**TEST PLAN**

**1. INTRODUCTION**

**1.1. Purpose**

This test plan describes the testing approach and overall framework that will drive the testing Outpatient Records Management System.

It includes:

Test Strategy: this refers to the rules that the project testing will be based on given the start date and end date. These could include, Objectives, assumptions, creation of test cases, specific tasks to perform, scheduling, data strategy e.t.c.

Execution Strategy: describes how the test will be performed and process to identify and report

Defects, and to fix and implement fixes.

Test Management: process to handle the logistics of the test and all the events that come up

During execution (e.g.: communications, escalation procedures, risk and mitigation, team roster).

**1.2. Audience**

Project team members perform tasks specified in this document, and provide input and recommendations on this document.

Project Manager Plans for the testing activities in the overall project schedule, reviews the document, tracks the performance of the test according to the task herein specified, approves the document and is accountable for the results.

The stakeholders’ representatives and participants may take part in the testing to ensure the medical operations is aligned with the results of the test.

Technical Team ensures that the test plan and deliverables are in line with the design, provides the environment for testing and follows the procedures related to the fixes of defects.

The Medical Department officials will provide their inputs on functional changes.

**2. TEST STRATEGY**

**2.1. Test Objectives**

The objective of the test is to verify that the functionality Outpatient Records management System works according to the specifications.

The test will execute and verify the test scripts, identify, fix and retest all high and medium severity defects per the entrance criteria, prioritize lower severity defects for future fixing.

**2.1. Test Assumptions**

Production like data required and be available in the system prior to start of Functional Testing

Exploratory Testing would be carried out once the build is ready for testing.

Project Manager/BUSINESS ANALYST will review and sign-off all test deliverables.

The project will provide test planning, test design and test execution support.

Project team has the knowledge and experience necessary, or has received adequate training in the system, the project and the testing processes.

The system will be treated as a black box; if the information shows correctly online and in the reports, it will be assumed that the database is working properly.

During Functional testing, testing team will use preloaded data which is available on the system at the time of execution.

Users have basic computer knowledge.

**2.3. Test Principles**

Testing will be focused on meeting the business objectives, cost efficiency, and quality.

There will be common, consistent procedures for all teams supporting testing activities.

Testing processes will be well defined, yet flexible, with the ability to change as needed.

Testing activities will build upon previous stages to avoid redundancy or duplication of effort.

Testing environment and data will emulate a production environment as much as possible.

Testing will be a repeatable, quantifiable, and measurable activity.

Testing will be divided into distinct phases, each with clearly defined objectives and goals.

There will be entrance and exit criteria.

**2.4. Data Approach**

In functional testing, the system will contain pre-loaded test data and which is used for testing activities.

**2.5. Scope and Levels of Testing**

**2.5.1. Functional Test**

Functional testing will be performed to check the functions of application. The functional testing is carried out by feeding the input and validates the output from the application.

The test will be performed according to functional requirements of the system.

In our system, ORMS this test will be done based on the existing preloaded test data.

**2.5.2. System Testing**

System testing of software will be done on a complete, integrated system to determine if it meets the specified requirements.

**2.5.2. Integration Testing**

This test is carried out to confirm if the various parts of the system are working together collaboratively. This will be done to ascertain that enough connection and integration of various system users was integrated.

**2.5.2. Regression Testing**

This will test to see if any component which was added to the system has any effect on the existing ones. Especially components pertaining non-functional requirements.

**2.5.3. User Acceptance Testing**

This test focuses on validating the business logic. It allows the end users to complete one final review of the system prior to deployment.

It is performed by the end users.

Since the users are the most indicated to provide input around business needs and how the system adapts to them, it may happen that the users do some validation not contained in the scripts. Test team write the UAT test cases based on the inputs from end user and Business Analyst’s.

**2.6. Testing Deliverables**

|  |  |  |
| --- | --- | --- |
| **Deliverable Name** | **Author** | **Reviewer** |
| Test plan | Test lead | Project manager, system analyst |
| Functional test cases | Test team | System analyst |
| Logging details | Test team | Programmer |
| Report generation | Test team | Test lead, project manager |
|  |  |  |
|  |  |  |
|  |  |  |

**2.7. Test Environment**

|  |  |
| --- | --- |
| Application url | http://127.0.0.1:8080 |
| App server | local host |
| operating system | Windows, Linux, MAC |
| web browser | Mozilla Firefox, Chrome, UC browser |
| Database | MongoDb |

**2.7. Level and Impacts of the Defects**

|  |  |
| --- | --- |
| **Severity** | **Impact** |
| Critical | This bug is critical enough to crash the system, cause file corruption, or cause potential data loss.  It causes an abnormal return to the operating system (crash or a system failure message appears).  It causes the application to hang and requires re-booting the system. |
| High | It causes a lack of vital program functionality with workaround. |
| Medium | This Bug will degrade the quality of the System. However there is an intelligent workaround for achieving the desired functionality.  This bug prevents other areas of the product from being tested.  However other areas can be independently tested. |
| Low | There is an insufficient or unclear error message, which has minimum impact on system use. |
| Cosmic | There is an insufficient or unclear error message that has no impact on system use. |

**3. Role Expectations**

**3.1. Project Management**

Project Manager: reviews the content of the Test Plan, Test Strategy and Test Estimates signs off on it.

**3.2. Test Planning (Test Lead)**

Ensure entrance criteria are used as input before start the execution.

Develop test plan and the guidelines to create test conditions, test cases, expected results and execution scripts.

Provide guidelines on how to manage defects.

Attend status meetings in person or via the conference call line.

Communicate to the test team any changes that need to be made to the test deliverables or application and when they will be completed.

Provide on premise or telecommute support.

Provide functional (Business Analysts) and technical team to test team personnel (if needed).

**3.3. Test Team**

Develop test conditions, test cases, expected results, and execution scripts.

Perform execution and validation.

Identify, document and prioritize defects according to the guidance provided by the Test lead.

Re-test after software modifications have been made according to the schedule.

Prepare testing metrics and provide regular status.

**3.4. Test Lead**

Acknowledge the completion of a section within a cycle.

Give the OK to start next level of testing.

Facilitate defect communications between testing team and technical / development team.

**3.5. Development Team**

Review testing deliverables (test plan, cases, expected results, etc.) and provide timely feedback.

Assist in the validation of results (if requested).

Support the development and testing processes being used to support the project.

Certify correct components have been delivered to the test environment at the points specified in the testing schedule.

Keep project team and leadership informed of potential software delivery date slips based on the current schedule.

Define processes/tools to facilitate the initial and ongoing migration of components.

Conduct first line investigation into execution discrepancies and assist test executors in creation of accurate defects.

Implement fixes to defects according to schedule.