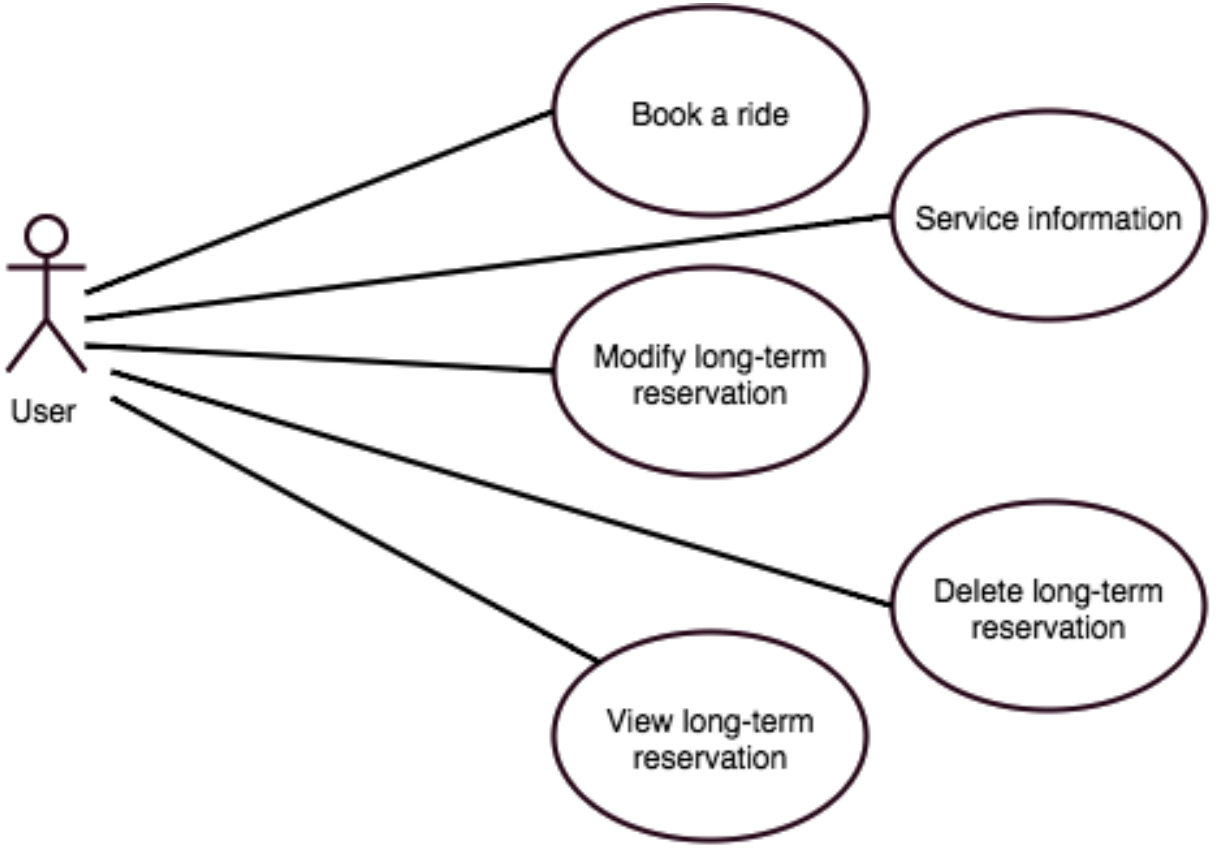


Service request and Taxi status are events that happen in the physical world; the machine then uses the data coming from the world to build a representation of it using taxi queue, taxi location and city zones. The database is only stored in the machine and contains all the data about users to send notification, long term reservations and data about taxis.

Use Cases (1)



