

POLYTECHNIC UNIVERSITY OF MILAN

School of Industrial and Information Engineering

Computer Science and Engineering



**Project of Software Engineering 2: MyTaxi
Service
Integration Test Plan Document**

Course Professor: Prof. Elisabetta DI NITTO

Authors:

Mattia CRIPPA 854126

Francesca GALLUZZI 788328

Marco LATTARULO 841399

Academic Year 2015–2016

Contents

1	Introduction	6
1.1	Revision History	6
1.2	Purpose	6
1.3	Scope	6
1.4	List of Definitions and Abbreviations	7
1.5	List of Reference Documents	7
1.6	Document Overview	7
2	Integration Strategy	8
2.1	Entry Criteria	8
2.2	Elements to be Integrated	8
2.3	Integration Testing Strategy	8
2.4	Sequence of Component / Function Integration	8
3	Individual Steps & Test Description	9
3.1	Test case specifications	9
3.1.1	Integration test case I1	9
3.1.2	Integration test case I2	10
3.1.3	Integration test case I3	10
3.1.4	Integration test case I4	10
3.1.5	Integration test case I5	11
3.1.6	Integration test case I6	11
3.1.7	Integration test case I7	13
3.1.8	Integration test case I8	15
3.1.9	Integration test case I9	17
3.1.10	Integration test case I10	18

3.1.11	Integration test case I11	18
3.1.12	Integration test case I12	19
3.1.13	Integration test case I13	19
3.1.14	Integration test case I14	19
3.2	Test procedures	20
3.2.1	Test procedure TP1	20
3.2.2	Test procedure TP2	20
3.2.3	Test procedure TP3	21
3.2.4	Test procedure TP4	21
3.2.5	Test procedure TP5	22
4	Tools & Test Equipment Required	23
5	Program Stubs & Test Data Required	24

List of Figures

List of Tables

Chapter 1

Introduction

Integration testing confirms that each piece of the application interacts as designed and that all functionality is working. Integration testing includes interactions between all layers of an application, including interfaces to other applications, as a complete end-to-end test of the functionality. The development team will be responsible for the creation of the integration test scripts in accordance to the integration test plan. A developer will be chosen by the team who will be responsible for execution of the test scripts and certifying that the integration testing is complete.

1.1 Revision History

Version 1.0, date 21/01/2016

1.2 Purpose

The purpose of the integration test plan is to describe the necessary tests to verify that all of the components of MyTaxiService are properly assembled. Integration testing ensures that the unit-tested modules interact correctly.

1.3 Scope

This document refers to the developing of an application called MyTaxiService, which is aimed to improve the quality and the efficiency of the taxi service of a large city by using localization, smartphones and IT technologies.

1.4 List of Definitions and Abbreviations

RASD: Requirement Analysis and Specification Document

ITPD: Integration Test Plan Document

IT: Integration Test

TP: Test Procedure

1.5 List of Reference Documents

List of all reference documents:

- Project Description: Assignments 1 and 2 (RASD and DD).pdf
- RASD: RASD v2.0-CrippaGalluzziLattarulo.pdf
- Design Document: DesignDocument v1.0-CrippaGalluzziLattarulo.pdf
- ITPD: Assignment 4 - integration test plan.pdf

1.6 Document Overview

This document is essentially structured in five parts:

- **Section 1** → Introduction: defines the purpose, the scope and an overview of this document.
- **Section 2** → Integration Strategy: defines all the items to be tested and the integration testing approach.
- **Section 3** → Individual Steps and Test Description: describes the type of test that will be used to verify that the elements perform as expected.
- **Section 4** → Tools and Test Equipment Required: defines all tools and test equipment needed to accomplish the integration.
- **Section 5** → Program Stubs and Test Data Required: defines any program stubs or special test data required.

Chapter 2

Integration Strategy

2.1 Entry Criteria

2.2 Elements to be Integrated

2.3 Integration Testing Strategy

2.4 Sequence of Component / Function Integration

ID	Integration Test	Paragraphs
I1	Visitor Manager \rightarrow Visitor managedBean	9.9
I2	comp1 \rightarrow comp2	9.9

Chapter 3

Individual Steps & Test Description

3.1 Test case specifications

3.1.1 Integration test case I1

Test Case Identifier	I1T1
Test Item	Visitor managedBeans → Visitor Manager
Input Specification	Create typical Visitor managedBeans input
Output Specification	Check if the correct methods are called in the Visitor Manager
Environmental Needs	Client driver

3.1.2 Integration test case I2

Test Case Identifier	I2T1
Test Item	Passenger managedBeans → Passenger Manager
Input Specification	Create typical Passenger managedBeans input
Output Specification	Check if the correct methods are called in the Passenger Manager
Environmental Needs	Client driver

