Politecnico di Milano

A.A. 2016-2017

Software Engineering 2: “PowerEnjoy”

Test Plan

Version 1.2

Prasanth Ravulapalli

Fathima B

Lipika L

January 15th 2017

Contents

Introduction

Purpose and Scope.

This document is intended to present the Integration test plan for the PowerEnjoy application, in the context of the project of the course Software Engineering 2.

This document describes the plans for testing the integration of the created components. The purpose of this document is to test the interfaces between the components. For that reason, this document is strongly related to the high-level design presented in the Design Document, and thus does not provide specific details neither on the testing of the individual components nor on the whole system testing.

Scope

The software implements a computational grid. This grid can execute jobs when it 20 receives an application accompanied by a set of data files. By hiding the complexity of grid technology, the system will be easy to use. Usability is also increases by offering a web-based front-end for users to access the system.

List of Definitions and Abbreviations

The following are abbreviations are used in the present document (and have not already been presented in any of the reference documents):

- IP: Integration procedure
- IS: Integration step
- ITP: Integration test plan
- Top down integration testing: it is an integration testing strategy which consists on performing the integration of the components from the uppermost to the lowermost. At each step, it requires the use of stub components to emulate the lower level parts of the system, where the invocations will be sent to.

List of Reference Documents.

The following is the list of documents that are related to this Test plan, and that totally define its context:

- RASD document for the same project
- Sample Test plan document.pdf

The Integration Test Plan Example was used as a guideline to develop the structure of this document though it is not relevant to understand the domain problem.

Integration Strategy

In this section, we present the strategy that will be used to guide the test plan. This includes the preconditions to execute it, the specific components that will be tested, the description and justification of the strategy, and a list of the concrete steps to be followed.

Entry Criteria

Within the context of the software development process, this test plan can only be executed when the following conditions hold:

- A requirements specification of the domain problem must have been done. In our case, this is presented in the RASD.
- A high-level design specification of the solution must have been performed. This is included in the DD.
- The source code of the implemented components that will be tested must be available.
- Code inspection activities are recommended to be executed on the source code of the components, even though it is not entirely mandatory.
- The components to be tested must have successfully passed the unit testing phase.

Elements to be integrated

The following list of components to be tested is based on the components presented in the Component view section of the Design Document:

- UserWebView
- UserController
- UserMobileView
- UserNotification
- GuestMobileView
- GuestController
- GuestWebView
- GuestNotification
- SystemOperationManager
- ReservationManager
- RideManager
- Model
- DB

The components WebBrowser, Email and SMS are also included in this test plan even though they are external systems. This is because the integration testing is intended to validate the functionalities of the system when the different components are put together.

Sequence of Component/Function Integration

Once that we have presented the strategy to guide the testing process and the specific parts of the system to be validated, we continue by listing the concrete steps that are to be followed to execute this test plan

Software Integration Sequence

All the components presented in the Component view of the Design Document can be considered as subsystems of the PowerEnjoy application. Nevertheless, we only provided a deeper view only for Model, so we list integration steps only for its subcomponents.

For applying the top-down strategy we take the NotificationManager as critical component. We incrementally add the real components to the integration:

Integration Test Step ID	**Involved components**
IS1-T1	RequestManager, ReservationManager
IS1-T2	ReservationManager, RequestManager
IS2-T1	RequestManager, QueueManager
IS2-T2	UserManager, QueueManager
IS3-T1	RequestManager, UserManager
IS4-T1	RequestManager, GuestManager
IS4-T2	GuestManager, RequestManager
IS5-T1	ReservationManager, DataManager
IS6-T1	UserManager, DataManager
IS6-T2	GuestManager, DataManager
IS7-T1	Model, NotificationManger
IS8-T1	NotificationManger, MessageManager
IS9-T1	Model, DB
IS10-T1	Model, SystemOperationManager
IS11-T1	NotificationManger, UserNotification
IS11-T2	NotificationManger, GuestNotification
IS12-T1	UserMobileView, UserController
IS12-T2	UserController, Model
IS13-T1	GuestMobileView, GuestController
IS13-T2	GuestController, Model
IS14-T1	WebBrowser, UserWebView
IS14-T2	UserWebView, UserController
IS14-T3	UserController, Model

Note: You can find diagrams illustrating the integration sequence of the system and the subsystem in the appendix

Individual Steps and Test Description

In this section, we provide the description of the integration steps. Since the purpose of this document is not to describe detailed testing but presenting the general guidelines for it.

