Integration Test Plan

Matteo Greselin - Matr. 862908

Version 1.0

Contents

1 Introduction

1.1 Revision History

- June 2016 - Integration Test Plan Document: Version 1.0

1.2 Purpose and Scope

This document provide the planning for integration tests about smartCityAdvisor system. Here will be describe interfaces between the differents implementation present in the application. With this documen, the scope is provide a usefull sequence of action that should allow to tester to work without wasting time.

1.3 Glossary

RASD

DD

SCA

1.4 Reference Documents

- SmartCityAdvisor RASD
- SmartCityAdvisor DD
- Project Assignment provided by professors

2 | Integraztion Strategy

2.1 Entry Criteria

As each procedure, also integration test needs some specific requirements that must be present before that actions descript in this document could be performed. It's mandatory to develop unit test for each class, this means that all classes without any test developed could not be used in the integration test. Unit tests guarantee a minimal automated testing environment, to ensure that each revision to the code can be tested immediately and also allow to simplify error-finding procedures.
Integration tests will follow a bottom-up approach with respect to whats each module will need in order to work correctly, this mean that Account and Data Handler will be the first one tested.

2.2 Element To Be Integrated

Elements to be integrated are those individuat**Mockito**ed in DD document (section). Here i reports an alphabetic list:

1. Controller:
    - City Administration Communicator
    - Fire Fighter Communicator
    - Guest Communicator
    - Hospital Communicator
    - Police Station Communicator
    - Public Transport Company Communicator
    - User Communicator
2. Model:
    - Account and Data Handler
    - Event Handler

- Help Request Handler
- Map Hanlder
- Notification Handler
- Parking Handler
- Pollution Handler
- Public Transport Handler
- Queue Handler
- Traffic Situation Handler

3. View:

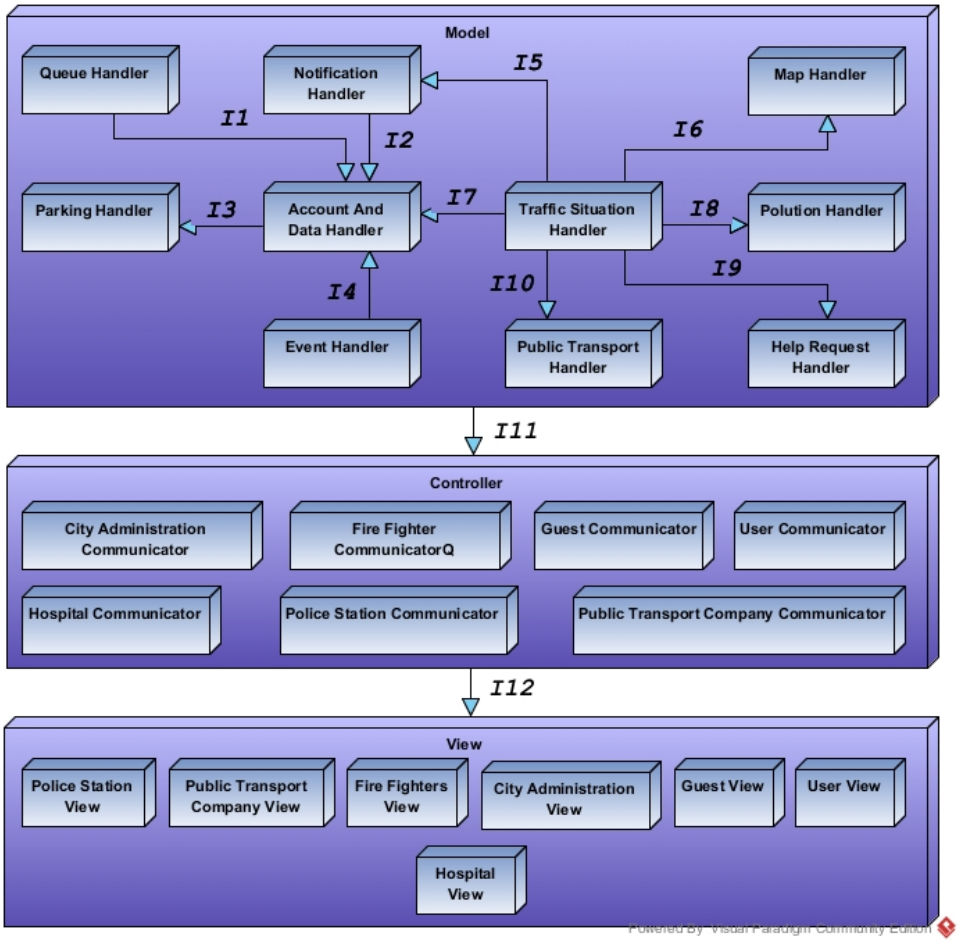
   - City Administration
   - Guest
   - Fire Fighters Station
   - Hospital
   - Police Station
   - Public Transport Company
   - User