3.1.3 Integration test case I3

Test Case Identifier	I3T1
Test Item	TaxiDriver managedBeans → TaxiDriver Manager
Input Specification	Create typical TaxiDriver managedBeans input
Output Specification	Check if the correct methods are called in the TaxiDriver Manager
Environmental Needs	Client driver

3.1.4 Integration test case I4

Test Case Identifier	I4T1
Test Item	Developer managedBeans → Developer Manager
Input Specification	Create typical Developer managedBeans input
Output Specification	Check if the correct methods are called in the Developer Manager
Environmental Needs	Client driver

3.1.5 Integration test case I5

Test Case Identifier	I5T1
Test Item	Passenger Manager → Calls Manager
Input Specification	Create typical Passenger Manager input
Output Specification	Check if the correct methods are called in the Calls Manager
Environmental Needs	I2 succeeded

Test Case Identifier	I5T2
Test Item	TaxiDriver Manager → Calls Manager
Input Specification	Create typical TaxiDriver Manager input
Output Specification	Check if the correct methods are called in the Calls Manager
Environmental Needs	I3 succeeded

3.1.6 Integration test case I6

Test Case Identifier	I6T1
Test Item	Passenger Manager → Ride Manager
Input Specification	Create typical Passenger Manager input
Output Specification	Check if the correct methods are called in the Ride Manager
Environmental Needs	I2 succeeded

Test Case Identifier	I6T2
Test Item	TaxiDriver Manager → Ride Manager
Input Specification	Create typical TaxiDriver Manager input
Output Specification	Check if the correct methods are called in the Ride Manager
Environmental Needs	I3 succeeded

Test Case Identifier	I6T3
Test Item	Calls Manager → Ride Manager
Input Specification	Create typical Calls Manager input
Output Specification	Check if the correct methods are called in the Ride Manager
Environmental Needs	I2, I3, I5 succeeded

3.1.7 Integration test case I7

Test Case Identifier	I7T1
Test Item	Visitor Manager → Java Persistence
Input Specification	Create typical Visitor Manager input
Output Specification	Check if the correct methods are called in the Persistence Module
Environmental Needs	I1 succeeded

Test Case Identifier	I7T2
Test Item	Passenger Manager → Java Persistence
Input Specification	Create typical Passenger Manager input
Output Specification	Check if the correct methods are called in the Persistence Module
Environmental Needs	I2 succeeded

Test Case Identifier	I7T3
Test Item	TaxiDriver Manager → Java Persistence
Input Specification	Create typical TaxiDriver Manager input
Output Specification	Check if the correct methods are called in the Persistence Module
Environmental Needs	I3 succeeded

Test Case Identifier	I7T4
Test Item	Developer Manager → Java Persistence
Input Specification	Create typical Developer Manager input
Output Specification	Check if the correct methods are called in the Persistence Module
Environmental Needs	I4 succeeded

Test Case Identifier	I7T5
Test Item	Calls Manager → Java Persistence
Input Specification	Create typical Calls Manager input
Output Specification	Check if the correct methods are called in the Persistence Module
Environmental Needs	I2, I3 and I5 succeeded

Test Case Identifier	I7T6
Test Item	Ride Manager → Java Persistence
Input Specification	Create typical Ride Manager input
Output Specification	Check if the correct methods are called in the Persistence Module
Environmental Needs	I2, I3, I5 and I6 succeeded

3.1.8 Integration test case I8

Test Case Identifier	I8T1
Test Item	Java Persistence → Visitor Manager
Input Specification	Create typical Java Persistence input
Output Specification	Check if the correct methods are called in the Visitor Manager
Environmental Needs	Database Driver

Test Case Identifier	I8T2
Test Item	Java Persistence → Passenger Manager
Input Specification	Create typical Java Persistence input
Output Specification	Check if the correct methods are called in the Passenger Manager
Environmental Needs	Database Driver

Test Case Identifier	I8T3
Test Item	Java Persistence → TaxiDriver Manager
Input Specification	Create typical Java Persistence input
Output Specification	Check if the correct methods are called in the TaxiDriver Manager
Environmental Needs	Database Driver

Test Case Identifier	I8T4
Test Item	Java Persistence → Developer Manager
Input Specification	Create typical Java Persistence input
Output Specification	Check if the correct methods are called in the Developer Manager
Environmental Needs	Database Driver

Test Case Identifier	I8T5
Test Item	Java Persistence → Calls Manager
Input Specification	Create typical Java Persistence input
Output Specification	Check if the correct methods are called in the Calls Manager
Environmental Needs	Database Driver

