Test Plan

OneClickVitals

Yukhe Lavinia

Sirisha Dumpala

Date: 5/6/2015

Revision History:

Revision Number	Revision Date	Description of change	Author

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this test plan document is to provide a guideline to plan, execute and manage the testing of OneClickVitals software. It describes the scope, approach and strategy for testing the application. A variety of testing tools will be used to test OneClickVitals software and the same will be discussed further in this test plan document.

1.2 Scope

OneClickVitals is web-based medical record system software that seeks to improve the efficiency of patient medical data storage and management. It will also allow doctors and their medical staff to focus their time and attention on the patient instead of spending hours on documentation, thus improving the quality of the care provided by their office. OneClickVitals is designed to cater to a medical clinic with one doctor, his/her medical staff and their patients. Users can add, view and edit information based on the role they play and the permissions assigned to them. OneClickVitals allows users to create, view and edit new patient accounts, add, view and edit appointments, add and view vital signs, diagnosis, prescriptions, order lab tests and add lab test results.

The test plan for OneClickVitals will include testing the functional requirements using various methodologies and tools. All the functional requirements listed in the software requirement specification document will be tested against the implemented application to ensure 100% implementation of all the requirements. Further, at least 30% of the code will be covered for unit testing. Static testing will be performed to ensure that the code is written as per the recommended programming conventions and standards.

1.3 Roles and Responsibilities

Project Manager is responsible for the planning, execution and deployment of the project. He/she is also expected to be familiar with the Software Development Life Cycle so they can make informed decisions about budget, schedule and resources.

Responsibilities of a project manager:

- Develop the project plan
- Review and approve the test plan, budget
- Manage communication
- Manage conflict

- Manage the team

Development Team consists of Developer Lead and Developers.

The development team is responsible for researching, designing, implementing, and testing the software.

Responsibilities of the development team:

- Requirement specification
- Requirement analysis
- Design and implementation
- Support the Testing team and
- Review the deliverables
- Maintenance

Testing Team consists of Testing Team Lead and Testers

Testing lead is responsible for facilitating the communication between the testing team and developer team. He/she should also approve the test plan and review the test cases.

Testing team is responsible to test the software and ensure that:

- It meets the requirements as stated in the SRS
- It responds correctly to all types of inputs
- It performs the expected functions within an acceptable time
- It is user friendly
- It achieves the general result the stakeholders desire

1.4 Definitions, Acronyms, and Abbreviations

Term	Definition and Information
SRS	Software requirement specification
User	Person who uses the software application
Patient Account	Created by the doctor's office to maintain patient records
Summary	Brief description of the patient's visit on a particular date including diagnosis, prescription, vital signs, appointment information

System	An assembly of interconnected electronic components. This word refers to the OneClickVitals application
Web application	A computer application that runs on a web browser
Staff	A person who performs routine clinical and administrative duties under the direct supervision of a physician or other health care professional
Doctor	Licensed physician permitted by law and by a hospital to provide medical care within that hospital or facility
Patient	A person who seeks medical care

2. Testing Strategy

2.1 Test Types

There are many different types of software testing technique, and they all have their own strengths and weaknesses. There are two main categories of testing techniques, Static testing and Dynamic testing. Testing team for OneClickVitals will perform tests from both categories using various testing tools.

Static testing techniques do not execute the code and are therefore called non-execution techniques. It is performed to check if the programs written by the development team meet the coding standards of the programming language used. Static testing is also done to check the correctness of the all documents and ensure that all the requirements are met as per the contract.

Dynamic testing includes both white box and black box testing. White box testing for OneClickVitals was done by writing test cases for unit testing. Black-box testing is done by using Selenium IDE and writing python scripts.

2.1.1 Unit Testing

In unit testing, individual units of source code, usually a function or entire module are tested. Unit testing is key to a successful and error free software as it helps find problems early in the development cycle. It helps identify the bugs in the programmer's implementation and flaws or missing parts of the specification for the unit. Unit tests are usually written by the developers to ensure that their code behaves as intended.

Please refer to Attachment " " for the unit tests.

2.1.2 User Acceptance Testing

User acceptance testing is a test conducted to determine if the requirements listed in the SRS document and/or the contract is met. Acceptance tests are black-box tests where the user tests to see if the system is behaving as expected.

