POLITECNICO
MILANO 1863

Politecnico di Milano
A.A. 2015/2016
Software Engineering 2: "myTaxiService"

Alessandro Baldassari (mat. 841561)
Alberto Bendin (mat. 841734)
Francesco Giarola (mat. xxxxxx)

October 24, 2015

Contents

Page

1 Introduction 1
- 1.1 Purpose 1
- 1.2 Scope 1
- 1.3 Actors 1
- 1.4 Goals 2
- 1.5 Definitions, acronyms, and abbreviations 3
  - 1.5.1 Definitions 3
  - 1.5.2 Acronyms 3
  - 1.5.3 Abbreviations 3
- 1.6 Reference Documents 3
- 1.7 Overview 3

2 Overall description 4
- 2.1 Product perspective 4
- 2.2 Product functions 4
- 2.3 User characteristics 4
- 2.4 Constraints 4
- 2.5 Assumptions and dependencies 4

3 Specific requirements 4
- 3.1 External Interface Requirements 4
  - 3.1.1 User Interfaces 4
  - 3.1.2 Hardware Interfaces 4
  - 3.1.3 Software Interfaces 4
  - 3.1.4 Communication Interfaces 4
- 3.2 Functional Requirements 4
  - 3.2.1 User Class 1 4
  - 3.2.2 User Class 2 4
- 3.3 Performance Requirements 4
- 3.4 Design Constraints 4
  - 3.4.1 Standards compliance 4
  - 3.4.2 Hardware limitations 4
- 3.5 Software System Attributes 4
  - 3.5.1 Reliability 4
  - 3.5.2 Availability 4
  - 3.5.3 Security 4

3.5.4 Maintainability ... 4
3.5.5 Portability ... 4
3.6 Other Requirements ... 4

1 Introduction

1.1 Purpose

The purpose of this document is to describe in a complete and sound way the *MyTaxiService* application that will be developed and the application domain in which it will run.
The intended audience for this document are the developers and programmers who have to implement the application, system and requirement analysts who want to integrate *MyTaxiService* with their system or software, testers who have to determine whether the requirements have been satisfied in the application implementation, projects managers who have to plan, estimate and control the analysis and development processes and finally the users themselves. This document could be used as a contractual agreement between the costumer and the entity who develops the application.

1.2 Scope

MyTaxiService is a new web and mobile application conceived to provide an immediate and user-friendly access to the taxi service of a large city; it aims at an overall improvement of the quality of the service offered.
This optimization is obtained thanks to the real-time interaction and feedback of all the parties involved in the service: taxi passengers can choose and book the ride, and the system will forward the request to the nearest available taxi drivers who can decide to take over the call; in this case the system will notify the client with the code of the incoming taxi and the waiting-time. The system guarantees a fair management of taxi queues. In particular, the city is divided in taxi zones and each zone is associated with its taxi queue. The system automatically computes the distribution of taxis in the various zones based on the GPS information it receives from each taxi. When a taxi is available, its identifier is stored in the queue of taxis in the corresponding zone. When a request arrives from a certain zone, the system forwards it to the taxis in the corresponding zone according to their order in the queue. Additional features of the application are the possibility for the passengers to reserve a ride at least two hours in advance, choosing the origin and destination, and the option to possibly share the ride with someone else, thus dividing the cost of the service.

1.3 Actors

- Clients: are the final users the taxi service is offered to. They can book the ride choosing among different options, for instance date and time, origin and destination places and the possibility of sharing the trip with other customers.

- Taxi Drivers: represent the other category of users of the application, they can accept a call for a service or turn it down, thus allowing the whole system to be synchronized, fast and efficient; moreover the system keeps the coordinates of the taxis automatically updated.

1.4 Goals

List of the goals of *MyTaxiService* application for taxi passengers:

- [G1] The user can ask for a taxi simply providing his/her own position.
- [G2] The user can plan the trip and preview the fare of the ride and decide whether to call a taxi.
- [G3] The user can customize the reservation, specifying the date and time of the ride, the origin and destination, the willingness to share the ride.
- [G4] The user can delete a reservation for a taxi. (???maybe only with a constraint eg. 30 minutes in advance, there is however a booking fee??)
- [G???????] Multiple goals: The user can sign up and login to the service, become a registered user and keep track of the favorite routes, favorite payment method????Link credit card???? Do we need this?? Maybe to pay the fee in case the user decides to delete a reservation???? Mandatory registration or possibility of unregistered user???

List of the goals of *MyTaxiService* application for taxi drivers:

- [G5] The user is notified by the system when there is a client nearby waiting for a taxi.
- [G6] The user can take in charge or reject the requests received as notification.
- [G7] The user can cancel a service that has already taken in charge and notify the system, specifying the reason for the emergency (eg. engine failure).
- [G???????] Multiple goals: the user can sign up and login to the service, become a registered driver, binding his/her taxi license and taxi number to the user, maybe used to control the taxi fleet, plan maintenance...??????????

Other goals of *MyTaxiService* application:

- [G8] The system is able to map the requests of the clients according to their location.

- [G9] The system is able to map the position of the taxi fleet and assign each taxi to a predetermined zone of the city according to its position.
- [G10] The system is able to control the queue of taxis in every zone and enforce the predetermined priority rules.

1.5 Definitions, acronyms, and abbreviations

1.5.1 Definitions

- Client (or Customer, Taxi passenger): is the user of the application that wants to use the taxi service.
- Taxi driver (or Taxi owner): is the user of the application that together with the back-end system makes the service functional and constantly updated, he/she controls the work which is assigned to himself/herself accepting or rejecting the proposals of clients that the system forwards.

1.5.2 Acronyms

- RASD: Requirements Analysis and Specification Document.

1.5.3 Abbreviations

- [Gn]: n^{th} goal.
- [Rn]: n^{th} functional requirement.
- [Dn]: n^{th} domain assumption.

1.6 Reference Documents

- Specification document: MyTaxiService project
- IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.

1.7 Overview

- Section 1: Introduction, it gives a brief description of the purpose, functionalities and goals of the application.
- Section 2: Overall Description, focuses more in-depth on features of the software, constraints and assumptions.
- Section 3: Specific Requirements, this part lists requirements, typical scenarios and use cases, together with UML diagrams to provide a more easy-to-read insight at the several functionalities of the software.

2 Overall description

2.1 Product perspective

2.2 Product functions

2.3 User characteristics

2.4 Constraints

2.5 Assumptions and dependencies

3 Specific requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

3.1.2 Hardware Interfaces

3.1.3 Software Interfaces

3.1.4 Communication Interfaces

3.2 Functional Requirements

3.2.1 User Class 1

3.2.2 User Class 2

3.3 Performance Requirements

3.4 Design Constraints

3.4.1 Standards compliance

3.4.2 Hardware limitations

3.5 Software System Attributes

3.5.1 Reliability

3.5.2 Availability

3.5.3 Security

3.5.4 Maintainability

3.5.5 Portability

3.6 Other Requirements