

Test Plan

24/11/2014

Stephen Tate

CO600: JustHealth
Supervisor: Yang He

Version 3.0

Last saved by: Ben McGregor

Last saved on: 30/03/2015

Contents

1. Introduction	2
1.1. Objectives	2
1.2. Testing Strategy	2
1.3. Scope	2
2. Roles and Responsibilities	3
3. Test Results	3
3.1. Pass Criteria	3
3.2. Failing a Test	3
3.3. Test Case Errors	4
3.4. Passing a test with exception	4
4. Testing Process	4
4.1. Schedule	4
4.2. Documenting Testing	4
4.3. Assumptions	5
4.4. Test Deliverables	6
5. Test Cases	7
5.1. Iteration 1	7

1. Introduction

This test plan is designed to outline the scope, approach, responsibility and schedule for all testing components. It will explicitly identify what is to be tested, the types of testing to be utilised, who is responsible for carrying out the testing and defining the testing schedule. Also, this plan will detail any assumptions and risks that may be associated with the plan.

1.1. Objectives

This test plan should outline a set of complete tests that are associated with every feature and functionality that JustHealth provide, in both their mobile and web applications. The test plan will detail in its entirety which tests apply specifically to the mobile and web application.

Before any feature can be added to a product version that is to be released, the feature must have passed all testing associated with it. This must have been signed off by the Tester and verified by the Test Manager.

The test plan will be a key document milestone for every feature; if it does not pass the required tests any version of software that contains the feature will not be released.

1.2. Testing Strategy

JustHealth have adopted a test-driven development strategy, all of the test cases will be written and based on the requirements that the web application and mobile application are expected to fulfill. Furthermore, due to the test-driven development that it being adopted throughout the project and for every iteration, the predominant method of testing that will be utilised is unit testing which will be mainly black box testing although, for some tests white box testing may be more appropriate. Both manual and automated testing will be used.

1.3. Scope

This test plan covers JustHealth for all of the testing that they carry out throughout the development lifecycle of both the JustHealth Web Application and the JustHealth Mobile Application. The JustHealth web and mobile applications are dependent on the testing and therefore, this will be something that will be continually performed throughout the development lifecycle of the products.

2. Roles and Responsibilities

Role	Name	Responsibility
Tester	TBD	<p>The tester is responsible for carrying out the tests that are detailed in the test cases. The tester is responsible for adding any problems that they come across to the issue tracker and also responsible to assign any issues that they open and inform the person assigned of the issue. The tester is responsible for ensuring that any test that fails is performed again.</p> <p><i>Note: The tester should not be the same person that has written the code for the specified functionality. Exceptions may be granted if no other tester is available.</i></p>
Test Manager	Stephen Tate	<p>The test manager is responsible for verifying all test cases are complete and test the desired functionality completely. Also, they are responsible to ensure that all tests are performed correctly and accurately. The test manager must verify and sign all test cases before any functionality can be added into a version of software that is suitable for release.</p> <p><i>Note: In the event that the Test Manager carries out the testing, both the technical and project managers will be required to sign the test cases too.</i></p>

3. Test Results

3.1. Pass Criteria

In order for a test to be marked as a pass, the expected test result, which is written in the test case, should be identical to the actual result. If this is not the case the test should be failed.

3.2. Failing a Test

Any fail that is recorded should subsequently be added to the issue log and assigned to the most appropriate person to track and follow through with the issue. If the test is unable to be passed even after troubleshooting, yet needs to be added to the software that is available for release, then a risk will need to be raised. This will need to be acknowledged using the document template that can be found in appendix one and signed by the Risk Manager, Test Manager, Technical Manager and the Project Manager.

3.3. Test Case Errors

If any errors are found within the test cases then an issue should be raised in the issue tracker. This should be followed up and amended by the tester and will be approved by the Test Manager once the test cases have been completed.

3.4. Passing a test with exception

Tests may be passed with exceptions if the functionality has changed slightly from what was proposed when the test was written. The reason for passing the test and the actual result of running the test should be written in the Notes section of the test result. This must be approved by the Test Manager and if required the project manager.

4. Testing Process

4.1. Schedule

Test cases will be designed at the start of every iteration and should be completed and agreed by the Test Manager before any coding commences.

The testing itself should be performed at the end of each iteration before the desired functionality is able to be added into the version that is identified as being fit for release (*i.e. master branch*). As JustHealth work on two week iterations it would be expected that testing also takes place every fortnight.

4.2. Documenting Testing

Before any testing can be carried out, the tester should have the testing portal running. Tests can be filtered by iteration and this is able to be done by going to the 'query test' page. Using the drop-down menu the tester will be able to select the iteration that they would like to view/run the tests for. All of the tests will then be displayed in a table below, each of these will have an associated 'Run Test' button and an 'Edit Test' button.

