Requirements Analysis and Specification Document

Grattarola Daniele, Inajjar Ilyas, Lui Andrea

October 27, 2015

Contents

1 Introduction

1.1 Purpose

The presented document is the Requirements Analysis and Specification document (RASD) for the MyTaxiService platform project.

The main purpose of this document is to analyze the problems that the new system is to solve and the customer's necessities, to identify use cases and actors, to describe the functional and non-functional requirements of the platform along with existing constraints, and to propose an in-depth specification of the system before moving on to the more concrete design phase.

This document is intended for stakeholders, software engineers, and programmers and is to be used as reference throughout the whole development of the system. Each phase of the project must be brought on with this document in mind, and any implementation of the system must be clearly designed to reflect the specifications presented here.

The secondary audience for this document includes system maintainers and developers who wish to integrate the platform's services within their own software.

1.2 Scope

The aim of this project is to create the MyTaxiService platform, a web based information system to manage the taxi service of Pallet Town.

The purpose of the platform is:

1. To provide users with a web application and a mobile application, with which to easily make use of the town's taxi service.

2. To provide taxi drivers with a mobile application, with which to manage customers' requests (submitted through the above mentioned systems)

In short, the platform is to be an integrated infrastructure to manage the initial interaction between customers and taxi drivers, to aid the former in requesting a taxi ride, and to enable the latter in managing such requests.

The platform will therefore be used exclusively to support the town's taxi service, and is aimed at making the taxi request process much efficient by reducing overall costs, communication overhead and potential requests overloads.

The platform will implement all necessary functions on both the users' side and the drivers', while keeping the two aspects separate and unaware of each other. It will also be given a particular focus to the storage of user data, in order to maximize privacy and security, and to the extensibility of the system, in order to make the platform flexible to potential changes in the requirements.

1.3 Definitions, acronyms, and abbreviations

Throughout this document, the following definitions will be applied without further explanations:

- **Platform**: the set of software applications and hardware infrastructure that make up the MyTaxiService service; these include the back-end server software, the web application and the mobile application used by the customers, the mobile application used by the taxi drivers, and all the necessary hardware needed to run the mentioned software and any needed support software.

- **System**: the software run on the back-end server of the platform which is used to handle the communication between the user applications. The term also addresses all the necessary software components that are needed to store data, perform calculations and manage the hardware (e.g. an operating system).

- **User-side application**: software run on a personal device which is used to send taxi requests to the system and to handle the system's replies. It is designed to be used by customers (see below) and can either be a mobile application (run on a smartphone or tablet) or a web application (run on any personal device through an Internet browser).

- **Taxi-side application**: software run on a personal device which is used to manage taxi requests forwarded by the system and to reply to the system with information on how to handle the requests. It is designed to be used by taxi drivers (see below) and is a mobile application (run on a smartphone or tablet).

- **Taxi driver**: the owner of a taxi license in Pallet Town, who uses the taxi-side application to interact with the platform.

- **Customer**: a person which intends to exploit the taxi service of the town, and who uses the user-side applications to interact with the platform.

The following acronyms will also be used in place of the extended form:

- **RASD**: Requirement Analysis and Specification Document

Finally, in order to increase readability the following abbreviations will be used in place of the full words or sentences:

- **e.g.:** *exempli gratia*

1.4 References

This document contains references to other documents which are part of the complete set of deliverables used to describe and design the project. The references are intended as follows:

- **Document 1**: ..

- **Document 2**: ...

1.5 Overview

The presented RASD is divided in sections and structured as follows:

- **Section 1 - Introduction**: contains a high level description of the project, along with its goals, and contains information about the document itself, such as abbreviations used and the document structure.
- **Section 2 - Overall description**:
- **Section 3 - Requirements**:

2 Overall Description

2.1 Product Prespective

The system requirements mainly depends on the web technology necessary to communicate between actors (taxi, customer, guest) and the GPS due to locate the taxi customer in the specific sector and also to create the taxi queue. The central node system need a server (connected on the web) to elaborate the requests and store the users and taxis information. The end-user system relies on the guest operating system (usually a mobile OS like Android, iOS, Windows Phone, blackberry OS, . . .) on the internal GPS system and Internet connection. Of course, all the overall system should only work inside the city.

2.1.1 System Interfaces

- Google Maps API for taxi ruote

2.1.2 User Interface

2.1.3 Hardware Interface

- Ethernet connection
- GPS system
- High fault tolerance of the central node
- Firewall: *server store sensitive data*
- Router

2.1.4 Software Interface

- Apache Server for CGI
- DBMS

2.1.5 Connection Interface

- EDGE, HSDPA, 3G, 4G
- HTML, SOAP/REST protocols

2.1.6 Memory Constrains

- Server: 128GB vRAM requiered, 512GB recommended
- 2TB storage capacity + 64GB flash (32GB x2 mirroring) for applications system and server OS

2.1.7 Operations

- Taxi Operations: login, signup, availability notifications, request reply, report
- User Operations: login, signup, requests operations, user profile operations (show, edit)
- System Operations: notifications (to user and taxi driver), queue managment
- Backup operations (scheduled at 4AM every day)
- Restore operaions from secondary storage

2.1.8 Site adaptation requirements

- UPS system
- 250V/110V Electrical system
- Ethernet connection
- Anti-thief system, safety room (safety grid on each door and eventually windows) and alarm
- Anti-fire system and alaram
- Anti-flooding system
- Air conditioning

2.2 Product Function

2.2.1 Taxi Function

- Taxi Login: the taxi driver logs into the system;
- Taxi Signup: the taxi driver provides his personal information, Username and Password, taxi and driving license to the Operator;
- Notification of Availability: with this operation the taxi driver notifies is presence in a specific zone of the town;
- Notification of Unavailability: the taxi driver notify the end of his service;
- Taking charge: the taxi driver positively reply to a customer request;
- Reject request: the taxi driver negatively reply to a customer request;
- Report user: the taxi driver report a problem with customer;

2.2.2 User Function

- User Login: the user get logged into the system providing his account information;
- User Signup: the user (previously Guest) request a signup to the system providing his personal information;
- Request Taxi: the user request a taxi to his location;
- Book Taxi: the user reserve a taxi to a specific location at the specific time;
- Cancel request: the user cancel the previous request or reservation;
- Report taxi: the user report a problem with the taxi driver or the service;
- Show Profile: the system provides the user information;
- Modify Profile: the user edit his information;

2.2.3 System Function

- Notification to User
  1. Incoming taxi: a taxi take charge of the request;
  2. No taxi available;
- Notification to Taxi driver
  1. Taxi request: a user need a taxi;

2. First of the queue: the taxi is the first of the queue of the current zone;
3. Queue change: notification to the taxi of the new position of the queue;

- Queues management: change the queues after a request reply or new taxi driver in zone;

2.3 User Characteristic

2.3.1 User:

- Low knowledge of mobile app use;
- Age: over 18;

2.3.2 Taxi Driver:

- Owner of taxi license and driver license;
- Low knowledge of mobile app use;
- Good knowledge of the local language;

2.4 Constrains

- GPS Coverage;
- Privacy policy;
- Devices Constrains (presence of GPS and Internet connection);

2.5 Assumption and Dependecies

2.6 Apportioning of requirements