User acceptance test for OneClickVitals is performed using Selenium IDE and Selenium Python scripts.

Please refer to Attachment " " for test cases and test reports of testing using Selenium.

2.1.3 Static Testing

Static testing for OneClickVitals is done using Pylint, which is a source code bug and quality checker for Python programming language. Pylint generates a very verbose report of the analysis, which helps in improving the quality of the source code.

Please refer to Attachment " " for the static test report generated for OneClickVitals.

2.2 Testing Tools

The software for OneClickVitals was tested using the following testing tools:

Black-box testing: Selenium

White-box testing: Django Unittest

Static testing: Pylint

Selenium is a portable software-testing framework for web applications. Selenium is implemented as a Firefox Add-On and allows recording, editing, and debugging tests.

A screenshot of the Selenium IDE pop up window in Firefox:
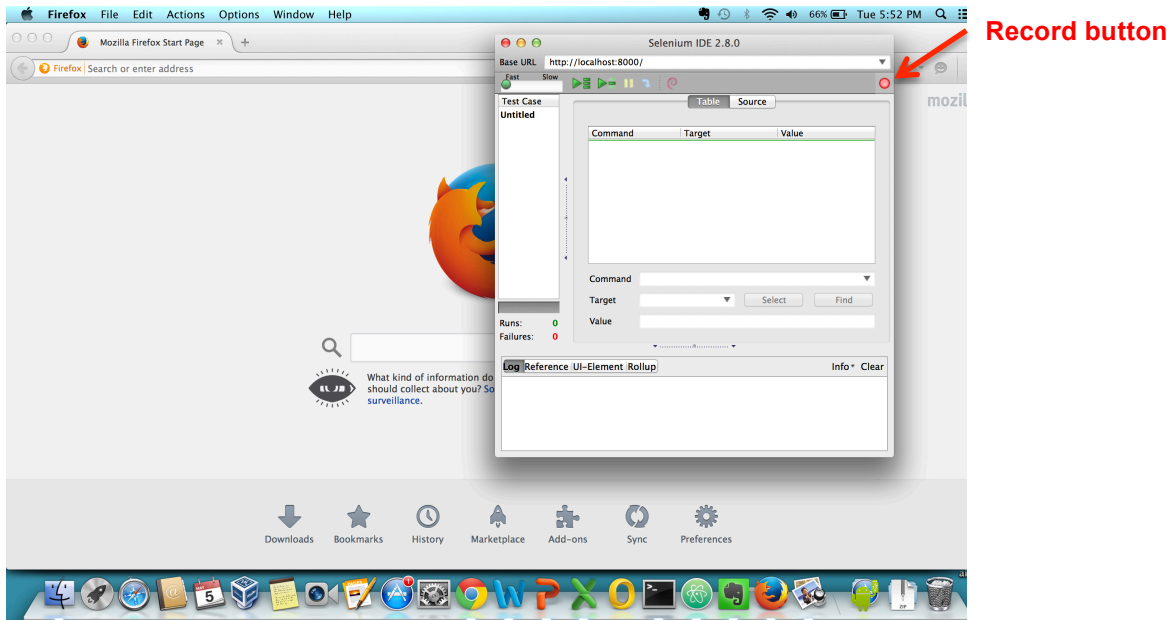
Record button

Selenium IDE can be selected from the Firefox Tools menu. It opens as follows with an empty script-editing window as shown in the above screenshot and a menu for loading, or creating new test cases.

