

Politecnico di Milano  
A.A. 2015-2016  
Software Engineering 2: “myTaxiService”  
**Integration Test Plan Document**

Roberto Clapis (841859), Erica Stella (854443)

January 15, 2016



# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Revision History . . . . .	3
1.2	Purpose and Scope . . . . .	3
1.3	List of Definitions and Abbreviations . . . . .	3
1.4	List of Reference Documents . . . . .	3
<b>2</b>	<b>Integration Strategy</b>	<b>3</b>
2.1	Entry Criteria . . . . .	3
2.2	Elements to be Integrated . . . . .	3
2.3	Integration Testing Strategy . . . . .	4
2.4	Sequence of Component/Function Integration . . . . .	4
2.4.1	Software Integration Sequence . . . . .	4
2.4.2	Subsystem Integration Sequence . . . . .	5
<b>3</b>	<b>Individual Steps and Test Description</b>	<b>6</b>
3.1	Test case specifications . . . . .	6
3.1.1	I1 . . . . .	6
3.1.2	I2 . . . . .	6
3.1.3	I3 . . . . .	6
3.1.4	I4 . . . . .	6
3.1.5	I5 . . . . .	6
3.1.6	I6 . . . . .	7
3.1.7	I7 . . . . .	7
3.1.8	I1 . . . . .	7
3.1.9	I2 . . . . .	7
3.1.10	I7 . . . . .	7
3.1.11	I8 . . . . .	8
3.1.12	I91 . . . . .	8
3.2	Integration Test Procedures . . . . .	8
<b>4</b>	<b>Tools and Test Equipment Required</b>	<b>8</b>
<b>5</b>	<b>Program Stubs and Test Data Required</b>	<b>8</b>

# 1 Introduction

## 1.1 Revision History

## 1.2 Purpose and Scope

This document describes the Integration Test Plan for the myTaxiService application. It provides a plan referring to how the various components described in the Design Document will be integrated for testing.

## 1.3 List of Definitions and Abbreviations

- *UI*: User Interface.
- *Database's stubs*: Active Requests and Reservations' stub and Accounts' stub.
- *UIs' stubs*: ClientUI's stub and DriverUI's stub.

## 1.4 List of Reference Documents

- The document with myTaxiService's description
- myTaxiService's RASD
- myTaxiService's Design Document

# 2 Integration Strategy

## 2.1 Entry Criteria

Before the integration testing, each single module must have been tested to verify the correct functioning of its methods according to its specifications.

## 2.2 Elements to be Integrated

According to the Design Document, the components to be integrated are:

- Database:
  - Accounts
  - Active Reservations and Requests
- Web Server:
  - DBConnector
  - APIBackend
  - WebpageCreator

- NotificationModule
- HttpHandler
- UI:
  - ClientUI
  - DriverUI
  - AdminUI

## 2.3 Integration Testing Strategy

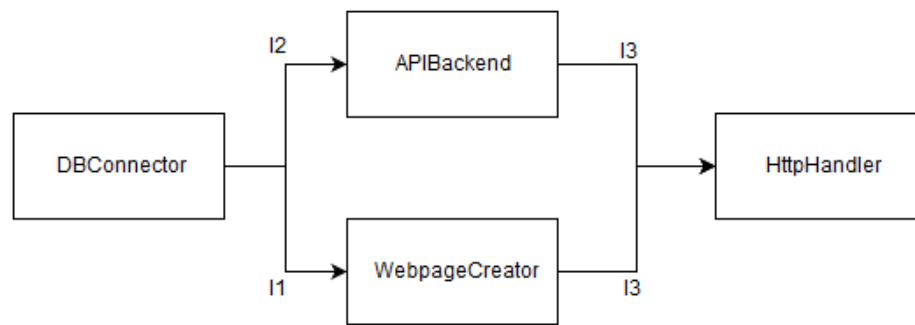
The decided testing approach is sandwich. This has been chosen in order to integrate first the components of the WebServer, and then integrate the WebServer with the Database and the UI.

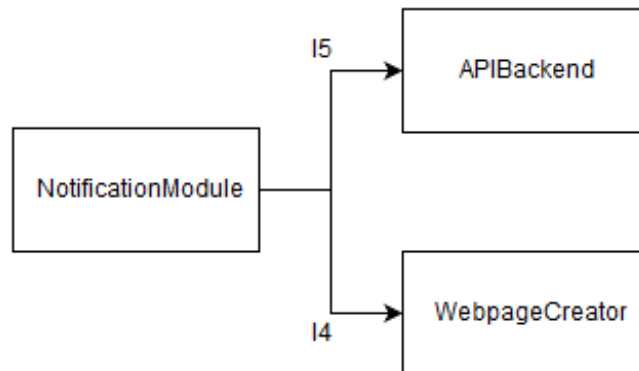
## 2.4 Sequence of Component/Function Integration

### 2.4.1 Software Integration Sequence

In the following graphs the arrows go from the callee module to the caller module and are marked with identifiers that define the order of integration.

