

Release date 09-12-2015

Software Engineering 2: MyTaxiService

Design Document V1.0

Dimitar Anastasovski, Marco Colombo

Contents

1	Introduction	2
1.1	Purpose	2
1.2	Scope	2
1.3	Definitions	3
1.4	Abbreviations	3
1.5	Acronymous	3
2	Architectural Design	4
2.1	Overview	4
2.2	High level components and interactions	5
2.3	Component view	6
2.4	Deployment view	7
2.5	Runtime view	8
2.6	Component interfaces	10
2.7	Selected architectural styles and patterns	11
2.8	Other design decisions	12
3	Algorithm Design	13
4	User Interface	14
4.1	User interface design	14
5	Requirements traceability	24
5.1	From requirements to diagrams	24
6	References	29
6.1	Software and used Tools	29
6.2	Working hours	29

Chapter 1

Introduction

1.1 Purpose

This document is intended to give a description of the software product. It consists of a diagram architectural, focusing on particular points and on some design pieces. (need write something more)

1.2 Scope

The main accent is to simplify and optimize the access of passengers to the system and to guarantee fair management of taxi queues. We will build flexible and user-friendly web application and a mobile application that will run on Android and IOS mobile phones. This application can be used by anyone who previously will be register on the registration page. After the registration is done the user will have a user name and password that should remember for furthermore usage of the system. The passenger can call a taxi after a successful logging on the application. After that he can call a taxi and he will be informed about the code of the incoming taxi, waiting time. On the other hand taxi drivers will have a mobile application where the major purpose will be to inform the system about their availability, confirmation of a certain call and global map navigation. City is divided into taxi zones that are uniquely associated with corresponding taxi queues for efficient usage of the system.

1.3 Definitions

- Request: Passenger filled form for immediate ride
- Reservation: Passengers can request for a vehicle at least 2 hours before the ride and can reserve his ride
- User: Is a customer who already registered and logged into the system
- Taxi driver: Is a person who legally drives taxi (with driver license and work license) already registered and logged into the system as a driver
- System: Is the system that has to be designed
- Taxi zone: Are the zones in which the city is divided in

1.4 Abbreviations

No abbreviations are been used in this document

1.5 Acronymous

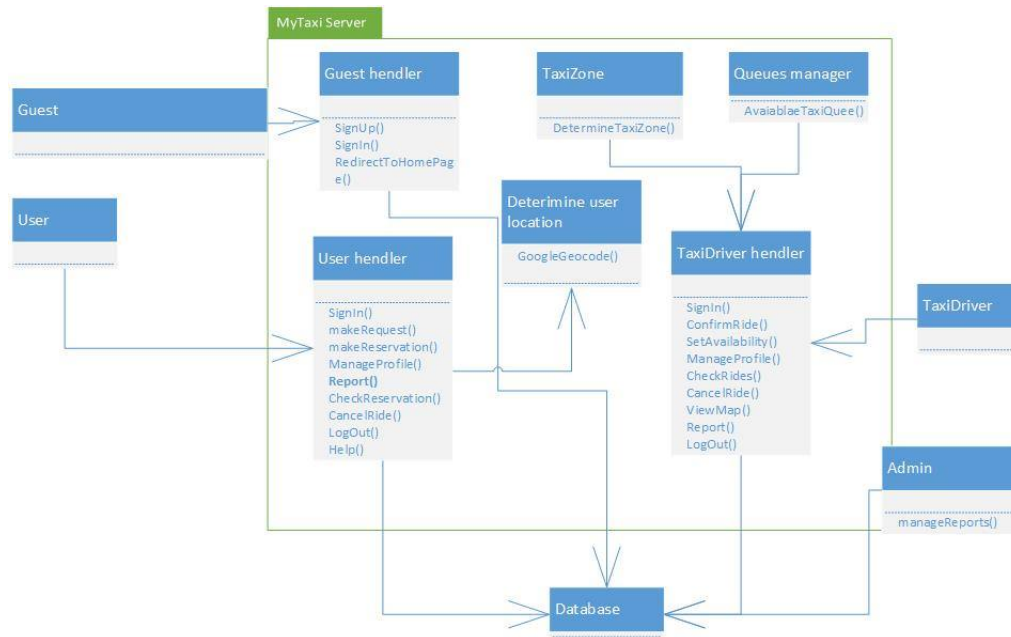
- IEEE Standard 1016-2009 for Information Technology–Systems Design–Software Design Descriptions
- Specification Document: myTaxiService Project