Use Cases (2)
User



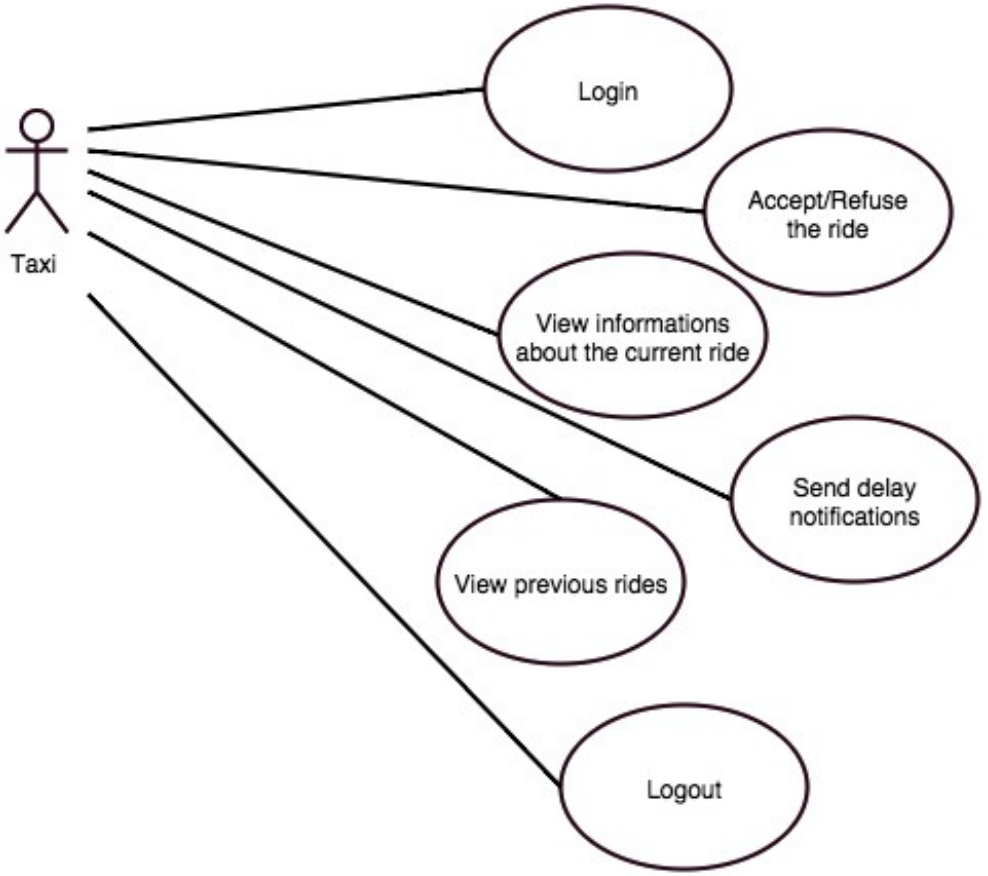
Use Cases (3)
Example

Name	Modify long-term reservation
Actors	User
Event Flow	The user • accesses to the web site or the mobile application; • clicks on "Modify/Delete long-term reservation"; • inserts the alphanumeric code; • clicks on the "Confirm" button; • chooses "Modification"; clicks on "Continue" link; modifies the date and/or the hour; clicks on "Confirm" button;

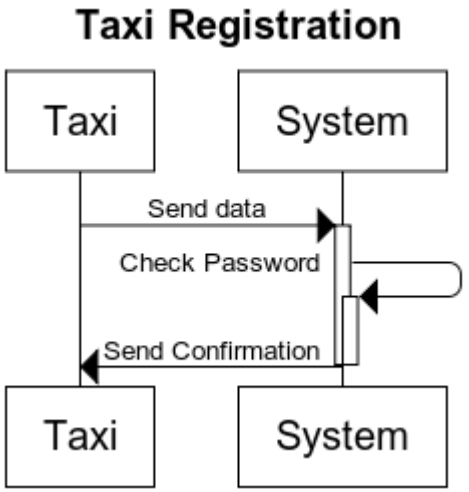
Use Cases (4) Continue

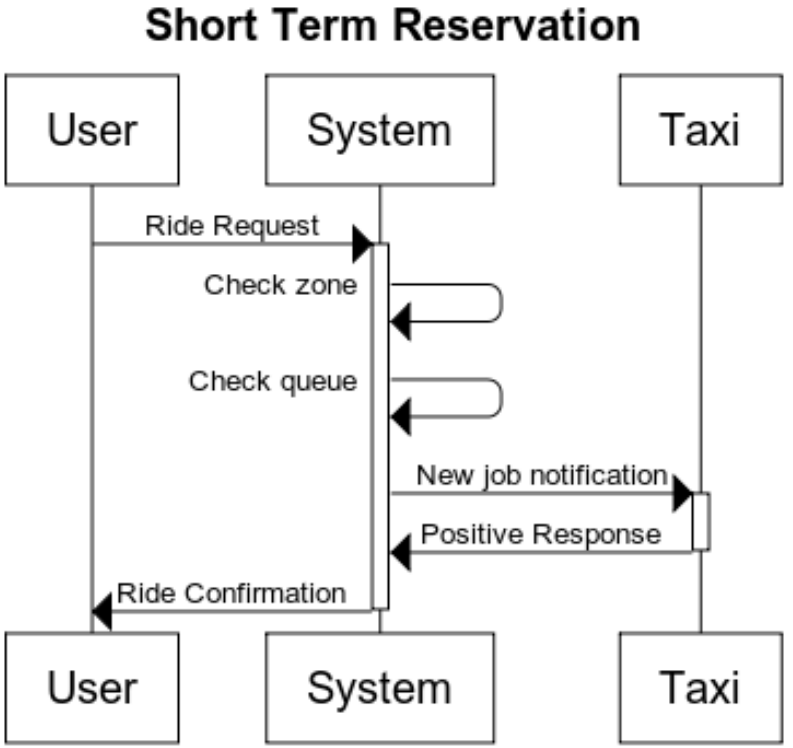
Event Flow (cont.)	the system • sends a confirmation via SMS to the user; • modifies the changed data of the long-term reservation from the database.
Exit Conditions	No exit conditions
Exceptions	• Alphanumeric code inserted wrongly; • data and/or hour not valid.