4.2.1. Test Cases

When running a test case the tester should identify the test case and select the 'Run Test' button that is associated with it. The test is then required to select their name from the drop down list, and select pass/fail depending on the outcome of the test. If the test is failed, an issue should be raised and this number should be recorded in the appropriate field. Comments should be included into the comments field which state the reason that the test failed. It is advisable to include the following:

- Reason for failure
- Detailed description of what was displayed
- Any information on how the issue may be fixed

4.2.2. Administration Procedures

All that are involved with any form of testing must adhere to the following procedures.

Failed Tests

The tests that initially fail, should be formally retested until they are passed. All failed tests should be recorded and must be kept for future reference.

Screen Shots

Where indicated screen shots should be obtained and included within an additional document, this should be saved in the following folder `JustHealth/Iteration [#]/Testing` on the master branch. Associated with all of the iterations that are carried out, there should be a separate document containing the screen shots; this will be created when the test cases are created. Tests that fail should have captured screenshots in the documents and should be mark appropriately.

4.3. Assumptions

This test plan assumes that both the Tester and Test Manager have a full understanding of the tests that need to be carried out and are capable of carrying them out accurately. It also assumes that individuals are trustworthy and will carry out the tests unbiased; ensuring that documentation reflects the tests actual results.

4.4. Test Deliverables

Deliverable Name	Required?	Responsibilities	Signed By
Test Plan	Yes	<p>Test Manager: The Test Manager should update the test plan with the test ID's and names of each test associated with a particular iteration. Furthermore, the test manager is also required to initial individual tests in the test plan once he is satisfied that the test cases have been completed and filed.</p> <p>Tester: Testers are required to date/time stamp and initial each individual test, once it has passed.</p>	<p>This document will need to be signed once all testing is complete by the Test Manager.</p> <p>Each individual test will need to be initialled by the Tester and the Test Manager.</p> <p><i>Note: In the event that the Test Manager carries out the testing, both the technical and project managers will be required to initial the individual tests.</i></p>
Test Cases	Yes	<p>Test Manager: The Test Manager is required check and verify every test case and test run.</p> <p>Tester: The test runs must be documented accurately in the test portal using the standard that has been outlined in this document.</p>	The tester is required to accurately fill out the form on the web portal and record all of the tests that are run.
Screenshots Document	If tests require screenshots to be captured.	Tester: The tester is required to capture screenshots and add them into the screenshot document for that particular iteration that the tests apply too.	N/A
Issue Entry	If any tests fail or issues are found with the test cases themselves.	Tester: The tester is responsible for raising the issues that are identified in the issue tracker and assigning these to the appropriate person.	N/A

5. Test Cases

5.1. Iteration 1

Test ID	Application Type	Date/Time Completed	Tester Initials	Test Manager Initials
1.1.1.	Database			
1.2.1.	Database			
1.2.2.	Database			
1.2.3.	Database			
1.2.4.	Database			
1.2.5.	Database			
1.2.6.	Database			
1.2.7.	Database			
1.3.1.	Database			
1.3.2.	Database			
1.3.3.	Database			
1.3.4.	Database			
1.3.5.	Database			
1.3.6.	Database			
1.3.7.	Database			
1.4.1.	Database			
1.4.2.	Database			
1.5.1.	Database			
1.5.2.	Database			
1.5.3.	Database			
1.5.4.	Database			
1.5.5.	Database			
1.5.6.	Database			
1.5.7.	Database			
1.5.8.	Database			
1.5.9.	Database			
1.6.1.	Database			

1.7.1.	Database			
1.7.2.	Database			
1.7.3.	Database			
1.7.4.	Database			
1.7.5.	Database			
1.8.1.	Database			
1.9.1.	Database			
1.10.1.	Web			
1.10.2.	Web			
1.10.3.	Web			
1.10.4.	Web			
1.11.1.	Web			
1.11.2.	Web			
1.12.1	Web			
1.12.2.	Web			
1.12.3.	Web			
1.12.4.	Web			
1.12.5.	Web			
1.12.6.	Web			
1.12.7.	Web			
1.13.1.	Web			
1.13.2.	Web			
1.13.3.	Web			
1.13.4.	Web			
1.13.5.	Web			
1.13.6.	Web			
1.13.7.	Web			
1.14.1.	ORM			
1.15.1.				
1.15.2.				
1.15.3.				

1.15.4.				
1.15.5.				
1.15.6.				
1.15.7.				
1.16.1.				
1.16.2.				
1.16.3.				
1.16.4.				
1.16.5.				
1.16.6.				
1.16.7.				
1.17.1.				
1.17.2.				
1.18.1.				
1.18.2.				
1.18.3.				
1.18.4.				
1.18.5.				
1.18.6.				
1.18.7.				
1.18.8.				
1.18.9.				
1.19.1.				
1.20.1.				
1.20.2.				
1.20.3.				
1.20.4.				
1.21.1.				
1.22.1.				