The integration testing involves all the code modules developed, moreover, the end-user application has to be tested in the various system environments, that are:

- Unix based OS
- Windows systems
- Android
- iOS

2.3 Integration Strategy

Integration strategy should be considered as a bottom-up approach with respect to interactions with other component.
In the next is proposed a schema that clarify the sequence of integration. The idea is that when a component for a component will be performed the integration test, all the component that should be considered as 'secondary' for it must be already integrated.



2.4 Software Integration Sequence

For integration tests phase is supposed that each Handler will correctly interact with its actuator: this mean that Map Handler will be already interfaced with Google Maps API, Pollution Handler receives correct data from pollution sensors, Traffic Situation Handler can correclty communicate with monitors on the street and with traffic lights and so on.

The software integration start with the DB: it must be complete develope before that any test could be done. Then it will be integrated with Account and Data Handler, the only component with connection and privileges for comunicating with the DB. After this, could be start the real integration phase.

It has been choosen to start with integration between Account and Data Handler and Queue Handler becouse it is a component without interaction with other one. It can be conseidered as an 'insolate system': it must return correct data when guests or user require to see the situation in a specific hospital for a specific type of visit and it must provide functionality for updating queue coherently with normal hospital activities.

The successive integration test will include Account and Data Handler yet, but with Notification Handler: the test will check if Notification Handler could receive different list of users for forwarding notification, this is one of the most important functionalities in the application and it is used by other Handler as Event Handler, Traffic Situation Handler, Parking Handler or Public Transport Handler when they need to comunicate with users.

After it, will be done the integration test between Account And Data Handler and Parking Handler, this becouse Parking Handler could provade all functionalities only if Notification Handler will work correctly.
Then will be the turn of Event Handler with Account and Data Handler: it will allow to schedule new event, notify this action to user and provide information for Traffic Situation Handler so that it could manage road condition when event will be started

At this point i have to integrate Traffic Situation Handler with different component. It is the most important Handler after Account And Data Handler, and this hierarchy is only give by interaction with DB.
First integration of Traffic Situation Handler will be with Account And Data Handler, in order to check the correct communicaation with DB. Then will be done integration with Notification Handler, Map Handler, Pollution Handler, Public Transport Handler and Help Request Handler.
I want to bring out an important detail: when a new help request is insert into the system, it is obviousy saved in the DB but it isn't this the

real impact of a new request for the application scope. The impact in the application scope is done by the possibility from Traffic Situation Handler to updating the traffic situation in the map and showing to user and guests the new accident. Furthermore there will be the possibilities to sending notifiacation to users and all rescue entities (as police, hospitals and fire fighters).

The last two steps, after that all the application modules will be integrated and will be veified that the back-end work correctly, will be integrate component for forwarding information to actor: for this reasons will be performed integration test between Model component and Controller component and between Controller component and View component.

3 | Individual Step And Description

As explain in the previous chapter, we can consider to have a step 0 in the integration that include the integration between Hanlers and external actuator ad sensors, monitors and so on. In particular will be foundamenta the correct integration between DB and Account And Data Handler.

3.1 Integration Test 1

Test Case Identifier	I1
Test Items	Account and Data Handler & Queue Handler
Secondary Items	None
Input Specification	Method calls from Queue Handler
Output Specification	Check that the correct methods are called in Account And Data Handler
Tests applied	Tests done must verify that the component: • calls correct methods for returning the required queue (with all data) • calls correct methods for updating an indicated queue

3.2 Integration Test 2

Test Case Identifier	I2
Test Items	Account And Data Handler and Notification Handler
Secondary Items	None
Input Specification	Method calls from Notification Handler
Output Specification	Check that the correct methods are called in Account And Data Handler
Tests applied	Tests done must verify that the component: • calls correct methods for extracting and returning lists of users that must receive a notification • calls correct methods for forwarding the notification to assigned set of users or single users (for example during a car park reservation confirm) • calls correct methods for checking if the notification's sender has privileges in order to perform that action • calls correct methods for requiring administration feedback before forward a notification (if necessary)