The record button is on by default and it starts recording the user's actions into a test case.
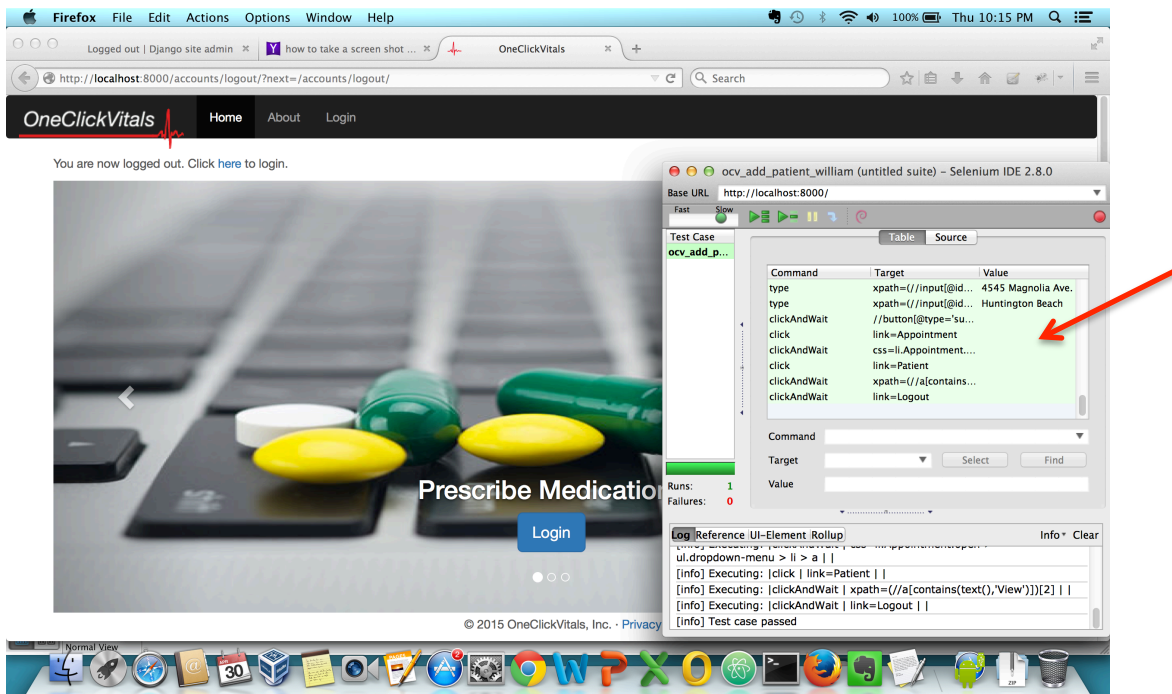
Please refer to Attachment "C" for OneClickVitals test reports generated by Selenium.

Selenium Python bindings provides a simple API to write functional/acceptance tests using Selenium WebDriver. It can be installed by using pip (Python installation packages). It is also recommended to use virtual environment to create isolated Python environments. Once installed, we can write test scripts in Python and save them as .py extension files and run on the terminal. Please refer to Attachment "D" for Python scripts and report.

Pylint is a Python source code analyzer, which looks for programming errors, helps enforcing a coding standard. It can be installed using pip(python installation package) and run on the terminal.

Installation(for mac): pip install pylint
Run: pylint ./<modulename>.py
After analyzing the source code, it generates a verbose report including all the metrics and produces a rating for the code.

Please refer to Attachment "A" for Pylint analysis reports.

Unittest is a dynamic testing tool or white-box testing and it comes as a part of the Django framework. In order to ensure that the software is robust, it is a good practice to write and develop tests and test suites. It saves time and helps identify and prevent problems.
When an application is created using the Django framework, a test.py file is created. Tests can be written using some of the following general rules:

- Tests should focus on one small bit of functionality
- Tests should have a clear purpose
- Tests should be independent.
- Run your tests, before you code, and before your commit and push your code.
- Even better create a hook that tests code on push.
- Use long and descriptive names for tests.

When we run the tests on terminal, a temporary database is constructed, which the tests can populate, and perform operations on. This way the testing is performed independently of the live database. The temporary database is destroyed once the tests are run.

Please refer to the Attachment "B" for unit test cases for OneClickVitals.

2.3 Test Cases and Results

The following are the test cases corresponding to the functional requirements of OneClickVitals:

Test Case ID	Type of the Test Case	Action	Expected	Status
TC_login_user_001	Login User	The User clicks on "Login" button	The system displays the login page with two fields for Username and Password	Pass
TC_login_success_001	Login Validation	The User enters correct Username and password	The system displays the User's Homepage	Pass
TC_login_empty_002	Login Validation	The User leaves one or both of the fields empty	The system prompts the User to fill in the required fields	Pass
TC_login_incorrect_003	Login Validation	The User enters incorrect Username and Password	The system displays an invalid login page and prompts the User to re-enter the correct Username and Password	Pass
TC_logout_user_001	Logout User	The User clicks on "Logout" button	The system displays a confirmation message that the User has been logged out of the application	Pass
TC_display_patient_form_001	Manage Patient Account	The User clicks on "Add New Patient " button	The system displays New Patient Information Forms in tabs	Pass
TC_display_next_patient_form_002	Manage Patient Account	The User fills the form and clicks on "Next" button	The system displays the form in the next tab	Pass
TC_empty_patient_form_003	Manage Patient Account	The user leaves some required fields empty and clicks on "Save " button	The system prompts the User to fill in the required fields	Pass
TC_save_patient_form_004	Manage Patient Account	The User fills all the fields in the form and clicks on "Save" button	The system confirms that the patient account is created by adding the new patient to the existing patient list	Pass
TC_view_patient_list_005	Manage Patient Account	The User clicks on "View Patient" button	The system displays a list of existing patient profiles	Pass
TC_view_patient_details_006	Manage Patient Account	The User clicks on the patient's first name	The system displays complete profile of the selected patient in tabs	Pass

TC_edit_patient_profile_007	Manage Patient Account	The User clicks on "Edit" profile button	The system displays a pre-populated profile that the User can edit	Pass
TC_empty_edit_patient_profile_008	Manage Patient Account	The User leaves some required fields empty and clicks on "Save " button	The system prompts the User to fill in the required fields	Pass
TC_save_edit_patient_profile_009	Manage Patient Account	The User fills all the fields in the form and clicks on "Save" button	The system confirms that the patient account is edited by adding the new patient to the existing patient list	Pass
TC_display_appointment_form_001	Manage Appointments	The User clicks on " Add Appointment" button	The system displays the Add New Appointment Form	Pass
TC_empty_appointment_form_002	Manage Appointments	The User leaves some required fields empty and clicks on "Save " button	The system prompts the User to fill in the required fields	Pass
TC_save_appointment_form_03	Manage Appointments	The User fills in all the required fields in the appointment form and clicks on "Save" button	The system confirms that appointment is saved	Pass
TC_edit_appointment_004	Manage Appointments	The User clicks on "Edit" button and makes the required changes and clicks on "Save " button	The system confirms that appointment is saved	Pass
TC_display_vitalsigns_form_001	Add Vital Signs	The User clicks on "Add Vital Signs/Diagnosis button	The system displays Add Vitals Signs form	Pass
TC_empty_vitalsigns_form_002	Add Vitals Signs	The User leaves some or all of the required fields empty in the Add Vitals Signs Form and clicks on the "Save" button	The system prompts the User to fill in the required fields	Pass
TC_save_vitalsigns_form_003	Add Vitals Signs	The User fills all the required fields in the form and clicks on "Save" button	The system confirms that the vitals are saved for the patient by displaying a list	Pass
TC_view_vitalsigns_list_004	Add Vital Signs	The User clicks on the patient	The system displays the vitals signs list	Pass

		name in the list vitals signs list		
TC_view_vitalsigns_details_005	Add Vital Signs	The User clicks on the patient name in the list vitals signs details	The system displays the vitals signs details	Pass
TC_display_diagnosis_form_001	Add Diagnosis	The User clicks on "Add Diagnosis" button in the vital sign details page	The system displays a Add Diagnosis Form	Pass
TC_empty_diagnosis_form_002	Add Diagnosis	The User fails to fill all or some of the required fields in the Add Diagnosis Form and clicks on "Save" button	The system prompts the User to fill in all the required fields	Pass
TC_save_diagnosis_form_003	Add Diagnosis	The User fills all the fields in the form and clicks on "Save" button	The system confirms that the diagnosis for the patient is saved	Pass
TC_display_prescription_form_001	Manage Prescriptions	The User clicks on "Add Prescription" button	The system displays a Prescription form	Pass
TC_empty_prescription_form_002	Manage Prescriptions	The User leaves some or all of the required fields empty in the Prescription Form and clicks on the "Save" button	The system prompts the User to fill in the required fields of the Prescription form	Pass
TC_save_prescription_form_003	Manage Prescriptions	The User fills all the fields in the form and clicks on "Save" button	The system confirms that the prescription is saved	Pass
TC_view_prescription_list_004	Manage Prescriptions	The User clicks on "View Prescription" button	The system displays a list of all the existing prescriptions	Pass
TC_view_prescription_details_005	Manage Prescriptions	The User clicks on the patient name	The system displays prescription details	Pass
TC_display_radiology_form_001	Manage Radiology	The User clicks on "Add Radiology" button	The system displays Add Radiology form	Pass
TC_empty_radiology_form_002	Manage Radiology	The User leaves some or all required fields in the form and	The system prompts the User to fill in the required fields	Pass

