

**I**ntegration **T**est **P**lan **D**ocument **(ITPD)**

Computer Science and Engineering (CSE)

Software Engineering 2 Project

Year 2015/16

*Date: 21/01/2016 – Version: 1.0*



**STUDENTS:**

*Martino Andrea (****788701****)*

*Marchesani Francesco (****852444****)*

**PROFESSOR:**

*Mirandola Raffaela*

# Introduction

## Table of contents

1. Introduction 2

1.0. Table of contents 2

1.1. Revision History 3

1.2. Purpose and Scope 3

1.3. List of Definitions and Abbreviations 4

1.4. List of Reference Documents 5

2. Integration Strategy 5

2.1. Entry Criteria 5

2.2. Elements to be integrated 5

2.3. Integration Testing Strategy 6

2.4. Sequence of Component/Function Integration 6

3. Individual Steps and Text Description 6

4. Tools and Test Equipment Required 6

5. Program Stubs and Test Data Required 7

6. Appendix 7

6.1. Hours of work 7

## Revision History

We will keep the **revision history** of the **Integration Test Plan Document** (**ITPD**) in this chapter.

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author(s)** | **Summary** |
| 1.0 | 21/01/2016 | Andrea Martino, Francesco Marchesani | Document  Creation |

## Purpose and Scope

This **Integration Test Plan Document** (**ITPD**) contains information about the **test plan**, the **integration strategy** and **other required material** of *myTaxiService*.

This **document** is coherent with *the official template* of the project on the *Beep platform* (see *Assignment 4 – integration test plan.pdf*).

As we said for the **RASD** and the **DD**, it is important to underline that some parts of this document may evolve in the future (this may occurs for several causes).

Anyway, we will try to maintain coherence as much as possible.

Here is a resume of the steps of the project, with the related deadlines (in green documents already delivered, in yellow the current document):

The main scope of this **ITPD** (*Integration Test Plan Document*) is to give an overall guidance to the **testing phase** of the **project**, which is *myTaxiDriver* (**Software Engineering 2 project** of year 2015/16 - **Politecnico di Milano**).

* We described the main **goals** and **objectives** of the project in the previous *Requirements Analysis and Specification Document*.
* We also specified the **general architecture**, with the **components** of the system in the other previous document (*Design Document*).

## List of Definitions and Abbreviations

* **RASD**: *Requirements Analysis and Specification Document*
* **DD**: *Design Document*
* **ITPD**: *Integration Test Plan Document*
* **mTS**: *myTaxiService*
* **SE**: *Software Engineering*
* **IDE**: Integrated Development Environment
* **mockito**: tool for mockups creation (useful in *Unit Testing*).
* **Unit Testing (UT)**: a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use.
* **Integration Testing (IT)**: is the phase in software testing in which individual software modules are combined and tested as a group.
* **System Testing**: is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.
* **Performance Testing**: is a testing practice performed to determine how a system performs in terms of responsiveness and stability under a particular workload.
* **Load Testing**: is the process of putting demand on a software system or computing device and measuring its response.
* **Stress Testing**: is a test that put a greater emphasis on robustness, availability, and error handling under a heavy load, rather than on what would be considered correct behavior under normal circumstances.
* **Test Plan**: is a document detailing the objectives, target market, internal beta team, and processes for a specific beta test for a software product.

**Other stuff, of course!**

**Note:** *for the full Glossary may be helpful to see also the paragraph 1.5 of the RASD 2.0 and paragraph 2.3 of DD.*

## List of Reference Documents

Here is a list of the **reference documents** for the *Integration Test Plan Document* of *myTaxiService*:

* **Project Description** (from *Beep*)
* **RASD 2.0** (hosted on *GitHub Repository*)
* **DD** (hosted on *GitHub Repository*)
* /\* Documentation of other tools used for testing \*/

# Integration Strategy

## Entry Criteria

It is important to underline the **entry criteria** before the application of the integration testing process. This is a list of the required entry criteria, with respect to *myTaxiService* project:

* Functions must been have **unit tested**, otherwise there is a high probability of issues with the standalone units (without looking at their interactions, as target of the integration tests).
* The code has a **proper documentation**, in order to be readable from the point of view of the testers. Otherwise, they may occur problems with the readability of some parts of the code.
* All the **required tools** are available and work without problems.
* OTHERS?

## Elements to be integrated

## Integration Testing Strategy

Mix Top-Down / Bottom-Up?

## Sequence of Component/Function Integration

### Software Integration Sequence

### Subsystem Integration Sequence

# Individual Steps and Text Description

# Tools and Test Equipment Required

In this chapter, we will show the **tools** and the **test equipment** required for the *Integration Testing*.

Note that this section regards only the Integration Testing. In fact, we will not talk about other useful tools like **mockito**,which are used in the **Unit Testing** **phase** (before the Integration Testing, as said in *Chapter 2.1 Entry Criteria*). In addition, we will also not talk about tools related to other types of testing, such as *System Testing*, *Performance Testing*, *Load Testing*, *Stress Testing* and so on.

We used the following tools:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Logo** | **Website** | **Function** |
| **arquillian** |  | *http://arquillian.org/* | Integration testing framework for containers |
| **JUnit** |  | *http://junit.org/* | Framework to write repeatable tests |
| **NetBeans** |  | *https://netbeans.org/* | IDE for manual integration testing |

# Program Stubs and Test Data Required

# Appendix

## Hours of work

* **Andrea Martino**: Hours
* **Francesco Marchesani**: Hours