Chapter 2

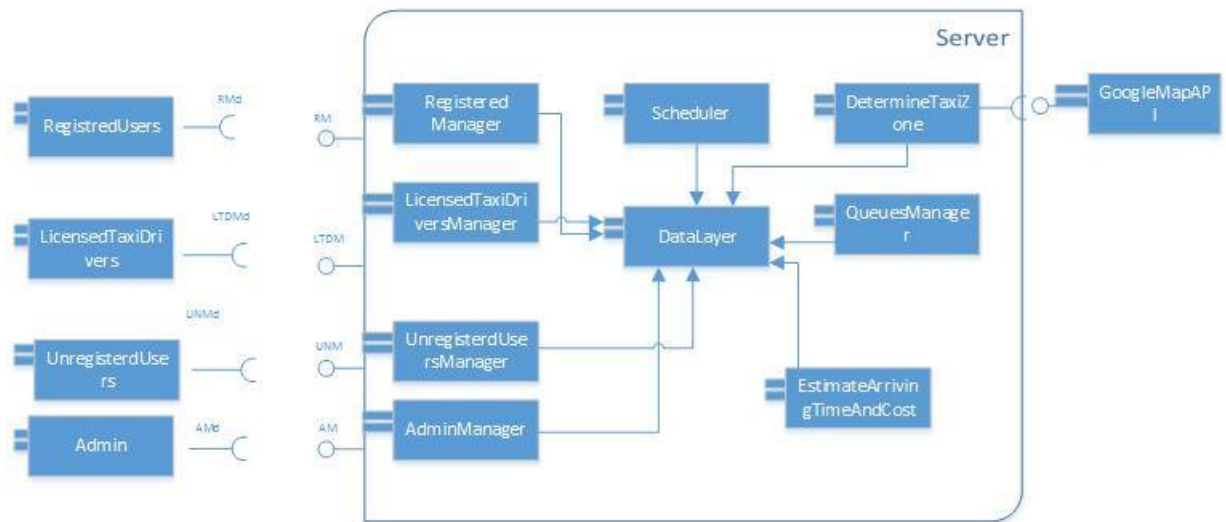
Architectural Design

2.1 Overview

2.2 High level components and interactions

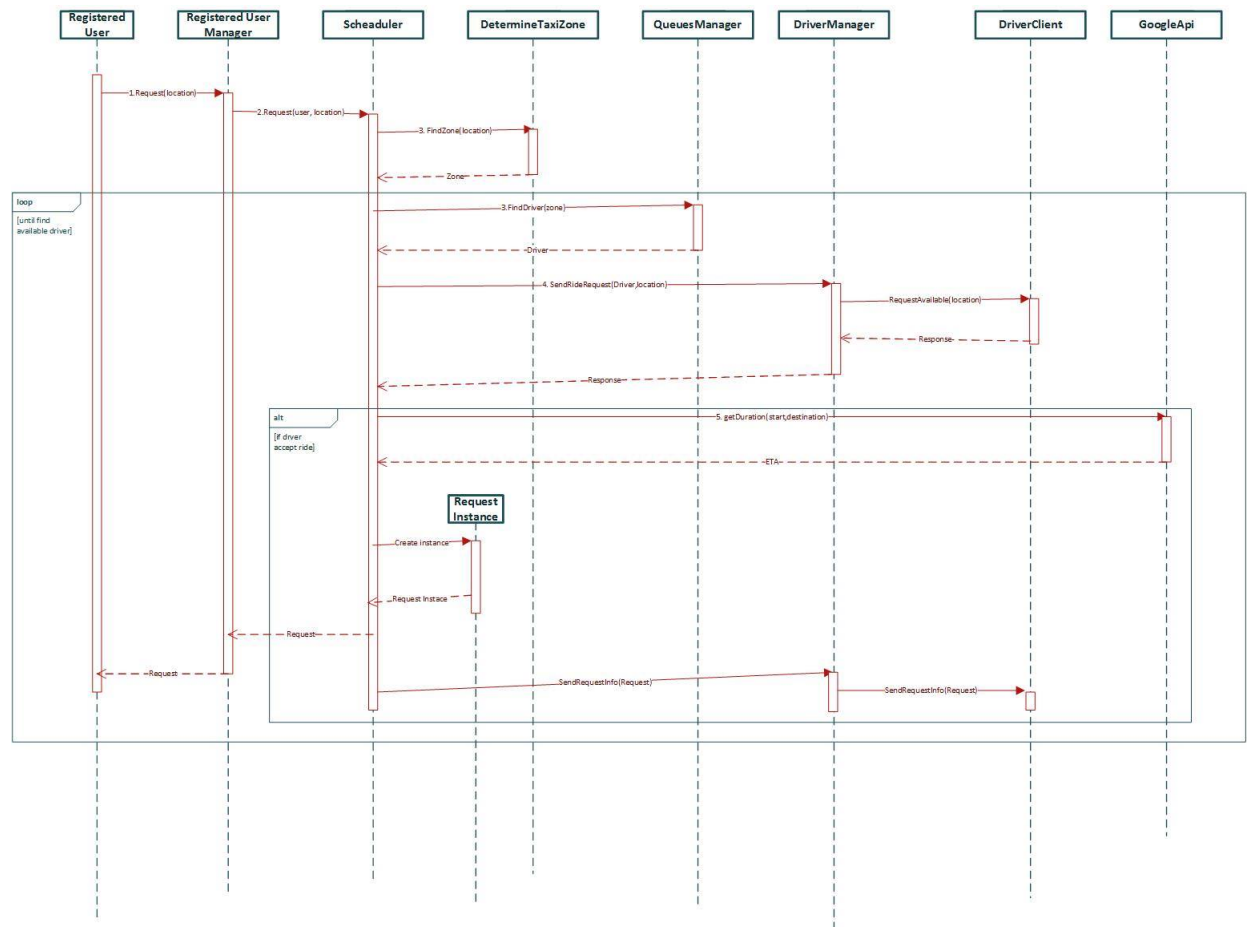


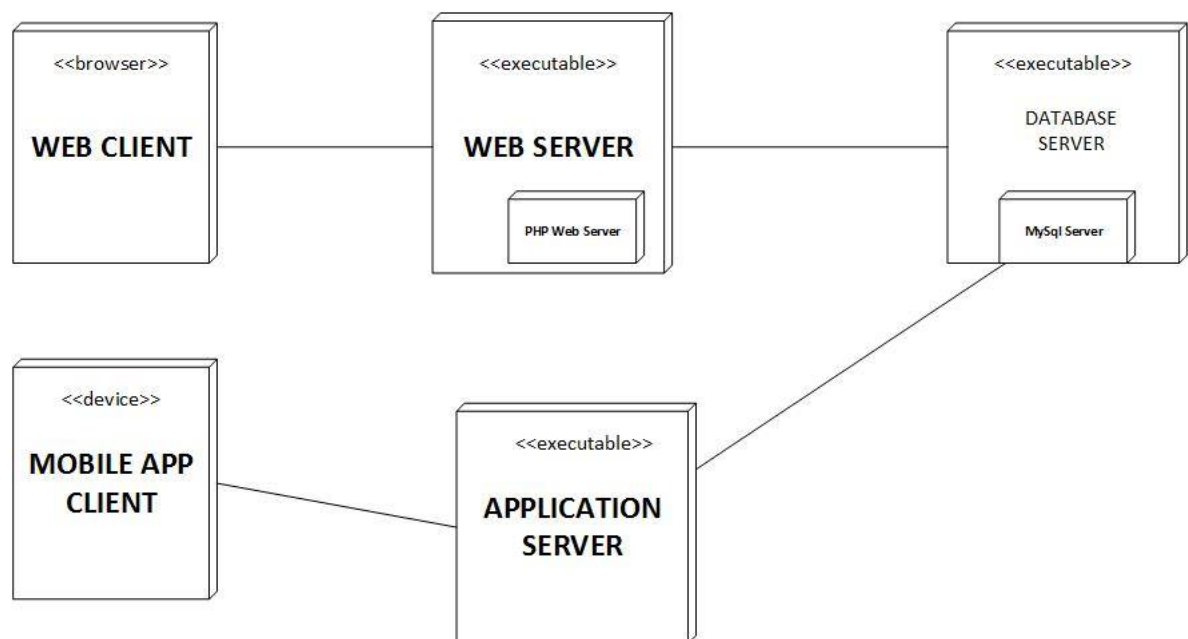
2.3 Component view



2.4 Deployment view

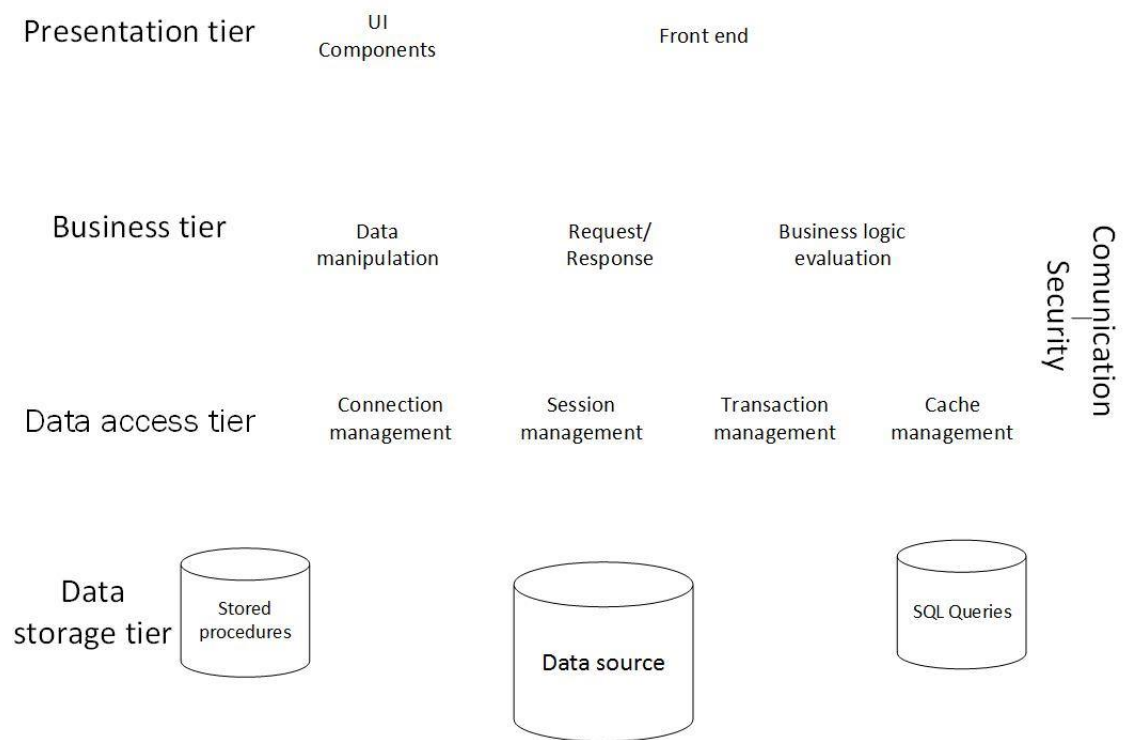
2.5 Runtime view