Use Cases (5)

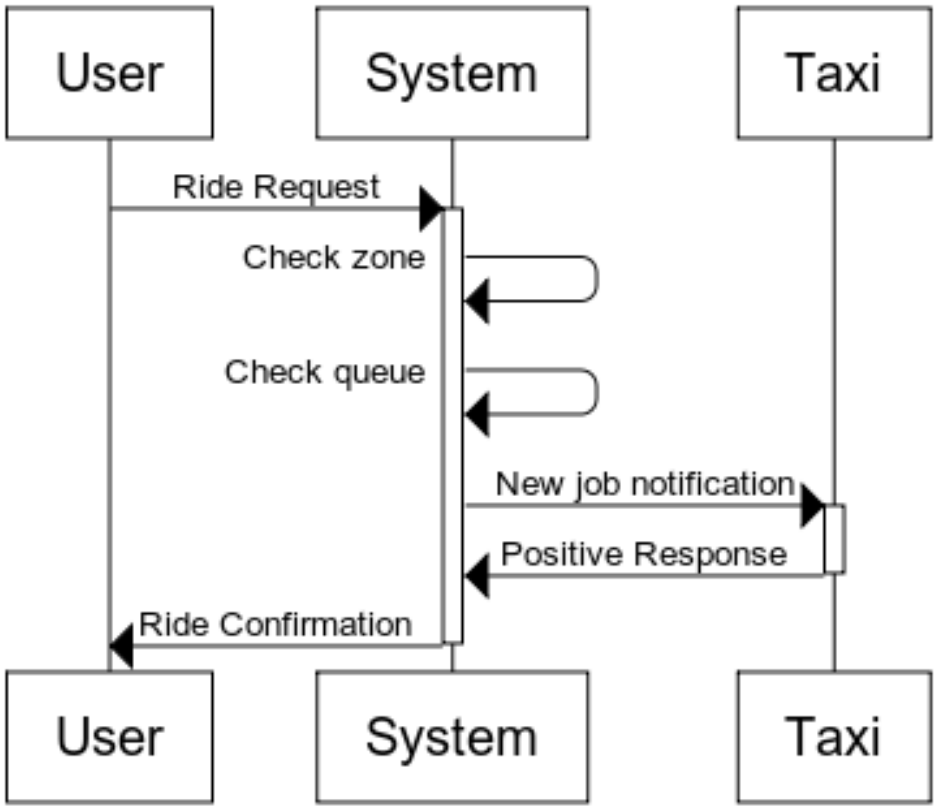

Login
Accept/Refuse the ride
View informations about the current ride
Send delay notifications
View previous rides
Logout
Taxi