Description of integration steps

ID	IS1-T1
Components involved	RequestManager → ReservationManager
Environmental conditions	RequestManager, ReservationManager components DataManager stub
Input description	Create a valid RequestManager inputs such as interactions between User and the application. The inputs are of following: • User request for a car • User request for end of the ride
Output description	We expect that the methods invoked during this series of calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS1-T2
Components involved	ReservationManager → RequestManager
Environmental conditions	RequestManager, ReservationManager components UserManager, GuestManager
Input description	Create a valid ReservationManager input such as results generated by application to the user: The inputs are of following: • Result for user's request for a car • Result for user's request for end of the ride
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS2 -T1
Components involved	RequestManager → QueueManager
Environmental conditions	RequestManager, QueueManager components
Input description	Create a valid RequestManager input such as creating a user request and push them into the queue through QueueManager. The inputs are of following: • User request to provide a car • User request to reserve a car • User request to cancel a ride
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS2-T2
Components involved	UserManager → QueueManager
Environmental conditions	RequestManager, UserManager component
Input description	Create a valid UserManager input to the QueueManager. The inputs are of following: • User reaching the reserved car • User update position • User unlocking/ locking the car
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS3-T1
Components involved	RequestManager → UserManager
Environmental conditions	RequestManager , UserManager components Model, Database
Input description	Create a valid RequestManager input such as result generated by application to the user's request. The inputs are of following: • Result for user's request for a car • Result for user's request for end of the ride • Result for user's request for show cars

Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS3-T2
Components involved	UserManager → RequestManager
Environmental conditions	RequestManager, UserManager components Model, Database The tests IS1 and IS2 must have succeeded.
Input description	Create a valid UserManager input such as request generated by user to the application through RequestManager. The inputs are of following: • User show cars • User request for car • User request for end of ride
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS4-T1
Components involved	RequestManager → GuestManager
Environmental conditions	RequestManager , GuestManager components Model, Database
Input description	Create a valid RequestManager input such as result generated by application to the guest's request. The inputs are of following: • Result for guest's request for show cars
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS4-T2
Components involved	GuestManager → RequestManager
Environmental conditions	RequestManager, GuestManager components Model, Database
Input description	Create a valid GuestManager input such as request generated by guest to the application through RequestManager. The inputs are of following: • Guest show cars
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS5-T1
Components involved	ReservationManager -> DataManager
Environmental conditions	The tests IS1 to IS3 must have succeeded. A stub for the DataBase component is used.
Input description	Create a valid ReservationManager input such as request generated by user to the application. The inputs are of following: • User cancel request • Process Request
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of ReservationManager with the DataManager and the already tested part of the system.

ID	IS6-T1
Components involved	UserManager -> DataManager
Environmental conditions	The tests IS1, IS2, IS3, IS5 must have succeeded. A stub for the DataBase component is used.
Input description	Create a valid UserManager input such as request generated by user to the application. The inputs are of following: • User email confirmation • User create account • User edit account • User log in
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of UserManager with the DataManager and the already tested part of the system.

ID	IS6-T2
Components involved	GuestManager -> DataManager
Environmental conditions	The tests IS1, IS2, IS4 and IS5 must have succeeded. A stub for the DataBase component is used.

Input description	Create a valid GuestManager input such as request generated by guest to the application. The inputs are of following: • Guest email confirmation
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of GuestManager with the DataManager and the already tested part of the system.