3.3 Integration Test 3

Test Case Identifier	I3
Test Items	Account and Data Handler & Parking Handler Handler
Secondary Items	Notification Handler
Input Specification	Method calls from Parking Handler
Output Specification	Check that the correct methods are called in Account And Data Handler
Tests applied	Tests done must verify that the component: • calls correct methods for updating parking car availability when someone go in or out from it. • call correct methods for activate Notification Handler if user has performed a reservation in a specific cars park (always after have performed the updatig).

3.4 Integration Test 4

Test Case Identifier	I4
Test Items	Account and Data Handler & Event Handler
Secondary Items	Notification Handler
Input Specification	Method calls from Event Handler
Output Specification	Check that the correct methods are called in Account And Data Handler
Tests applied	Tests done must verify that the component: • calls correct methods for saving new event after that administration have scheduled it • calls correct methods for sending notification to all user in order to remind them the events

3.5 Integration Test 5

Test Case Identifier	I5
Test Items	Traffic Situation Handler Notification Handler
Secondary Items	None
Input Specification	Method calls from duringTraffic Situation Handler
Output Specification	Check that the correct methods are called in Notification Handler
Tests applied	Tests done must verify that the component: • calls correct methods for sendig notification to users when pollution sensors releve something • calls correct methods for sendig notification to users when is happened an accidents • calls correct methods for sendig notification to users when there is some problem with public transport • calls correct methods for sendig notification to users when an event block some streets

3.6 Integration Test 6

Test Case Identifier	I6
Test Items	Traffic Situation Handler & Map Handler
Secondary Items	None
Input Specification	Methods call from Traffic Situation Handler
Output Specification	Check that correct methods are called in Map Handler
Tests applied	Tests done must verify that the component: • calls correct methods in roder to integrate Google Map API with traffic situation information so all user and guest could see the real time traffic situation. • calls correct methods in order to mantain application's maps as current as possible

3.7 Integration Test 7

Test Case Identifier	
Test Items	Traffic Handler & Account And Data Handler
Other Stubs	Map Handler, Pollution Handler
Input Specification	Method calls from Traffic Situation Handler
Output Specification	Check that the correct methods are called in Account And Data Handler
Tests applied	Tests done must verify that the component: • call correct methods for updating Map Handler, if necessary • call correct methods for storing an high pollution level

3.8 Integration Test 8

Test Case Identifier	
Test Items	Traffic Handler & Pollution Handler
Secondary Items	None
Input Specification	Method calls from Traffic Situation Handler
Output Specification	Check that the correct methods are called in Pollution Handler
Tests applied	Tests done must verify that the component: • call correct methods for receivig data about pollution situation from polution sensors

3.9 Integration Test 9

Test Case Identifier	
Test Items	Traffic Situation Handler & Help Request Handler
Secondary Items	Notification Hanlder, Map Handler
Input Specification	Method calls from Help Request Handler
Output Specification	Check that the correct methods are called in Traffic Situation Handler
Tests applied	Tests done must verify that the component: • calls correct methods for activating traffic deviation, if necessary • calls correct methods for indicating new accindents in the maps • calls correct methods for requiring the activation of all functionalities related to new accidents, as notification to users • calls correct methods for advicing Police Station, Hospital and Fire Fighters near the accident, in order to be sure that everybody know what is happenend

3.10 Integration Test 10

Test Case Identifier	
Test Items	Traffic Situation Handler & Public Transport Company Handler
Secondary Items	Notification Handler
Input Specification	Method calls from Public Transport Company Handler
Output Specification	Check that the correct methods are called in Traffic Situation Handler
Tests applied	Tests done must verify that the component: • call correct method in order to forward notification to notification handler, after city administration approval • call correct method for updating transport line's scheduling • call correct method for constantly updating public transports position

3.11 Integration Test 11

Test Case Identifier	I11
Test Items	Model Component & Controller Component
Secondary Items	All Module Item must be already integrated
Input Specification	Method calls from Model Component
Output Specification	Check that the correct methods are called in Controller Component
Tests applied	Tests done must verify that the component: • calls