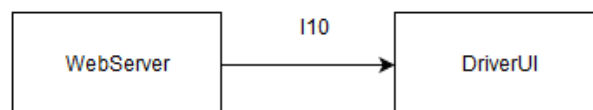
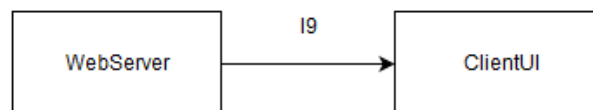
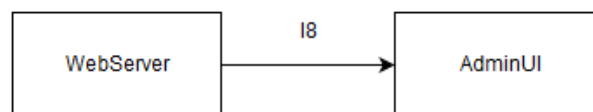
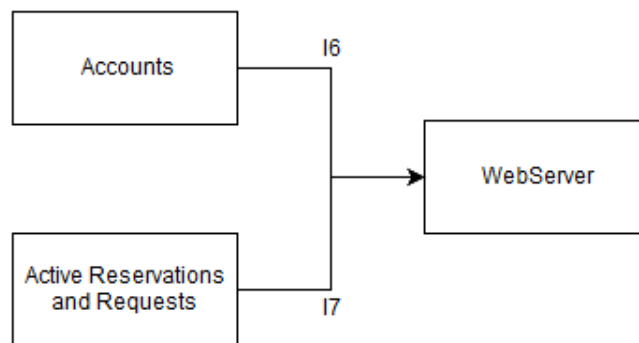
**2.4.1.1 Web Server** The following images describe how the Web Server's components will be integrated for testing.





### 2.4.2 Subsystem Integration Sequence

The following graphs describe how the three main components of myTaxiService application, the Database, the Web Server and the UIs, will be integrated.



### 3 Individual Steps and Test Description

#### 3.1 Test case specifications

##### 3.1.1 I1

Test Case identifier	I1T1
Test Items	DBConnector → WebpageCreator
Input Specification	Perform valid and invalid requests on the WebPage Creator
Output Specification	All and only the queries that should be allowed are executed and the correct DBConnector's methods are called
Environmental Needs	Database's stub, WebpageCreator driver

##### 3.1.2 I2

Test Case identifier	I2T1
Test Items	DBConnector → APIBackend
Input Specification	Perform valid and invalid requests on the APIBackend
Output Specification	All and only the queries that should be allowed are executed and the correct DBConnector's methods are called
Environmental Needs	Database's stub, APIBackend driver

##### 3.1.3 I3

Test Case identifier	I3T1
Test Items	WebpageCreator → HttpHandler
Input Specification	Perform valid and invalid requests on the HttpHandler
Output Specification	Verify if only the requests intended for the WebpageCreator are forwarded to it and the correct WebpageCreator methods are called
Environmental Needs	Database's stub, HttpHandler driver, I1,I2 successful

---

Test Case identifier	I3T2
Test Items	APIBackend → HttpHandler
Input Specification	Perform valid and invalid requests on the HttpHandler
Output Specification	Verify if only the requests intended for the APIBackend are forwarded to it and the correct APIBacken methods are called
Environmental Needs	Active Requests and Reservations' stub, Accounts' stub, HttpHandler driver, I1,I2 successful

##### 3.1.4 I4

Test Case identifier	I4T1
Test Items	NotificationModule → WebpageCreator
Input Specification	
Output Specification	
Environmental Needs	UIs' stub, WebpageCreator driver

##### 3.1.5 I5

Test Case identifier	I5T1
Test Items	NotificationModule → APIBackend
Input Specification	
Output Specification	
Environmental Needs	UIs' stub, APIBackend driver

### 3.1.6 I6

<b>Test Case identifier</b>	I6T1
<b>Test Items</b>	Accounts → WebServer
<b>Input Specification</b>	Queries to manipulate (creation modification and deletion) of accounts
<b>Output Specification</b>	The correct Accounts' method have been called
<b>Environmental Needs</b>	WebServer driver, Active Reservations and Requests' stub, I1-15 successful

### 3.1.7 I7

<b>Test Case identifier</b>	I7T1
<b>Test Items</b>	Active Reservations and Requests → WebServer
<b>Input Specification</b>	Queries to place/accept/delete reservations and requests, in every possible order of execution
<b>Output Specification</b>	
<b>Environmental Needs</b>	WebServer driver, I1-I6 successful

### 3.1.8 I1

<b>Test Case identifier</b>	I1T1
<b>Test Items</b>	DBConnector → Accounts
<b>Input Specification</b>	Queries to manipulate (creation modification and deletion) of accounts
<b>Output Specification</b>	The actual modifications intended
<b>Environmental Needs</b>	DBMS Driver

### 3.1.9 I2

<b>Test Case identifier</b>	I2T1
<b>Test Items</b>	DBConnector → Active Reservations and Requests
<b>Input Specification</b>	Queries to place/accept/delete reservations and requests, in every possible order of execution
<b>Output Specification</b>	The actual modifications intended and the rejection of the invalid requests
<b>Environmental Needs</b>	DBMS Driver

### 3.1.10 I7

<b>Test Case Identifier</b>	I7T1
<b>Test Items</b>	Client UI → HTTP Handler
<b>Input Specification</b>	Typical ClientUI input in the web form
<b>Output Specification</b>	The full content of the page
<b>Environmental Needs</b>	All previous tests

---

<b>Test Case Identifier</b>	I7T2
<b>Test Items</b>	Client UI → HTTP Handler
<b>Input Specification</b>	Typical ClientUI input in the mobile form
<b>Output Specification</b>	The dynamic content of the page
<b>Environmental Needs</b>	All previous tests

### 3.1.11 I8

---

<b>Test Case Identifier</b>	I8T1
<b>Test Items</b>	Driver UI → HTTP Handler
<b>Input Specification</b>	Typical DriverUI input
<b>Output Specification</b>	The content of the dynamic parts of the application
<b>Environmental Needs</b>	I1 → I6

---

### 3.1.12 I91

---

<b>Test Case Identifier</b>	I9T1
<b>Test Items</b>	Admin UI input → HTTP Handler
<b>Input Specification</b>	Typical AdminUI input
<b>Output Specification</b>	The pages expected to be rendered
<b>Environmental Needs</b>	I1 → I6

---

## 3.2 Integration Test Procedures

## 4 Tools and Test Equipment Required

## 5 Program Stubs and Test Data Required