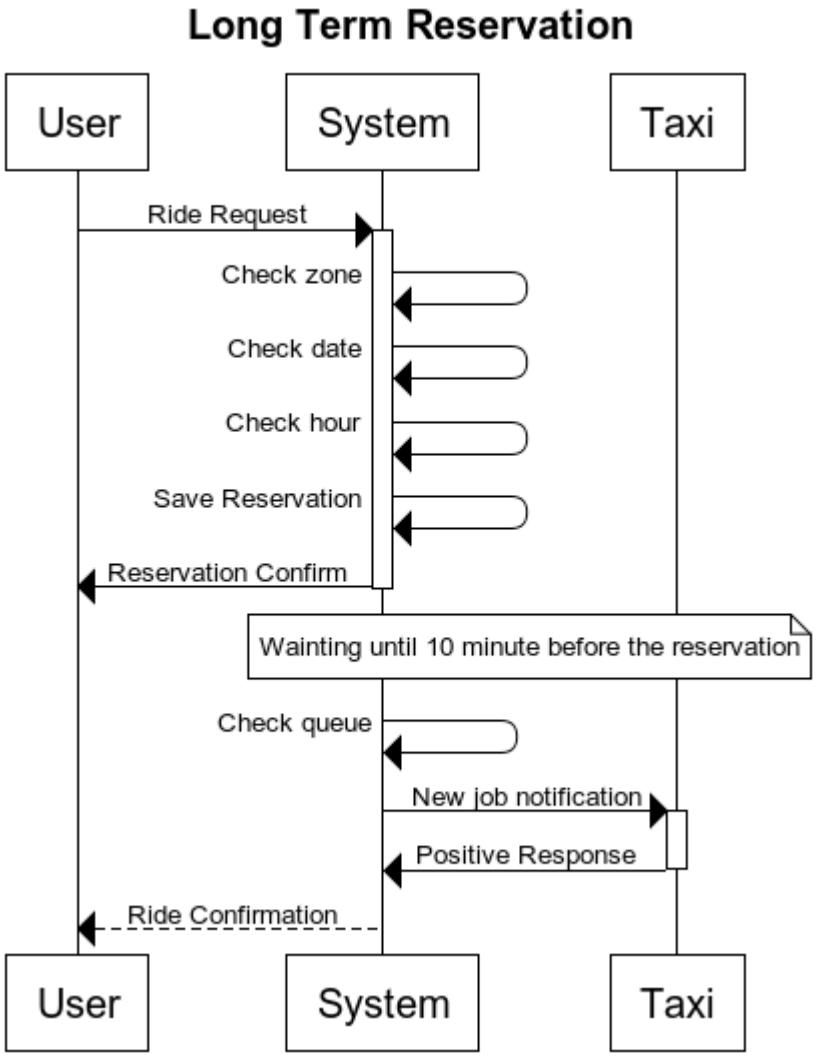
Sequence Diagrams (1) Examples



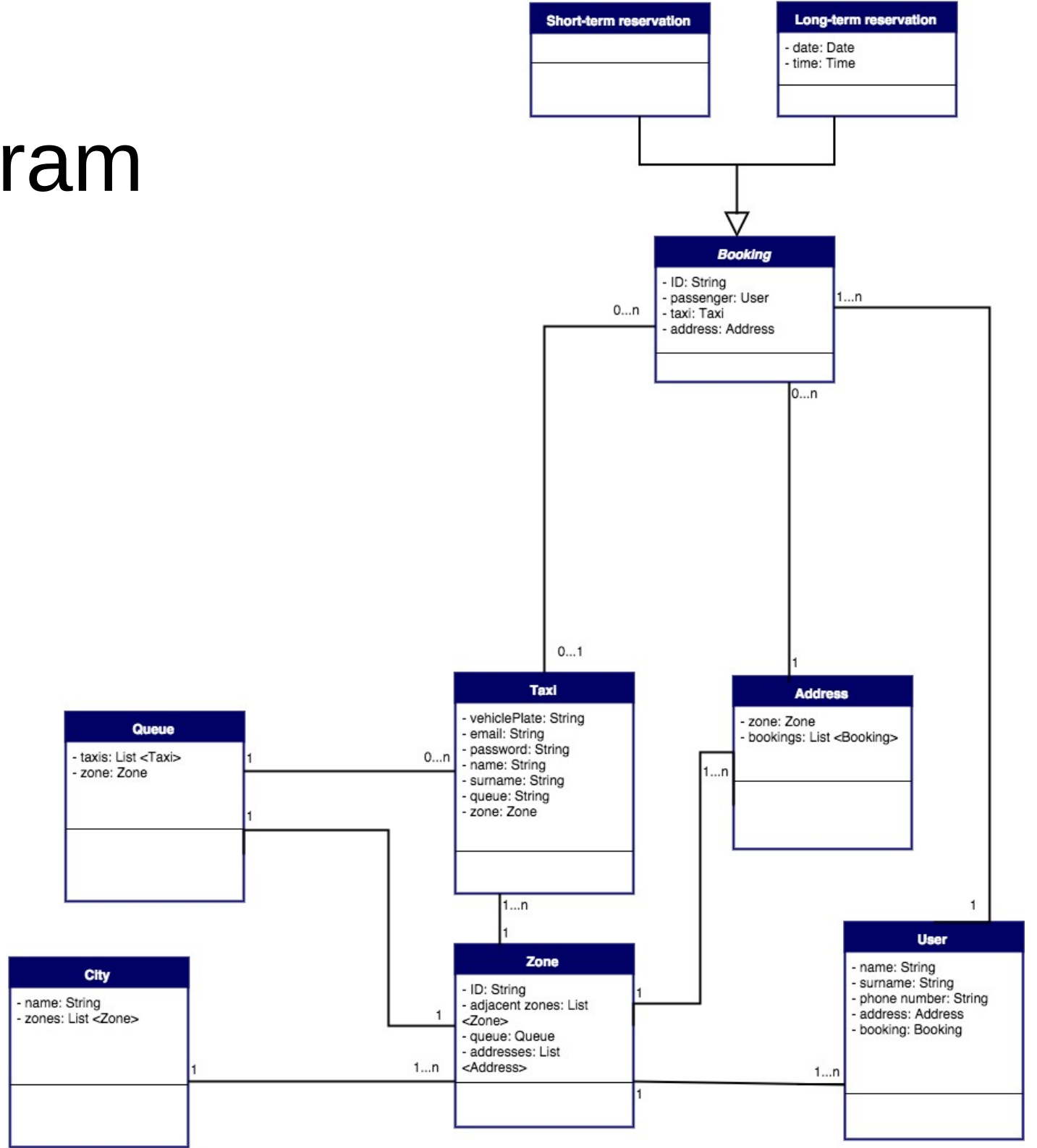


Sequence Diagrams (2) Examples

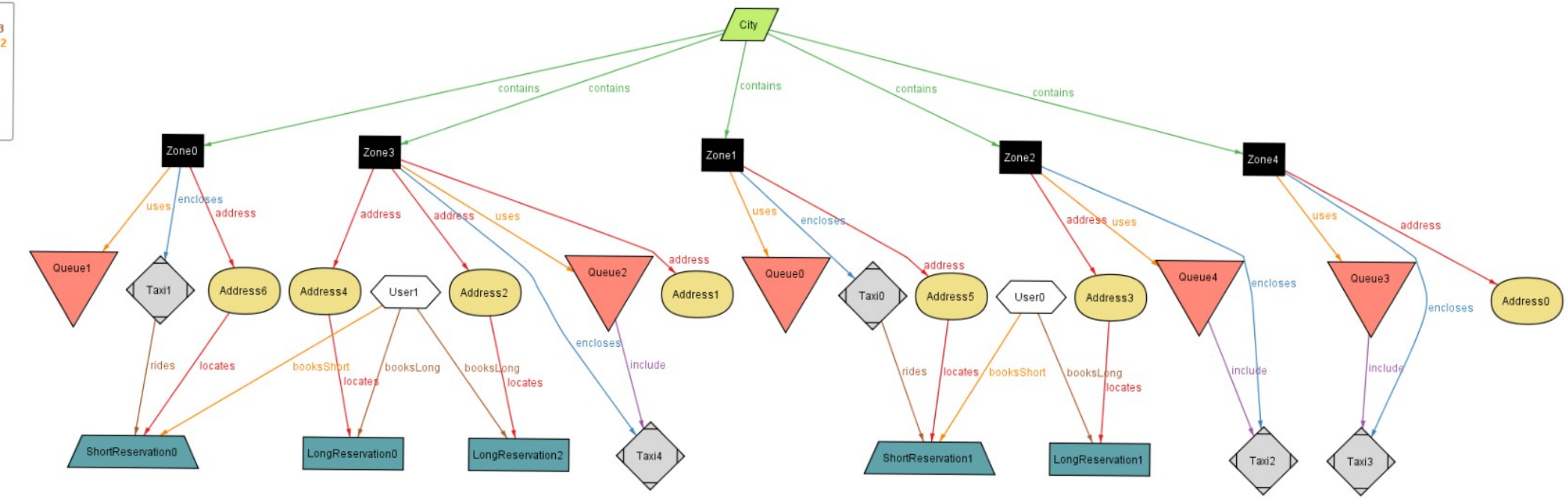




Class Diagram



Alloy


City
contains
Zone0
Zone3
Zone1
Zone2
Zone4
uses
encloses
address
Queue1
Taxi1
Address6
Address4
User1
Address2
Queue2
Address1
Queue0
Taxi0
Address5
User0
Address3
Queue4
Queue3
Address0
rides
locates
booksShort
booksLong
include
ShortReservation0
LongReservation0
LongReservation2
Taxi4
ShortReservation1
LongReservation1
Taxi2
Taxi3

Non Functional Requirements

- The system will provide a secure access to their personal page for all the taxi drivers, using a username and a password.
- The system should be working 24/24 7/7, with an availability of at least 98% (mean down time in a year: 7.30 days).
- The UI of both mobile and web applications has to be very responsive, and it should allow to execute actions on the system spending less time as possible.

Hours of Work

In order to write this document the we have done the following hours of work:

- Paolo Paterna: 40 Hrs
- Lara Premi: 40 Hrs