		clicks on "Save" button		
TC_save_radiology_form_003	Manage Radiology	The User fills the required fields and selects a file to be uploaded and clicks on "Save" button	The system confirms that the radiology film is added by displaying the list of all the radiology films	Pass
TC_view_radiology_list_004	Manage Radiology	The User clicks on the "View Radiology" button	The system displays the radiology list	Pass
TC_display_lab_test_order_form_001	Manage Lab Test	The User clicks on "Order Lab Test "	The system displays a Lab Test form	Pass
TC_empty_lab_test_order_form_002	Manage Lab Test	The User leaves some or all of the required fields empty and clicks on "Save" button	The system prompts the User to fill in all the required fields	Pass
TC_save_lab_test_order_form_003	Manage Lab Test	The User fills the lab test form and clicks on "Save" button	The system confirms that the newly added lab test is saved	Pass
TC_view_lab_test_order_list_004	Manage Lab Test	The user clicks on "View Lab Test Order" button	The system displays a list of all the lab test orders placed	Pass
TC_display_lab_result_form_001	Manage Lab Result	The User selects the "Add Lab Result" button	The system displays the Lab Result form	Pass
TC_save_lab_result_form_002	Manage Lab Result	The User clicks on the "Save" button	The system confirms that the lab result has been saved	Pass
TC_view_lab_result_list_003	Manage Lab Result	The User clicks on "View Lab Result" button	The system displays a Lab Test Result List	Pass
TC_patient_summary_001	View Summary	The User navigates to Appointment list page, selects a patient and clicks on "summary" button	The system displays the visit summary for the specific patient	Pass
TC_staff_login_user_001	Login User	The User clicks on "Login" button	The system displays the login page with two fields for Username and Password	Pass
TC_staff_login_success_001	Login Validation	The User enters correct Username and password	The system displays the User's Homepage	Pass
TC_staff_login_empty_	Login Validation	The User leaves	The system prompts	Pass

002		one or both of the fields empty	the User to fill in the required fields	
TC_staff_login_incorrect_003	Login Validation	The User enters incorrect Username and Password	The system displays an invalid login page and prompts the User to re-enter the correct Username and Password	Pass
TC_staff_logout_user_001	Logout User	The User clicks on "Logout" button	The system displays a confirmation message that the User has been logged out of the application	Pass
TC_staff_display_patient_form_001	Manage Patient Account	The User clicks on "Add New Patient " button	The system displays New Patient Information Forms in tabs	Pass
TC_staff_display_next_patient_form_002	Manage Patient Account	The User fills the form and clicks on "Next" button	The system displays the form in the next tab	Pass
TC_staff_empty_patient_form_003	Manage Patient Account	The user leaves some required fields empty and clicks on "Save " button	The system prompts the User to fill in the required fields	Pass
TC_staff_save_patient_form_004	Manage Patient Account	The User fills all the fields in the form and clicks on "Save" button	The system confirms that the patient account is created by adding the new patient to the existing patient list	Pass
TC_staff_view_patient_list_005	Manage Patient Account	The User clicks on "View Patient" button	The system displays a list of existing patient profiles	Pass
TC_staff_view_patient_details_006	Manage Patient Account	The User clicks on the patient's first name	The system displays complete profile of the selected patient in tabs	Pass
TC_staff_edit_patient_profile_007	Manage Patient Account	The User clicks on "Edit" profile button	The system displays a pre-populated profile that the User can edit	Pass
TC_staff_empty_edit_patient_profile_008	Manage Patient Account	The User leaves some required fields empty and clicks on "Save " button	The system prompts the User to fill in the required fields	Pass
TC_staff_save_edit_patient_profile_009	Manage Patient Account	The User fills all the fields in the form and clicks on "Save" button	The system confirms that the patient account is edited by adding the new patient to the existing patient list	Pass
TC_staff_display_appointment_form_001	Manage Appointments	The User clicks on " Add Appointment"	The system displays the Add New Appointment Form	Pass