2.6 Component interfaces

2.7 Selected architectural styles and patterns



2.8 Other design decisions

Chapter 3

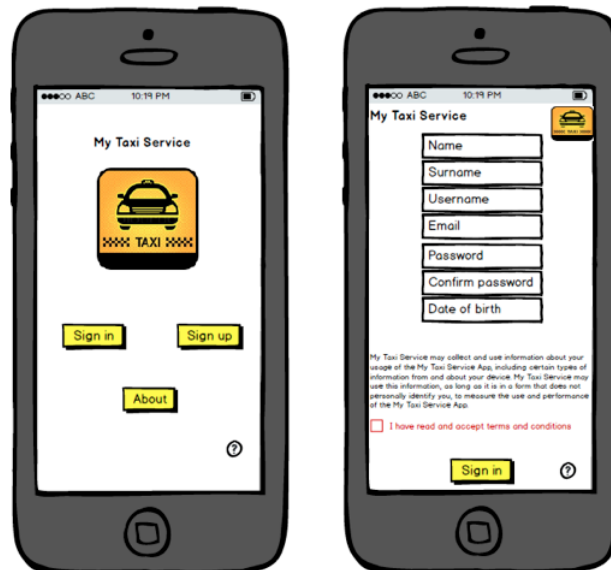
Algorithm Design

Chapter 4

User Interface

4.1 User interface design

Homepage & registration page



The passenger can access to the registration page and register himself to the application.