ID	IS7-T1
Components involved	Model -> NotificationManger
Environmental conditions	The tests IS1 to IS6 must have succeeded. Stubs of the MessageManager components are used. Drivers of the OtherControllers components are used.
Input description	Create a valid Model input. The inputs are of following: • Guest email confirmation • User account creation • User edit account • User email confirmation • User log in • Process request
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of Model with the NotificationManager only.

ID	IS8-T1
Components involved	NotificationManger -> MessageManager
Environmental conditions	The tests IS1 to IS7 must have succeeded. Stubs of the MessageManager components are used. We have actual access to the Email servers and SMS protocols which belongs to external system.
Input description	Create a valid NotificationManger input which can be sent to the MessageManager. The inputs are of following: • Send a mail body with the recipient added • Send a SMS to a mobile number
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of NotificationManger with the external system MessageManager's Email/SMS system.

ID	IS9-T1
Components involved	Model -> DB
Environmental conditions	DataBase should be accessible by the Model component. Tests from IS1 to IS8 are to be succeeded.
Input description	Create a valid input which can be sent to the Model which are required by the procedures.
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of Model with the DB and the already tested part of the system.

ID	IS10-T1
Components involved	Model -> SystemOperationManager
Environmental conditions	The tests IS1 to IS9 must have succeeded. Stubs from the SystemOperationManager and OtherControllers components are used.
Input description	Create a valid Model input which can be sent to the SystemOperationManager. The inputs are of following: • Calculate the price

	• Add bank details • Make a transaction • Input validators
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of Model with the SystemOperationManager.

ID	IS11-T1
Components involved	NotificationManger -> UserNotification
Environmental conditions	
Input description	Create a valid NotificationManger input which can be sent to the UserNotification. The inputs are of following: • Send Email • Send SMS
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of NotificationManger with the UserNotification.

ID	IS11-T2
Components involved	NotificationManger -> GuestNotification
Environmental conditions	
Input description	Create a valid NotificationManger input which can be sent to the GuestNotification. The inputs are of following: • Send Email • Send SMS
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of NotificationManger with the GuestNotification.

ID	IS12-T1
Components involved	UserMobileView -> UserController
Environmental conditions	IS11-T1 succeeded
Input description	Create a valid UserMobileView input which can be sent to the UserController. The inputs are of following: • User create account • User edit account • User login • User accept request • User cancel request
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS12-T2
Components involved	UserController -> Model
Environmental conditions	IS12-T1 succeeded
Input description	Create a valid input which can be sent to the UserController.
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of UserController with the Model and the already tested part of the system.

ID	IS13-T1
Components involved	GuestMobileView -> GuestController
Environmental conditions	
Input description	Create a valid GuestMobileView input which can be sent to the GuestController. The inputs are of following: • Guest create request • Guest send details
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	

ID	IS13-T2
Components involved	GuestController -> Model
Environmental conditions	IS13-T1 succeeded
Input description	Create a valid input which can be sent to the GuestController.
Output description	We expect that the methods invoked during the series of function calls generate the output and actions that correspond to the input provided.
Observations	This test is supposed to validate the integration of GuestController with the Model and the already tested part of the system.

ID	IS14-T1
Components involved	WebBrowser -> UserWebView
Environmental conditions	
Input description	The input will be the typical data for the procedure required by WebBrowser.
Output description	The output will be characterized by invoked methods and it match with the given input.
Observations	This test is supposed to validate the integration of WebBrowser with the MTSPassengerWebView.

ID	IS14-T2
Components involved	UserWebView -> UserController
Environmental conditions	IS14-T1 succeeded
Input description	The input will be the typical data for the procedure required by UserController.
Output description	The output will be characterized by invoked methods and it match with the given input.
Observations	This test is supposed to validate the integration of UserWebView with the UserController.

ID	IS14-T3
Components involved	UserController -> Model
Environmental conditions	IS14-T2 succeeded
Input description	The input will be the typical data for the procedure required by Model.
Output description	The output will be characterized by invoked methods and it match with the given input.
Observations	This test is supposed to validate the integration of UserController with the Model and the already tested part of the system.

Description of test procedures

Once we have described the test steps that make part of the integration plan, we now put them together into procedures accordingly to the functional role that each one of them has in the overall system.