		button		
TC_staff_empty_appoin tment_form_002	Manage Appointments	The User leaves some required fields empty and clicks on "Save " button	The system prompts the User to fill in the required fields	Pass
TC_staff_save_appoint ment_form_03	Manage Appointments	The User fills in all the required fields in the appointment form and clicks on "Save" button	The system confirms that appointment is saved	Pass
TC_staff_edit_appointm ent_004	Manage Appointments	The User clicks on "Edit" button and makes the required changes and clicks on "Save " button	The system confirms that appointment is saved	Pass
TC_staff_display_vitalsi gns_form_001	Add Vital Signs	The User clicks on "Add Vital Signs/Diagnosis button	The system displays Add Vitals Signs form	Pass
TC_staff_empty_vitalsig ns_form_002	Add Vitals Signs	The User leaves some or all of the required fields empty in the Add Vitals Signs Form and clicks on the "Save" button	The system prompts the User to fill in the required fields	Pass
TC_staff_save_vitalsign s_form_003	Add Vitals Signs	The User fills all the required fields in the form and clicks on "Save" button	The system confirms that the vitals are saved for the patient by displaying a list	Pass
TC_staff_view_vitalsign s_list_004	Add Vital Signs	The User clicks on the patient name in the list vitals signs list	The system displays the vitals signs list	Pass
TC_staff_view_vitalsign s_details_005	Add Vital Signs	The User clicks on the patient name in the list vitals signs details	The system displays the vitals signs details	Pass
TC_staff_display_radiol ogy_form_001	Manage Radiology	The User clicks on "Add Radiology" button	The system displays Add Radiology form	Pass
TC_staff_empty_radiolo gy_form_002	Manage Radiology	The User leaves some or all required fields in the form and clicks on "Save" button	The system prompts the User to fill in the required fields	Pass

TC_staff_save_radiolog y_form_003	Manage Radiology	The User fills the required fields and selects a file to be uploaded and clicks on "Save" button	The system confirms that the radiology film is added by displaying the list of all the radiology films	Pass
TC_staff_view_radiolog y_list_004	Manage Radiology	The User clicks on the "View Radiology" button	The system displays the radiology list	Pass
TC_staff_display_lab_re sult_form_001	Manage Lab Result	The User selects the "Add Lab Result" button	The system displays the Lab Result form	Pass
TC_staff_save_lab_res ult_form_002	Manage Lab Result	The User clicks on the "Save" button	The system confirms that the lab result has been saved	Pass
TC_staff_view_lab_resu lt_list_003	Manage Lab Result	The User clicks on "View Lab Result" button	The system displays a Lab Test Result List	Pass
TC_staff_patient_summ ary_001	View Summary	The User navigates to Appointment list page, selects a patient and clicks on "summary" button	The system displays the visit summary for the specific patient	Pass
TC_patient_login_user_ 001	Login User	The User clicks on "Login" button	The system displays the login page with two fields for Username and Password	Pass
TC_patient_login_succe ss_001	Login Validation	The User enters correct Username and password	The system displays the User's Homepage	Pass
TC_patient_login_empt y_002	Login Validation	The User leaves one or both of the fields empty	The system prompts the User to fill in the required fields	Pass
TC_patient_login_incorr ect_003	Login Validation	The User enters incorrect Username and Password	The system displays an invalid login page and prompts the User to re-enter the correct Username and Password	Pass
TC_patient_logout_user _001	Logout User	The User clicks on "Logout" button	The system displays a confirmation message that the User has been logged out of the application	Pass
TC_view_personal_profi le_001	View Profile	The User clicks on "profile" button	The system displays the profile of the User who is signed in to the application	Pass

TC_view_personal_appointment_001	View Appointment	The User clicks on "Appointment" button	The system displays the list of all the appointments of the User who is signed in to the application	Pass
TC_view_personal_prescription_001	View Prescriptions	The User clicks on the "Prescriptions" button	The system displays a list of all prescriptions received by the User who is signed into the application	Pass
TC_view_personal_diagnosis_001	View Diagnosis	The User clicks on the "Diagnosis" button	The system displays a list of all diagnosis received by the User who is signed into the application	Pass
TC_view_personal_lab_test_order_001	View Lab Test	The User clicks on "Lab Test" button	The system displays a list of all the lab tests ordered for the User who is signed into the application	Pass
TC_view_personal_vitals_chart_001	View Vitals Charts	The User clicks on the "Vitals Charts" button	The system displays the vitals charts	Pass