Login



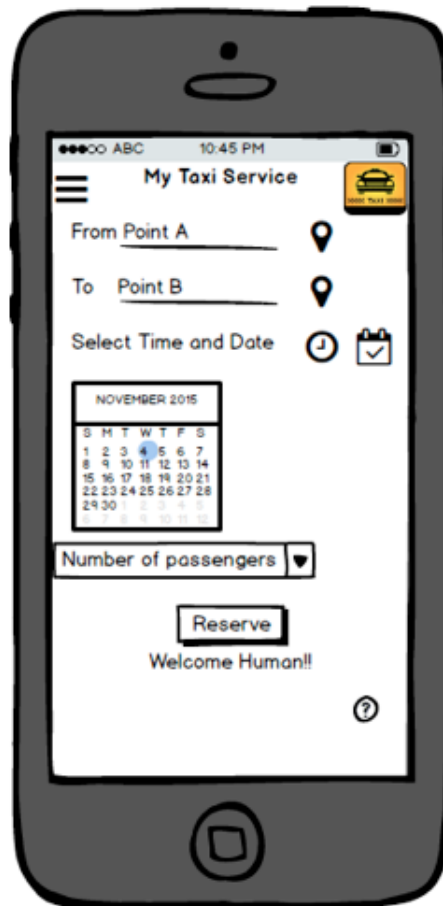
This is the login page. Here the user after entered his data can sign in the application.

Menu



This is the menu page, here the user can navigate and choose what to do. There are several options like: reserve taxi, check reservation, profile, help or log out.