We have defined 6 procedures:

Data management

Test Procedure ID	TP1-1
Related functionalities	This procedure is intended to verify the access to the database.
Chain of steps	Execute IS9, after having executed IS1 to IS7.

External systems communication

Test Procedure ID	TP2-1
Related functionalities	This procedure is intended to verify the communication with external components through the NotificationManager component.
Chain of steps	Execute IS8-T1, after executing IS7.

Notifications

Test Procedure ID	TP3-1
Related functionalities	This procedure is intended to verify the communication with UserNotification and GuestNotification through the NotifierManager component.
Chain of steps	Execute IS11-T1 and IS11-T2, after executing IS7 and IS8.

User functionalities

Test Procedure ID	TP4-1
Related functionalities	This procedure is intended to verify the functionalities offered to the User through the web application. The steps will allow to verify the behavior of both graphical and logical components.
Chain of steps	Execute IS14-T1 to IS14-T3, after having executed IS1 to IS11.

Test Procedure ID	TP4-2
Related functionalities	This procedure is intended to verify the functionalities offered to the User through the mobile application. The steps will allow to verify the behavior of both graphical and logical components.
Chain of steps	Execute IS12-T1 to IS12-T2, after having executed IS1 to IS11.

Guest functionalities

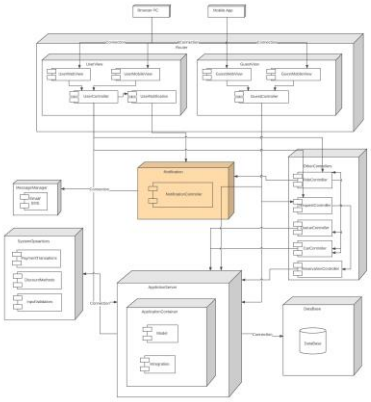
Test Procedure ID	TP5-1
Related functionalities	This procedure is intended to verify the functionalities offered to the Guest through the mobile application. The steps will allow to verify the behavior of both graphical and logical component.
Chain of steps	Execute IS13-T1 and IS13-T2, after having executed from IS1 to IS11.

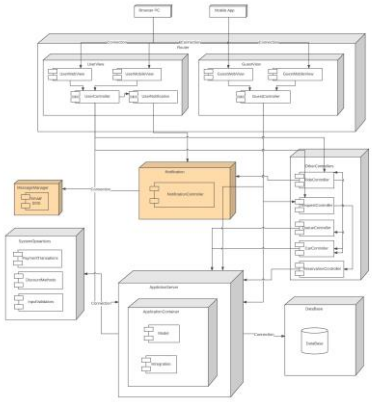
Business logic

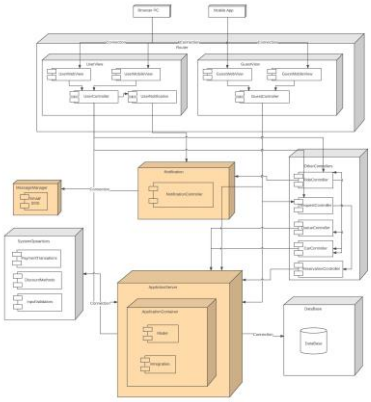
Test Procedure ID	TP1-1
Related functionalities	This test procedure verifies whether the Model: - handle Guest input - handle User input
Chain of steps	Execute IS10-T1, after executing IS1 to IS9.

