

Software Engineering 2 “myTaxiService”

Project Plan Version 0.1

1/21/2016

Politecnico di Milano A.A. 2015-2016

Milica Jovanovic (mat. 835953);Pavle Vidanovic (mat. 854472)

Contents

[1 Introduction 4](#_Toc441142475)

[1.1 Revision History 4](#_Toc441142476)

[1.2 Purpose and Scope 4](#_Toc441142477)

[1.3 Definitions and Abbreviations 5](#_Toc441142478)

[1.4 Reference Documents 5](#_Toc441142479)

[1.5 Document Overview 5](#_Toc441142480)

[2 Integration Strategy 7](#_Toc441142481)

[2.1 Entry Criteria 7](#_Toc441142482)

[2.2 Elements to be integrated 8](#_Toc441142483)

[2.3 Integration Testing Strategy 9](#_Toc441142484)

[2.4 Sequence of Component/Function Integration 10](#_Toc441142485)

[2.4.1 Software Integration Sequence 10](#_Toc441142486)

[2.4.1.1 Integration Test of Guest Component 10](#_Toc441142487)

[2.4.1.2 Integration Test of Admin Component 10](#_Toc441142488)

[2.4.1.1 Integration Test of TaxiDriver Component 10](#_Toc441142489)

[2.4.1.2 Integration Test of Scheduler Component 10](#_Toc441142490)

[2.4.1.3 Integration Test of User Component 10](#_Toc441142491)

[2.4.2 Subsystem Integration Sequence 11](#_Toc441142492)

[3 Individual Steps and Test Description 12](#_Toc441142493)

[3.1 Test case specification 12](#_Toc441142494)

[3.1.1 Integration test case I1 12](#_Toc441142495)

[3.1.2 Integration test case I2 12](#_Toc441142496)

[3.1.3 Integration test case I3 12](#_Toc441142497)

[3.1.4 Integration test case I4 13](#_Toc441142498)

[3.1.5 Integration test case I5 13](#_Toc441142499)

[3.1.6 Integration test case I6 13](#_Toc441142500)

[3.1.7 Integration test case I7 15](#_Toc441142501)

[3.2 Test procedures 16](#_Toc441142502)

[3.2.1 Integration test procedure TP1 16](#_Toc441142503)

[3.2.2 Integration test procedure TP2 16](#_Toc441142504)

[3.2.3 Integration test procedure TP3 16](#_Toc441142505)

[3.2.4 Integration test procedure TP4 16](#_Toc441142506)

[3.2.5 Integration test procedure TP5 17](#_Toc441142507)

[4 Tools and Test Equipment Required 18](#_Toc441142508)

[4.1 Manual 18](#_Toc441142509)

[4.2 Automatic Test 18](#_Toc441142510)

[5 Program Stubs and Test Data Required 18](#_Toc441142511)

[6 References 18](#_Toc441142512)

# Introduction

## Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Data | Authors | Summary |
| 0.1 | 13/01/2016 | Pavle Vidanovic  Milica Jovanovic | Initial Draft |
| 1.0 | 20/01/2016 | Pavle Vidanovic  Milica Jovanovic | Final Version |

## Purpose and Scope

The purpose of this document is to define plan for testing, integration testing and verifying that system development during the project complies with the requirements of Requirement document and Design Document. This document also presents test results in order to determinate if the application meets predetermined requirements and functionalities.

The aim of this project is to develop and implement myTaxiService, an application similar to Uber, which makes the process of assigning an available taxi vehicle to possible passengers.

The developed system should allow new users to register. Users, once logged in, should be able to:

* request a taxi
* reserve a taxi
* cancel a ride
* check taxi availability around him
* receive a confirmation with information about the assigned vehicle and ETA once taxi is requested
* create/maintain user profile
* report a taxi driver

The developed system should allow new taxi drivers to register. Drivers, once logged in, should be able to:

* inform the system about their availability
* confirm/decline that they are going to take care of a certain call
* create/maintain taxi driver profile
* report a passenger

The system should keep information about new arrived requests, as well as the confirmed rides. A ride should have and id number, information about the passenger that requested the ride, as well as the code of the assigned vehicle and ETA. System should also keep information about taxi queues connected to particular zone of the city and ensure fair management of the queues. Developed system should keep information about the list of reservations made by passengers, such as id number of the reservation, information about the passenger that made the reservation and the time of reservation and time of the ride.