Reservation



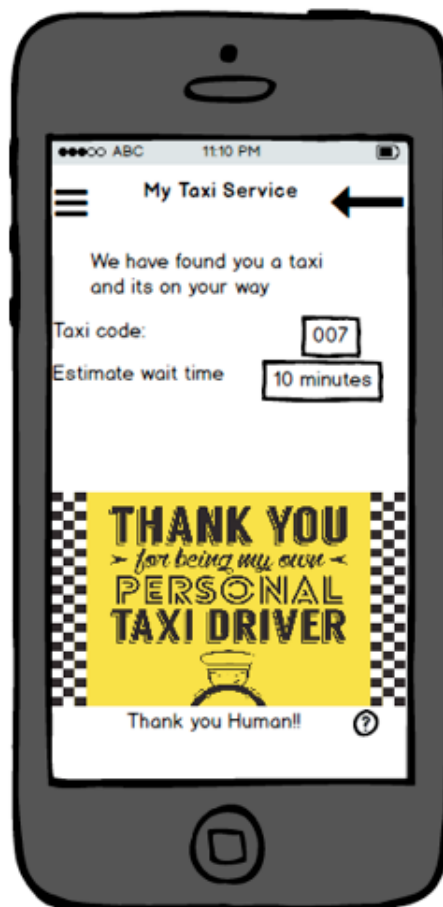
Here the passenger can reserve a taxi. He needs to insert the starting and destination point, the date and the number of passenger.

Confirm reservation



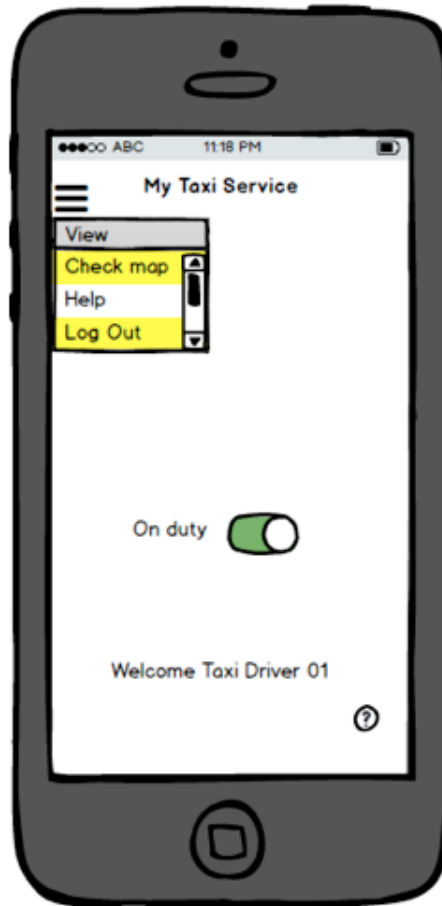
In this page the passenger can show a recap of his trip and confirm the reservation by clicking on the button

Recap reservation



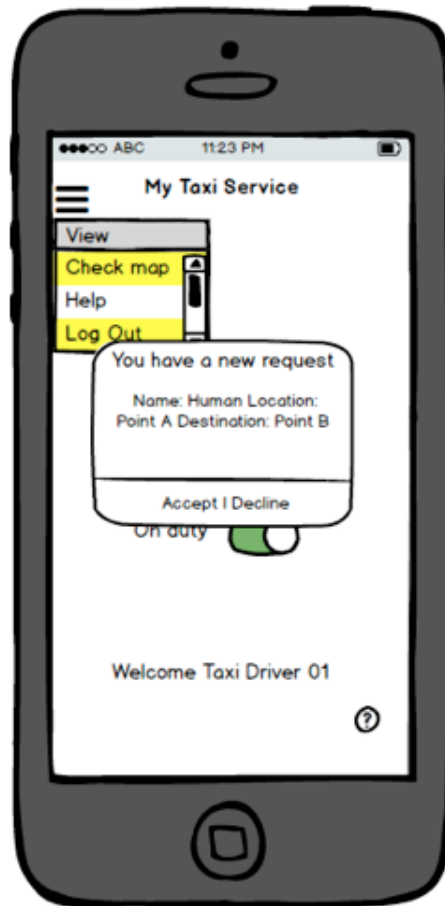
Here the passenger can see the estimate waiting time and the code of the taxi that will pick him up.