Test Case Identifier	I8T6
Test Item	Java Persistence → Ride Manager
Input Specification	Create typical Java Persistence input
Output Specification	Check if the correct methods are called in the Ride Manager
Environmental Needs	Database Driver

3.1.9 Integration test case I9

Test Case Identifier	I9T1
Test Item	Ride Manager → Passenger Manager
Input Specification	Create typical Ride Manager input
Output Specification	Check if the correct methods are called in the Passenger Manager
Environmental Needs	I8 succeeded

Test Case Identifier	I9T2
Test Item	Ride Manager → TaxiDriver Manager
Input Specification	Create typical Ride Manager input
Output Specification	Check if the correct methods are called in the TaxiDriver Manager
Environmental Needs	I8 succeeded

Test Case Identifier	I9T3
Test Item	Ride Manager → Calls Manager
Input Specification	Create typical Ride Manager input
Output Specification	Check if the correct methods are called in the Calls Manager
Environmental Needs	I8 succeeded

3.1.10 Integration test case I10

Test Case Identifier	I10T1
Test Item	Calls Manager → Passenger Manager
Input Specification	Create typical Calls Manager input
Output Specification	Check if the correct methods are called in the Passenger Manager
Environmental Needs	I8 and I9 succeeded

Test Case Identifier	I10T2
Test Item	Calls Manager → TaxiDriver Manager
Input Specification	Create typical Calls Manager input
Output Specification	Check if the correct methods are called in the TaxiDriver Manager
Environmental Needs	I8 and I9 succeeded

3.1.11 Integration test case I11

Test Case Identifier	I11T1
Test Item	Visitor Manager → Visitor managedBeans
Input Specification	Create typical Visitor Manager input
Output Specification	Check if the correct methods are called in the Visitor managedBeans
Environmental Needs	I8 succeeded

3.1.12 Integration test case I12

Test Case Identifier	I12T1
Test Item	Passenger Manager → Passenger managedBeans
Input Specification	Create typical Passenger Manager input
Output Specification	Check if the correct methods are called in the Passenger managedBeans
Environmental Needs	I8, I9 and I10 succeeded

3.1.13 Integration test case I13

Test Case Identifier	I13T1
Test Item	TaxiDriver Manager → TaxiDriver managedBeans
Input Specification	Create typical TaxiDriver Manager input
Output Specification	Check if the correct methods are called in the TaxiDriver managedBeans
Environmental Needs	I8, I9 and I10 succeeded

3.1.14 Integration test case I14

Test Case Identifier	I14T1
Test Item	Developer Manager → Developer managedBeans
Input Specification	Create typical Developer Manager input
Output Specification	Check if the correct methods are called in the Developer managedBeans
Environmental Needs	I8 succeeded

3.2 Test procedures

3.2.1 Test procedure TP1

Test Procedure Identifier	TP1
Purpose	<p>This test procedure verifies whether all the Managers of the Business Layer can successfully:</p> <ul style="list-style-type: none">• receive requests from the corresponding managedBeans• correctly elaborate those requests
Procedure Steps	Execute I1, I2, I3, I4

3.2.2 Test procedure TP2

Test Procedure Identifier	TP2
Purpose	<p>This test procedure verifies whether the Calls Manager and the Ride Manager can successfully receive and handle requests from:</p> <ul style="list-style-type: none">• Visitor Manager• Passenger Manager• TaxiDriver Manager• Developer Manager <p>Also, this procedure verifies whether the Ride Manager can successfully receive and handle requests from the Calls Manager</p>
Procedure Steps	Execute I5 and I6

3.2.3 Test procedure TP3

Test Procedure Identifier	TP3
Purpose	<p>This test procedure verifies whether the Java Persistence Module can successfully receive, handle and reply to requests from:</p> <ul style="list-style-type: none">• Visitor Manager• Passenger Manager• TaxiDriver Manager• Developer Manager• Calls Manager• Ride Manager
Procedure Steps	Execute I8 after I7

3.2.4 Test procedure TP4

Test Procedure Identifier	TP4
Purpose	<p>This test procedure verifies whether the Passenger Manager and the TaxiDriver Manager can successfully receive and handle inputs from:</p> <ul style="list-style-type: none">• Calls Manager• Ride Manager <p>Also, this procedure verifies whether the Calls Manager can successfully receive and handle inputs from Ride Manager</p>
Procedure Steps	Execute I9 and I10

3.2.5 Test procedure TP5

Test Procedure Identifier	TP5
Purpose	<p>This test procedure verifies whether all the managed-Beans can successfully:</p> <ul style="list-style-type: none">• receive inputs from the corresponding Managers• correctly elaborate those inputs
Procedure Steps	Execute I11

Chapter 4

Tools & Test Equipment Required

Chapter 5

Program Stubs & Test Data Required