3. Test Deliverables

3.1 Test Plan

Traceability Table

Test Case	Shall
TC_login_user_001	Shall_3.1.1.0
TC_login_success_001	Shall_3.1.1.1
TC_login_empty_002	Shall_3.1.1.2
TC_login_incorrect_003	Shall_3.1.1.3
TC_logout_user_001	Shall_3.1.2.0
TC_display_patient_form_001	Shall_3.1.3.0
TC_display_next_patient_form_002	Shall_3.1.3.1
TC_empty_patient_form_003	Shall_3.1.3.2
TC_save_patient_form_004	Shall_3.1.3.3
TC_view_patient_list_005	Shall_3.1.3.4
TC_view_patient_details_006	Shall_3.1.3.5

TC_edit_patient_profile_007	Shall_3.1.3.6
TC_empty_edit_patient_profile_008	Shall_3.1.3.7
TC_save_edit_patient_profile_009	Shall_3.1.3.8
TC_display_appointment_form_001	Shall_3.1.4.0
TC_empty_appointment_form_002	Shall_3.1.4.1
TC_save_appointment_form_03	Shall_3.1.4.2
TC_edit_appointment_004	Shall_3.1.4.3
TC_display_vitalsigns_form_001	Shall_3.1.5.0
TC_save_vitalsigns_form_003	Shall_3.1.5.1
TC_empty_vitalsigns_form_002	Shall_3.1.5.2
TC_view_vitalsigns_list_004	Shall_3.1.6.0
TC_view_vitalsigns_details_005	Shall_3.1.6.1
TC_display_diagnosis_form_001	Shall_3.1.7.0
TC_empty_diagnosis_form_002	Shall_3.1.7.1
TC_save_diagnosis_form_003	Shall_3.1.7.2
TC_display_prescription_form_001	Shall_3.1.8.0
TC_empty_prescription_form_002	Shall_3.1.8.1
TC_save_prescription_form_003	Shall_3.1.8.2
TC_view_prescription_list_004	Shall_3.1.8.3
TC_view_prescription_details_005	Shall_3.1.8.4
TC_display_radiology_form_001	Shall_3.1.9.0
TC_empty_radiology_form_002	Shall_3.1.9.1
TC_save_radiology_form_003	Shall_3.1.9.2
TC_view_radiology_list_004	Shall_3.1.9.3
TC_display_lab_test_order_form_001	Shall_3.1.10.0
TC_empty_lab_test_order_form_002	Shall_3.1.10.1
TC_save_lab_test_order_form_003	Shall_3.1.10.2
TC_view_lab_test_order_list_004	Shall_3.1.10.3
TC_display_lab_result_form_001	Shall_3.1.11.0
TC_save_lab_result_form_002	Shall_3.1.11.1
TC_view_lab_result_list_003	Shall_3.1.11.2
TC_patient_summary_001	Shall_3.1.12.0
TC_staff_login_user_001	Shall_3.1.13.0
TC_staff_login_success_001	Shall_3.1.13.1
TC_staff_login_empty_002	Shall_3.1.13.2
TC_staff_login_incorrect_003	Shall_3.1.13.3