Taxi driver homepage



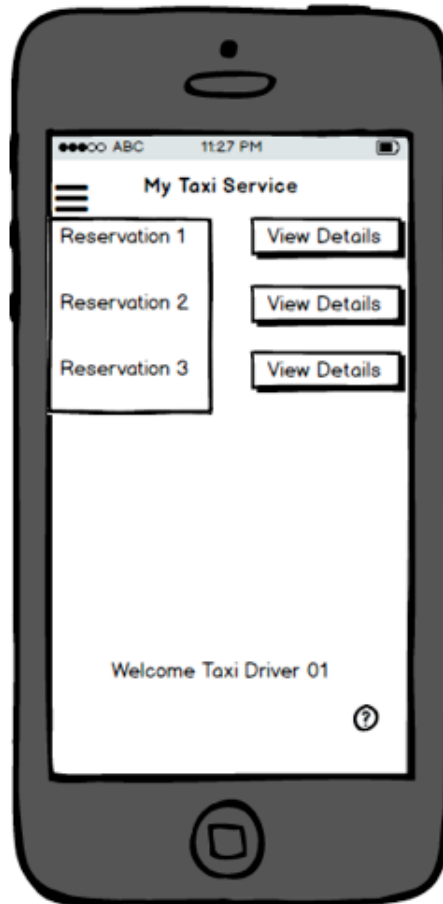
This is the home page for the taxi driver. He can choose from different option such as: Check map, Help or Log out. Also he can change his status in busy or free.

Taxi driver request



This is the notification that the taxi driver will receive when there is a request of a taxi. He can choose if accept or decline this request.

Reservation list for taxi driver



Here the taxi driver can give a look at the reservation he has in list and its details.

Taxi driver map



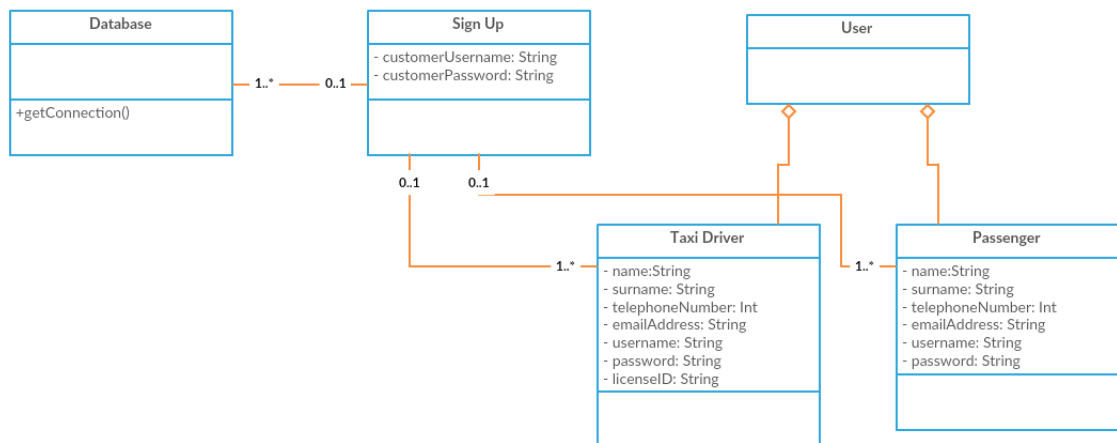
In this page the taxi driver can see the map of the itinerary for the reservation he chooses.