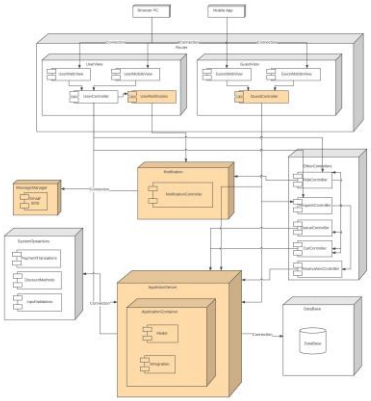
Appendix

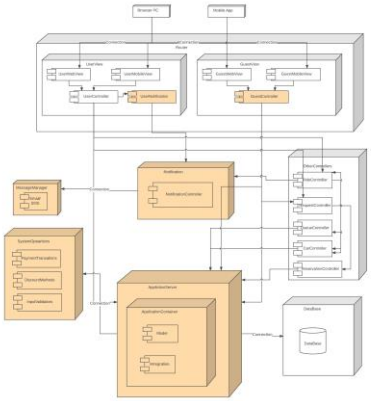
System integration sequence

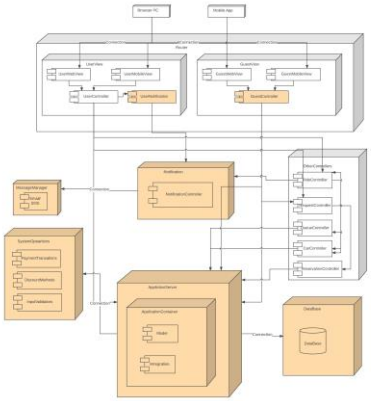


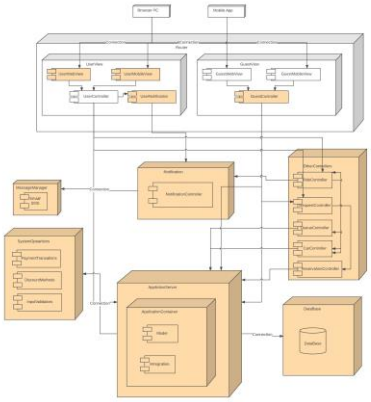


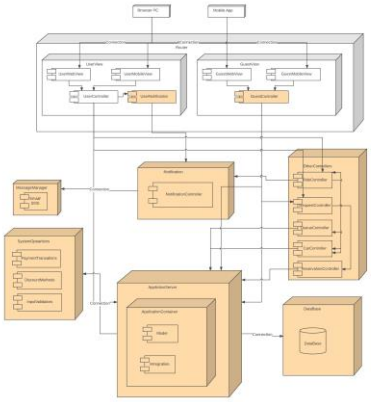


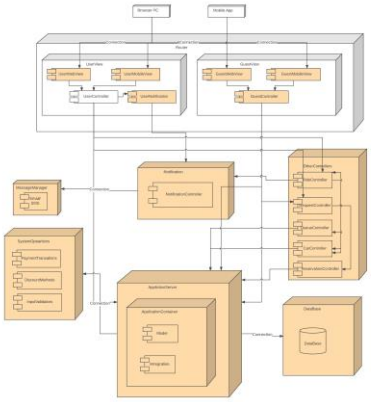


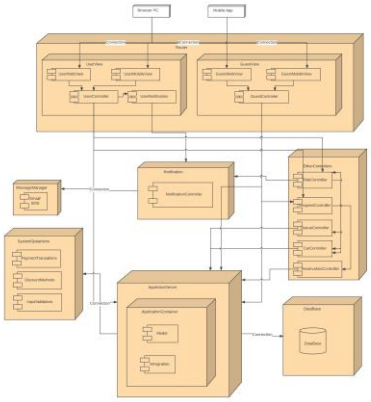


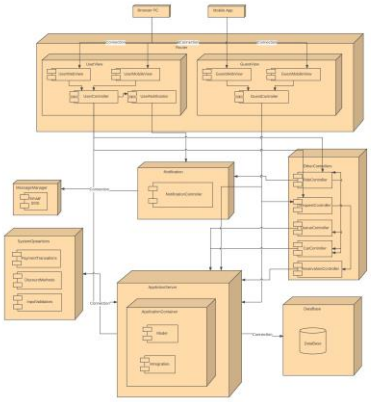












Hours of Work

Prasanth R: 18 hrs

Fathima B:

Lipika L:

Change Log:

- V 1.1
  - Created basic structure of the document.
  - Added all initial basic things to the document.
- V 1.2
  - Added all tests needed to performed.
  - Added all diagrams which represents the integration sequence.