myTaxiService
Software Design Document
Version 1.0

Losio Davide Francesco, Luchetti Mauro, Mosca Paolo

November 19, 2015

Contents

1 — Introduction

This documentation will be used to aid in software development by providing further details of how the software should be built. Within the Software Design Document are narrative and graphical documentation of the software design of the project including use case models, sequence diagrams, and other supporting requirement information.

1.1 Purpose

The purpose of the Software Design Document is to provide a description of MyTaxiService system design and architecture fully enough to allow software development to proceed with an understanding of what is to be built and how it is expected to be built. To achieve this DD(**D**esign **D**ocument) translates and states more accurately the Requirement Specifications described in the MyTaxyService RASD document. It identifies high-level system architecture and design framework as well as hardware, software, communication and interface components.

1.2 Scope

MyTaxiService application is a server/client combination that will allow a user to handle different type of taxi service, keeping track of all the transaction necessary for the completion of each operations. This will include booking a taxi, request a taxi as soon as possible, the handling of the sharing option and, for the taxi drivers, the managing of the taxi queue. All this functionalities will be guaranteed in the way and in the manner explained in the RASD document. Via a Cross Platform Web Environment (by the use of angularJS, Ionic, Cordova and nodeJS frameworks), the MyTaxiService will be able to run on various platforms, including Unix, Linux and Windows based systems, and all the portable devices based on Android and Ios. When a network connection to the server is available, the user will be able to synchronize his PD (**P**ortable **D**evice) or PC with the server, he will be able to log in or register and makes his own request in the case of the passenger-user. Or to set is availability, accepts or rejects request in the case of the taxi-driver user.

Below are stated some main issues with which the system has to be capable to cope with.

- **PD Issues:** Because of memory limitations, a PD will only store data and application parts that are strictly necessary for a PD user. Also, PDs have reduced screen size and limited input capability compared to PCs, so we will design PD standalone functionality in manner that can be easily presented on a typical 240x320 screen.
- **Synchronization:** We will implement server software to serve as an interface between the PC or PD and the Application logic, by the re-using of already existent services offered by third parties company.
- **Transaction and Queue Managing:** all the transactions and booking procedure will be handled by the application logic layer that will be divided from the presentation layer, as well as the queue managing features and algorithm. PDs will implement only presentation layer and connection functionalities.

1.3 Definitions, Acronyms, Abbreviations

- **RASD: R**equirements **S**pecification **A**nalysis **D**ocument
- **DD: D**esign **D**ocument
- **PD: P**ortable **D**evice

1.4 References

- **MyTaxyService RASD** - November/6/2015, edited by Dadoz+Grin-Go+Pol Corporation;

1.5 Document Structure and Overview

- **Architectural Design:** this section is the main focus of this document. It provides an overview of the system's major components and architecture, as well as architectural styles, pattern used and other design decision.

  a detailed analysis of modules will also describe lower-level classes, components, and functions, as well as the interaction between these internal components.

- **Algorithm design:** this section is focused on the definition of the most relevant algorithm of the project.
- **User interface design:** this section provide an overview on how the user interface(s) of the system will look like. In particular, referring on what has already stated in the RASD, here some further details is specified.
- **Requirements Traceability:** this section explain how the requirements defined in the RASD map into the design elements that are defined in this document.

2 — Architectural Design