3.12 Integration Test 12

Test Case Identifier	I12
Test Items	Controller Component & View Component
Secondary Items	Module's items and Controller Component must be already integrated
Input Specification	Method calls from Controller Component
Output Specification	Check that the correct methods are called in View Component
Tests applied	The tests done must verify that: • guest communicator component call the correct method for: -showing traffic situation and requireing best way -showing events already scheduled by city administration -showing hospital queue -showing parking situation -showing public transport position • user communicator component call the correct method for: -showing notification when someone send something -showing form for reserve a car park • city administration communicator component call the correct method for: -showing form for schedule new events -showing all notification that require an approvation • police station communicator, hospital communicator and fire fighters communivator component call the correct method for: -showing the form for insert a new help request • public trasport company communicator component call the correct method for: -showing the form for send special notification to user, but only after city administration approval

4 | quipment RequiredUnit Test

In order to perform integration test, have been used some tools presented during lectures. These tools have been integrated with manual, as ultimate testing resource.

- **Mockito** has been used to test interatction with external service, as Google APIs. It also allow to isolate dependencies.
- **Arquillian** has an important role during the integration between system and DB: it is very helpful in order to understand if the database works well.
- When i need to simulate a fulfilled form or an information overload i might use **Jmeter** that allow to have consistency and fault tolerance tested. For these reasons it should also used to check if the system can work well under strain or if a big number of incoming requests may cause a failure.
- For most simple component with few interactions, i have used normal **Unit Test** becouse they not required complex tool for testing

5 | Program Stubs and Test Data Required

In this chapter i bring out some test procedure that have the scope to verify that each integration test is done correctly. Each TP will be structured with a short description about what the simulation includes andwhich integration test should be already done if i want that the TP could be performed in the right way.

5.1 Integration Test Procedure TP1

Test Procedure ID	TP1
Purpose	This test procedure simulate a precise situation in order to check the correct integration between different component. Will be necessary simulate a guest as actor that will check the queue situation in a specifi hospital
Procedure Steps	I0 (DB/Account and Data Handler) and I1 must be done

5.2 Integration Test Procedure TP2

Test Procedure ID	TP2
Purpose	This test procedure simulate a precise situation in order to check the correct integration between different component. Will be necessary simulate a user as actor that will check the car park availability in a specific car parking and he will decide to reserve on park
Procedure Steps	I0, I2 and I3 must be done

5.3 Integration Test Procedure TP3

Test Procedure ID	TP3
Purpose	This test procedure simulate a precise situation in order to check the correct integration between different component. Will be necessary simulate city administration that insert a new event. This mean that users will receive notification and data will be update.
Procedure Steps	I0, I2 and I4 must be done

5.4 Integration Test Procedure TP4

Test Procedure ID	TP4
Purpose	This test procedure simulate a precise situation in order to check the correct integration between different component. Will be necessary simulate that an event is started and user must be advice about each temporary change. Here will be involved also Traffic Situation Handler and Map Handler if we want to provide a real time information for our users
Procedure Steps	I0, I2, I4, I5, I6 and I7 must be done

5.5 Integration Test Procedure TP5

Test Procedure ID	TP5
Purpose	This test procedure simulate a precise situation in order to check the correct integration between different component. Will be necessary simulate the block of car traffic due to pollution levels
Procedure Steps	I0, I2, I5, I6, I7, I8 must be done

5.6 Integration Test Procedure TP6

Test Procedure ID	TP6
Purpose	This test procedure simulate a precise situation in order to check the correct integration between different component. Will be necessary simulate the integration of Public Transport Situation in the map and also simulate an attempt by Public Transport Company Administration to send an email that must be verify from City Administration
Procedure Steps	I0, I2, I5, I6, I7 and I10 must be done

5.7 Integration Test Procedure TP7

Test Procedure ID	TP7
Purpose	This test procedure simulate a precise situation in order to check the correct integration between different component. Will be necessary simulate a new help request captured by hospital, police station or fire fighter station, with respect to who will be contacted as first from the plece where the accident is happened
Procedure Steps	I0, I2, I4, I5, I6, I7 and I9 must be done

6 | Appendix

6.1 Used Tools

In order to create this document i have used the following tools:

1. **Lyx** for writing the pdf
2. **VisualParadigm**10 Community Edition for creating diagrams

6.2 Time Spent

For drafting this document i have spent about 15 hours.