POLITECNICO MILANO 1863

POWER ENJOY

Integration Test Plan Document

Lorenzo Casalino - 877421

Tommaso Castagna

Document version 1.0

Contents

1 Introduction

1.1 Revision History

The history of document revisions is here recorded in tabular format, mapping the document version with the major changes brought to.

The current version of the document is highlighted by the version number in bold format.

Version	**Revision**
1.0	Document first final version.

1.2 Purpose

The *Integration Test Plan Document*, also abbreviated as *ITDP*, aims to provide to the development team the path to follow for the integration testing process of the software system through a complete description of the elements of the system to test, the integration strategy to adopt, the integration sequence forecasted and stubs/drivers and tools needed to accomplish the integration test phase.

1.3 Scope

This document contains a detailed description of the integration testing plan that the development team should follow to successfully complete the integration testing process.

The integration plan starts with the description of the overall integration strategy: the entry criteria that must be met before any specific unit of the software may be tested are presented, following with the units to test. The strategy to follow in order to guide the integration process is described and justified through a rationale.

2 Integration Strategy

2.1 Entry Criteria

Before the integration testing phase of specific components may take place, it is fundamental that the following criteria (or conditions) here described are satisfied. The verification of these conditions is really important in order to have as output of the integration test phase meaningful result, useful to assess the quality of the software system designed and, possibly, improve it.

It's worthful point out that some of the criteria presented are strictly tied to the kind of strategy chosen to perform the integration testing of the system's components. For informations about the integration testing strategy picked out for this software system, please refer to the *Integration Testing Strategy* section.

Other criteria, instead, are more general and act as pre-conditions to the whole integration testing process.

GENERAL CRITERIA

- **RASD complete draw up:** the RASD has the main function of thoroughly document the functionalities and requirements of the software system. Because of its purpose, it is the first important source of informations and comparison that the development team has to refer to check the results obtained after the integration testing of components. Therefore, the complete draw up of the RASD document is an important condition to satisfy in order to proceed with the integration testing phase.
- **RASD positive assessment**: since the development team refers to the RASD to veify the results of the integration tests, it's fundamental that the RASD content reflects the goals of the stakeholders involved into the project. Documented functionalities diverging from the stakeholders' desires lead to wrong implementations, regardless of the integration testing results.

COMPONENT TESTING CRITERIA

- **Complete functionality-to-component(s) mapping:** the development team needs to know what components support a precise functionality. Hence, when a certain functionality is going to be tested, is necessary that all the components needed are known.
- **Complete component-related-to-functionality documentation:** the kind of support provided by a certain component to a certain system functionality must be described in the Design Document.
- **Complete components interaction description:** to test the integration and interaction between components the Design Document must report in depth how they interact and communicate to support a specific

functionality.

- **Complete component static and dynamic analysis:** the supporting functionalities provided by the components to be integrated must thoroughly inspected through static analysis methods, such as code inspection, and dynamic analysis methods, i.e. unit testing. This step is important because allows to discover possible fault in the implementation, easing and speeding up the integration testing process.
- **Regression testing:** before starting the integration testing of a certain functionality through new test cases, the old test cases should be run in order to verify the compatibility and the correctenes of the new integration. This process should be executed before the new functionality testing because if the old tests show an incompatibility, the components integration must be reviewed.

2.2 Elements to be Integrated

In the following, the system's elements to be integrated are spotted and described. With *elements* here we mean the subsystems composing the system's design. The description of a subsystem is recursive, meaning that if a subsystem is composed by other subsystems, even these are described. Description of atomic components is avoided since are well described into the DD.

The main *elements* composing our system are:

- **Data Tier:** This element represents the DBMS. Even though we are not developing the DBMS itself, it is still part of the system, hence it has to be integrated.
- **Business Logic:** This element contains all the application logic present in our system. The main components of this element are: the **Authentication Manager**, the **Account Manager**, the **Maintenance Manager**, the **Vehicle Manager** and the **Reservation Manager**.
- **Web Server:** This is the element responsible for the communications between the business logic tier and the clients. It exposes a RESTful implementation of the communication interfaces.
- **Client:** This element contains the three different kind of clients that can access our application, namely the **mobile client**, the **web application client** and the **car central system**.

Every component that forms the different *elements* has aledy been tested individually, the tests to be performed are applied to the interfaces that connect a component with another one.

2.3 Integration Testing Strategy

3 Effort Spent

The effort spent by each member of the group in terms of hours is shown in the following:

- 27 December 2016 - 1h 10m
- 28 December 2016 - 1h
- 29 December 2016 - 1h 10m