RASD

Giorgio Pea, Andrea Sessa

October 24, 2015

Contents

1 Introduction

1.1 Purpose

This document represent the Requirement Analysis and Specification Document (RASD). The main goal of this document is to completely describe the system in terms of functional and non-functional requirements, to show the constraints and the limit of the software and simulate the typical use cases that will occur after the development. This document is intended to all developer and programmer who have to implement the requirements, to system analyst who want to integrate other system with this one, and could be used as a contractual basis between the customer and the developer.

1.2 Scope of the project

The aim of this project is to develop an application myTaxiService, a web/mobile applications that makes easier and quicker taking taxies. Thanks to MyTaxiService, anyone can request or book a taxi and get realtime information about how long it will take to be picked up or about taxi's current position and code. In addition, MyTaxiService provides an efficient way to allocate taxies by dividing the city in zones and using a queue based allocation system, in order to reduce the waiting time and city's traffic.

1.3 Goals

In this subsection we describe a set of high level goals that myTaxiService is proposed to achieve.

1. Simplify and speed up the process of taking a taxi
   - When a user has entered his taxi ride details and clicks or taps the request button, then MyTaxiService will find the first available taxi that fits for the inserted ride details, booking it to the user
   - When a user has entered his taxi reservation details and clicks or taps the book button, then MyTaxiService will book a taxi that fits for the inserted booking details and for the indicated meeting time
2. Guarantee an efficient and fair management of taxi queues
   - Guarantee a right distribution of taxies in the city
   - Guarantee short taxi availability times and short waiting times

1.4 Glossary

1.4.1 Terms disambiguation

MyTaxiService(F)

The front end of MyTaxiService, that is to say the components of the application that manage the interaction with the user and the logic behind this interaction

MyTexiService(B)
The back end of MyTaxiService, that is to say the components of the application that manage the forwarding of the ride / reservation request with all their associated notifications, the search of available taxies that are compatible with the request/reservation inserted, and other internal tasks not exposed to the user or the taxies

MyTaxiService
MyTaxiService(F) + MyTaxiService(B)

Taxi driver
The person who is licensed to drive a taxi cab.

Taxi(with capital T)
A taxi that uses the MyTaxiService

Request
An electronic message sent by a user through MyTaxiService(F) to MyTaxiService(B). This electronic message refers to the case in which the user wants to be picked up by a taxi asap.

Reservation
An electronic message sent by a user through MyTaxiService(F) to MyTaxiService(B). This electronic message refers to the case in which the user wants to be picked up by a taxi at a specific time.

Zone
An area of the city.

Credentials
A combination of username and password, used by a registered user to access the myTaxiService application.

Taxi ride
A movement of people, through a taxi cab, from one geogrphical point to another

Queue
A data structure managed with a FIFO(First in First Out) policy.

User
A person that wants to take a taxi and is not registered to MyTaxiService.

Registered User
A person that needs to take a taxi and is registered to MyTaxiService.

1.4.2 Acronyms

RASD: Requirements Analysis and Specification Document

FIFO: First In First Out

1.5 Reference Documents

- Specification Document: MyTaxiService-AA2015-2016.pdf
- IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.
- IEEE Std 1016 tm -2009 Standard for Information Tecnology-System Design-Software Design Descriptions.

1.6 Assumptions

1. MyTaxiService has been commissioned by the city local government. Each taxi in the city must have a taxi license provided by the local government. This license requires the registration to the myTaxiService and forces drivers to accept ride request/reservations only using MyTaxiService. During the registration process taxi drivers are asked to provide their personal and vehicle data and work time table. At the end of the registration process a unique code will be assigned to each taxi. Eventually, the city's local government provides each taxi a device. This device is used to see incoming ride requests or reservations and to signal their acceptance

2. Since MyTaxiService is aware of the work timetable of each taxi and GPS data, taxies are considered unavailable if and only if they are serving a ride/reservation request. After a taxi has finished serving a passenger he has to notify that to myTaxiService and so myTaxiService will consider the taxi available again. Considering that myTaxiService knows the position of the taxi and the destination of each ride there is no chance a taxi drive can cheat by not signalling he hasn't yet finished service a passenger.

3. A Taxi might have an accident, if that happens, the taxi driver can report it and so the taxi is considered unavailable.

4. Once a user sends a ride request, he or she cannot change any detail of the request nor can undo the request.

5. MyTaxiService is aware of the characteristics of each taxies (number of passengers)

6. MyTaxiService is aware of all the possible valid location in the city, so the user is forced to select one of them and not insert one of them.

7. Taxi can accept only ride reservation/request within the city borders.

8. Accepting a ride/reservation request by the taxi driver: signaling that the taxi has already left in order to pick a registered user up.

9. Visitor: A generic person that is not registered to the service.

10. Exists an unique application(myTaxiService(F)) both for taxi driver and normal user, myTaxiService(F) give access to the proper set of functionalities according to the login credentials.

1.7 Main Actors

In this section are defined all the actors that interact with the myTaxiService during its operation.

Visitor

A user can only see the login page and complete the registration(which is mandatory to use the service) to be able to access to all the functionality of the application.

Registered User

A registered user can, after a successful login, access all the functionality of the application: it can request/reserve a taxi ride, view the status of older reservations and cancel a already confirmed taxi reservation.

Taxi driver

A taxi driver, after a successful login, is granted to: being notified by the system of an incoming request, notify the system about the conclusion of a taxi ride.

MyTaxiService(B)

This actor represent the back-end part of the myTaxiService software system, its main purpose is to forward request coming from users to a taxi available in the zone, also it has to collect informations relative to the availability of the taxi drivers.

2 Requirements Specification

2.1 Functional Requirements

2.2 Non Functional Requirements

2.3 Constraints

2.4 Jackson-Zave approach

2.5 Scenarios definition

3 UML Diagrams

3.1 U

se Case diagram