Salute Trasparente

Testing Plan

Version:

1.0

Data:

23/11/2017

Authors:

Diego Donaggio Mat.853837

Ignazio Carbonaro Mat.855503

Angelko Fericean Mat.995851

Rodica Maria Țecu Mat.995845

CONTENT 2

Introduction **3**
Testing process **3**
Incremental Testing 3
Black-box testing 3
White-box testing 4
Traceability of requirements **4**
Traceability of Requirements Table
5
Tested elements **6**
Tests scheduling **6**
Testing table 7
Test recording procedure **8**
Hardware and software requirements **8**
Constraints **8**

Introduction

The purpose of this document is to provide detailed information that creates a quick overview of the testing activity performed by the group during the development of the application with incremental testing and white-box testing and the end of it with black-box testing.

Testing Process

Following an analysis of the possible testing strategies, it was decided to adopt the testing techniques listed below. The following tests will be performed on at least three different (real) devices and also using Android Studio Emulator.

Incremental Testing

We will proceed to testing the application step by step as new functionalities are implemented. In this way, we will verify the solidity of the application during the software development process.

Each time you implement one of the classes specified in the design document, tests will be carried out for early detection of any implementation errors.

Black-box testing

When the implementation of the application is complete, there will be carried out a set of "black box" type tests, using defined inputs (valid and invalid) and comparing the results obtained with the results expected.

This testing technique is very useful for detecting errors that common users could encounter during everyday use, without any knowledge of the implementation of the application, without any contact with the source code.

White-box testing

At the same time as the Incremental Testing we will adopt a White-Box testing approach. These will consist of detailed tests that will evaluate the structure of the application in order to carry out a final verification for acceptance of correctness of the entire software product.

Traceability of requirements

The functional requirements of the application will be verified by following the specification of the requirements in the Requirement Specification Document. In this way we are sure of the validity of the tested functional requirement. There will be tests performed also for some non-functional requirements, following the features of the Black-Box approach.

Traceability of Requirements Table

Requirement ID	Requirement Name	Test
FR-01	Map visualization	Scroll through the map, zoom in, zoom out. Tap on a few regions to see if the app gets you to the charts screen for those regions. Check to see if it is the same region.
FR-02	Statistical report	Tap on some values from the charts and check if the app correctly displays the detailed expenditures for that item.
FR-03	Menu	Tap on the menu button and check if the menu drops down and shows all the options available.
FR-04	Help section	Tap on the help button and check if it displays correct.
FR-05	App dictionary	In the help tap on the App Dictionary option and check if it displays correct.
FR-06	Info section	Check if in the menu button is an info section button and after tap on the info section button to check if it correctly displays the app info.
NFR-01	Available tutorial in the menu button	Check if in the menu button is available the tutorial and tap on the button to get to the Tutorial screen.
NFR-02	Colored regions	Check for between 4 and 6 regions the values for each and the values threshold for their colores. Check if the colors match.
NFR-03	Navigation outside the italian map	While the map is visible try to scroll through the map to other regions out of Italy and see if you are brought back to the map of Italy.
NFR-04	Legend with scale value	Compare with a Google Maps map of Italy and check if the scales are the same.
NFR-06	Drop down menu	Tap on the menu button and check if the menu drops down and shows all the options available.

NFR-07	Dictionary as a sorted list of terms	In the help tap on the App Dictionary option and check if it displays correct.

Tested elements

The elements subject to testing will be all the classes and features specified by the class diagram in the Document of Design. Functional and non-functional requirements will also be checked defined in the Requirements Specification Document.

The other elements of the application that will be thoroughly tested will be:

- Starting and closing the app
- Navigating the map and clicking on the regions
- Navigating the menu
- The tutorial
- The page with the charts
- The help .

Tests scheduling

Time and resources will be distributed as follows:

- Incremental Testing: Will be done whenever you add a new module to the application. In terms of time and resources, this process will handle as much as the application development;
- White-Box Testing: Even in this case, the test will proceed step by step with development, taking the same time and the resources of the previous case;

- Black-Box Testing: Will be applied at completion of the application. All members of the group will contribute to the test for a duration of about 5 days;
- All the testing of the requirements that are specified in this document will be performed as shown in the Testing Table below.

Testing Table

Requirement ID	Requirement Name	Tester	Date	Outcome
FR-01	Map visualization	Fericean Angelko	03.01.2018	
FR-02	Statistical report	Tecu Rodica	03.01.2018	
FR-03	Menu	Fericean Angelko	05.01.2018	
FR-04	Help section	Tecu Rodica	05.01.2018	
FR-05	App dictionary	Fericean Angelko	07.01.2018	
FR-06	Info section	Tecu Rodica	07.01.2018	
NFR-01	Available tutorial in the menu	Fericean Angelko	09.01.2018	
NFR-02	Colored regions	Tecu Rodica	09.01.2018	
NFR-03	Navigation outside the italian map	Fericean Angelko	11.01.2018	
NFR-04	Legend with scale value	Tecu Rodica	11.01.2018	
NFR-06	Drop down menu	Fericean Angelko	13.01.2018	
NFR-07	Dictionary as a sorted list of terms	Tecu Rodica	13.01.2018	

Test recording procedure

The tests will be recorded in a table, where the specification headline refers to the requirement from the Requirements Specification Document and the test result contains the considerations of the tester on the tests performed over the expected postconditions. So, the table will be very much alike the one that we have put in the Requirement Traceability section of this document.

The test results will be stored in a separate document to be able to be reported in case of necessity.

Hardware and software requirements

For testing, developers have available the following hardware platforms :

- Samsung S8+. Android version 7.0
- HTC One M7. Android version 5.0.2
- Android Studio Emulator. Google Nexus 5 Android versions 6.0 Marshmallow / 5.0 Lollipop / 4.4 KitKat

Constraints

The testing will begin officially on the 3rd of January 2018 and will end on the 13th of January 2017. During this period, team members will

gather information and will provide immediate correction / improvement in anticipation of the submission of the application on the 31st January 2017.