Chapter 5

Requirements traceability

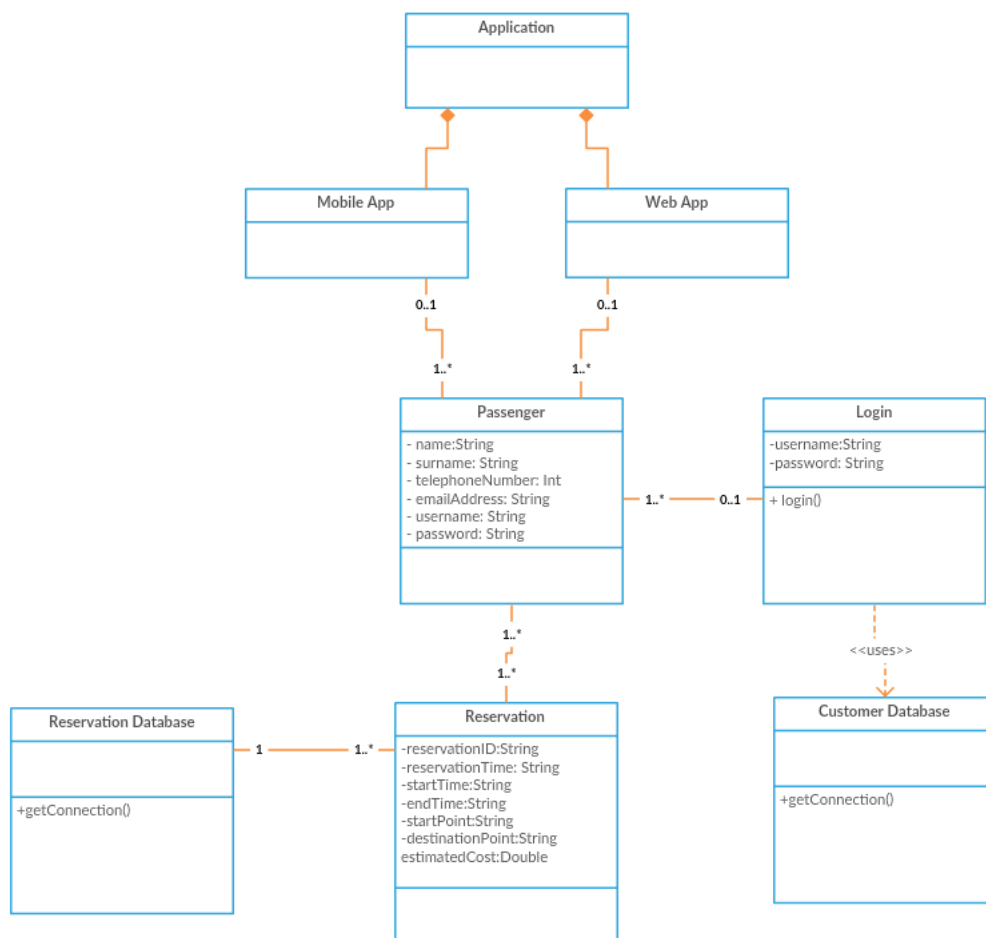
5.1 From requirements to diagrams

1. Allow the registration in the application to the taxi drivers and passengers:
 - The system should be able to provide a sign up functionality in order to create a new taxi driver or passenger profiles.



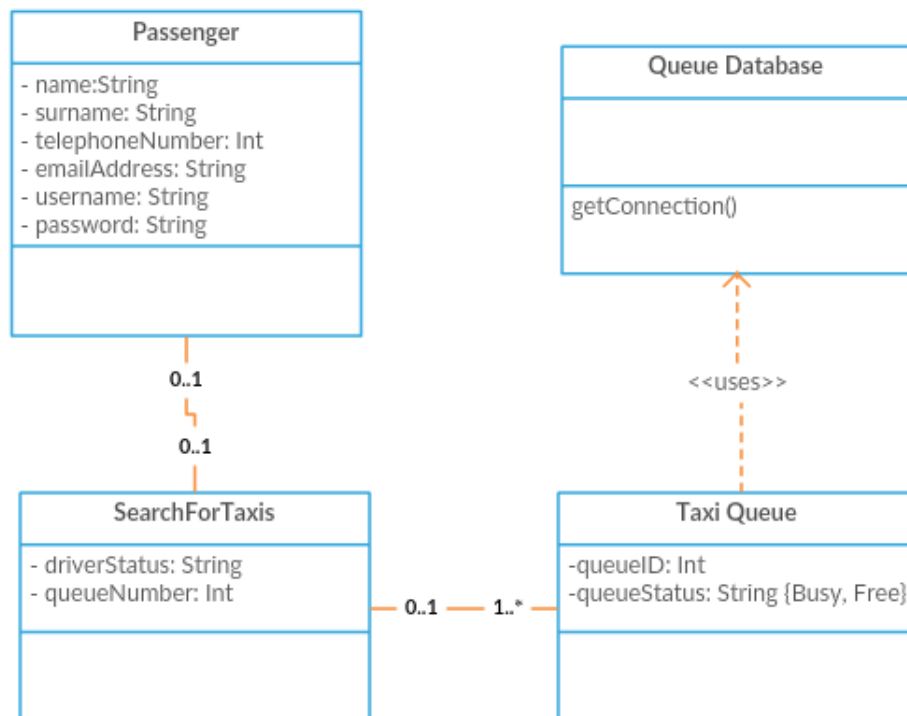
2. Simplify the access of passengers to the taxi service:

- The system has to provide the possibility to the passenger to reserve a taxi
- The system has to provide both a web application and mobile application in order to allow the passenger to save his time.