## Definitions and Abbreviations

|  |  |
| --- | --- |
| *ETA* | Estimated Time of Arrival, approximated time of arrival of taxi vehicle to destination |
| *Reservation* | Passenger request for a vehicle at least 2 hours before the ride |
| *Request* | Passenger filled form for immediate ride |
| *Reservation* *Conformation* | Notification sent to the user about the confirmed reservation |
| *Ride Conformation* | Notification sent to the user about the confirmed ride with information of the ride |
| *Report* | Short description of problem that user/driver stumped into |
| *User* | A person already registered and logged into the system |
| *Guest* | A person accessing a system that has either never registered or hasn't logged in yet. Guest has only two available options, to log in or to register for the first time |
| *Taxi* *driver* | A person already register and logged into the system as a driver |
| *GPS* | Global Positioning System |
| *API* | *:* Application Programming Interface*.* |
| *DD* | Design Document |
| *DB* | Database |
| *DBMS* | |  | | --- | | Database Management System | |
| *RASD* | Requirement Analysis and Specification Document |
| *ITPD* | Integration Test Plan Document |

## Reference Documents

* RASD - RASD myTaxiService - final v2.0
* DD - DD myTaxiService - final
* Specification Document: myTaxiService Project AA 2015-2016
* Assignments 1 and 2 (RASD and DD)
* Assignment 4 - integration test plan
* Integration Test Plan Example

## Document Overview

The document is essentially structured in six parts:

* Chapter 1: Introduction, gives description of document and some basic information about the software
* Chapter 2: Integration Strategy, gives an overview of entry criteria for the integrating components and how the elements will be integrated as well as used testing strategy and  sequences of component/function integration
* Chapter 3: Individual Steps and Test Description, description of type of tests for verifying elements defined in one step, verifying the results are as expected
* Chapter 4: Tool and Test Equipment Required, overview on tools and equipment used to support integration test
* Chapter 5: Program Stubs and Test Data Required, gives an overview of how the requirements defined in RASD map into the design elements defined in DD.
* Chapter 6: References

# Function Points

## Brief Introduction

## FP Estimation

### Internal Logic Files

### External Logic Files

### External Inputs

### External Inquiries

### External Outputs

### Resuming

## Evaluation of Estimation

# Effort Estimation COCOMO II

## Brief Introduction

## Scale Drivers

## Cost Drivers

## Effort Equation

## Schedule Estimation

Functions that need to pass Unit testing are entry criteria for following components of myTaxiService System:

|  |  |
| --- | --- |
| Component | Functions to be unit tested |
| Guest Manager | * signUp() * signIn() |
| User Manager | * makeRequest() * makeReservation() * report() * manageProfile() * checkTaxisAvailable() * checkReservation() * cancelRide() |
| TaxiDriver Manager | * confirmDeclineRide() * setAvailable() * manageProfile() * report() * cancelRide() * checkRides() |
| Admin Manager | * banUser() * viewReports() * signIn() |
| Scheduler |  |
| Request Manager | * createRequest() * provideTaxi() * calculateETA() * sendConfirmation() * findZone() * findDriver() * rideProposal() |
| Reservation Manager | * createReservation() * findDriver() * findZone() * sendConfirmation() * reservationConfirmation() * rideProposal() |
| Zone Manager | * determineZone() //getZone * findAvailableDriver() * enqueDriver() * dequeueDriver() * peekDriverOnQueue() |

# 

# References

* Slides of the Software Engineering 2 course (Beep platform)
* Testing in Software Development, Martyn A. Ould, Charles Unwin, British Compute
* Official Cucumber website
* [www.watirwebdriver.com](http://www.watirwebdriver.com) – Watir-Webdriver