TC_staff_logout_user_001	Shall_3.1.14.0
TC_staff_display_patient_form_001	Shall_3.1.15.0
TC_staff_display_next_patient_form_002	Shall_3.1.15.1
TC_staff_empty_patient_form_003	Shall_3.1.15.2
TC_staff_save_patient_form_00	Shall_3.1.15.3
TC_staff_view_patient_list_005	Shall_3.1.15.4
TC_staff_view_patient_details_006	Shall_3.1.15.5
TC_staff_edit_patient_profile_007	Shall_3.1.15.6
TC_staff_empty_edit_patient_profile_008	Shall_3.1.15.7
TC_staff_save_edit_patient_profile_009	Shall_3.1.15.8
TC_staff_display_appointment_form_001	Shall_3.1.16.0
TC_staff_empty_appointment_form_002	Shall_3.1.16.1
TC_staff_save_appointment_form_03	Shall_3.1.16.2
TC_staff_edit_appointment_004	Shall_3.1.16.3
TC_staff_display_vitalsigns_form_001	Shall_3.1.17.0
TC_staff_empty_vitalsigns_form_002	Shall_3.1.17.1
TC_staff_save_vitalsigns_form_003	Shall_3.1.17.2
TC_staff_view_vitalsigns_list_004	Shall_3.1.18.0
TC_staff_view_vitalsigns_details_005	Shall_3.1.18.1
TC_staff_display_radiology_form_001	Shall_3.1.19.0
TC_staff_empty_radiology_form_002	Shall_3.1.19.1
TC_staff_save_radiology_form_003	Shall_3.1.19.2
TC_staff_view_radiology_list_004	Shall_3.1.19.3
TC_staff_display_lab_result_form_001	Shall_3.1.20.0
TC_staff_save_lab_result_form_002	Shall_3.1.20.1
TC_staff_view_lab_result_list_003	Shall_3.1.20.2
TC_staff_patient_summary_001	Shall_3.1.21.0
TC_patient_login_user_001	Shall_3.1.22.0
TC_patient_login_success_001	Shall_3.1.22.1
TC_patient_login_empty_002	Shall_3.1.22.2
TC_patient_login_incorrect_003	Shall_3.1.22.3
TC_patient_logout_user_001	Shall_3.1.23.0
TC_view_personal_profile_001	Shall_3.1.24.0
TC_view_personal_appointment_001	Shall_3.1.25.0
TC_view_personal_prescription_001	Shall_3.1.26.0
TC_view_personal_diagnosis_001	Shall_3.1.27.0

TC_view_personal_lab_test_order_001	Shall_3.1.28.0
TC_view_personal_vitals_chart_001	Shall_3.1.29.0

3.2. Test Cases and Reports

Refer to Attachment "A" for Pylint

Refer to Attachment "B" for Unittest

Refer to Attachment "C" for Selenium IDE

Refer to Attachment "D" for Selenium Scripts

Refer to Attachment "E" for Code Coverage Report

4. Features to be Tested

All the functional requirements listed in the software requirement specification document for OneClickVitals will be tested for completion and meeting the stated requirements. This will be done by unit testing, manual testing and user acceptance test.

The features to be tested include:

- Login User

- Logout User

- Manage Patient Account

- Manage Appointments

- Add Vital Signs

- View Vital Signs

- Add Diagnosis

- Add Prescriptions

- Manage Radiology

- Order Lab Tests

- Add Lab Results

- Print Summary

- View Personal Profile

- View Appointments

- View Prescriptions

- View Diagnosis

- View Lab Tests

- View Vital Signs charts

5. Features not to be Tested

Non-functional requirements listed in the software requirement specification document for OneClickVitals will not be tested as a part of this Test Plan.

6. Risks and Mitigation

Risk Factors	Description	Risk Level	Mitigation Plan
Resource risk	A team member might be sick for more than one day	Medium	The team will ensure that every member of the team is well aware of what other members are doing so he/she can take over the gap created by the absent member.
Resource risk	Team member might not be contribute as per expectations and requirements	High	The team will ensure proper delegation of work and have clearly defined roles and responsibilities. Also conduct regular status meetings.
Technical knowledge risk	Some or all the team members might not have the knowledge and experience is using the proposed tools and technology	High	Proper training sessions should be conducted and the same should be included in project budget and schedule.

Schedule risk	Team might not finish the project on time	High	The team should periodically check the status of the project and modules being implemented against the schedule.
Project Scope Risk	Scope of the project is not defined clearly and is subject constant changes	High	The team has to agree on the scope of the project prior to implementation. Team should have a contract signed by the client agreeing to the scope of the requirements and the project.

7. References

- Pressman, Roger S. *Software Engineering: A Practitioner's Approach*. 7th ed. New York: McGraw-Hill, 2010. Print.
- Futrell, Robert T., Donald F. Shafer, and Linda I. Shafer. *Quality Software Project Management*. Upper Saddle River, NJ: Prentice Hall PTR, 2002. Print.
- Larman, Craig. *Applying Uml and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. Upper Saddle River, N.J: Prentice Hall PTR, c2005, 2005. Print.
- IEEE Recommended Practice for Software Requirements Specifications," *IEEE Std 830-1998* , vol., no., pp.1,40, Oct. 20 1998