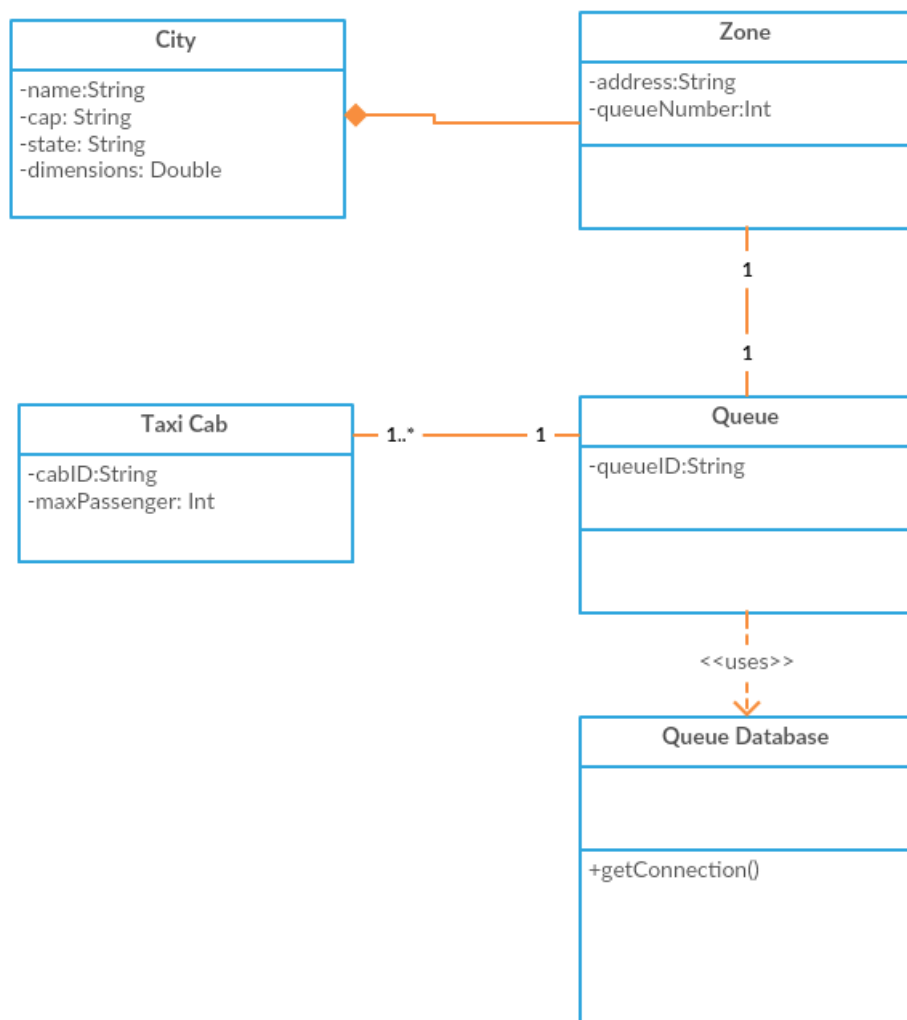
3. Allow passenger to view if there is any taxi driver free:

- The system has to provide a functionality that allow a passenger to view if there is any free taxi driver for the desired route.



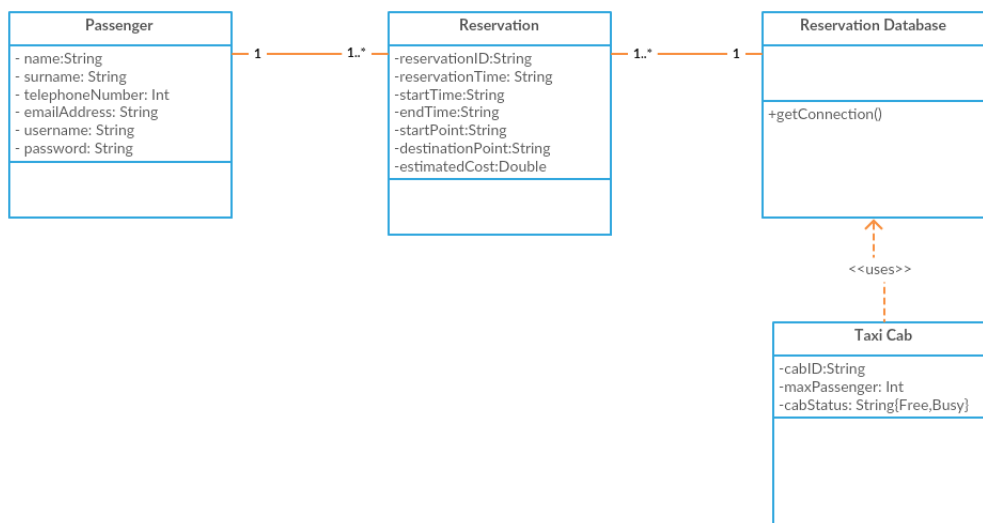
4. Divide the city into different areas covered by some taxi drivers, organized in queues:

- The system has to provide a functionality to divide the city in some areas
- The system has to organize the taxi of these areas into queues



5. Allow the client to reserve taxi for an exact time:

- The system has to provide a functionality that allow the passenger to reserve a taxi for a desired time.



Chapter 6

References

6.1 Software and used Tools

- TexShop (<http://pages.uoregon.edu/koch/texshop/>): to redact this document
- CreatelyApp (<https://creately.com/app/>): to create the class diagrams
- Balsamiq Mockups (<https://balsamiq.com/products/mockups/>): to create the user interface mockup

6.2 Working hours

Dimitar Anastasovski: ~

Marco Colombo: ~