|  |
| --- |
|  |
| System Test Plan for |
|  |
|  |
|  |
|  |

|  |
| --- |
|  |

**Table of Contents**

[1. Introduction 4](#_Toc352754594)

[**1.1.** Purpose The System Test Plan (STP) covers the following points: 4](#_Toc352754595)

[2. Acronyms 5](#_Toc352754596)

[3. Scope of Testing 6](#_Toc352754597)

[3.1 Features to be tested 6](#_Toc352754598)

[3.2 Features not to be tested 7](#_Toc352754599)

[4. Entry and Exit Criteria 8](#_Toc352754600)

[5. Test Environment 9](#_Toc352754601)

[5.1 Hardware Requirements 9](#_Toc352754602)

[5.2 Software Requirements 9](#_Toc352754603)

[6. Test Strategy 10](#_Toc352754604)

[Testing Process 10](#_Toc352754605)

[Build Process 10](#_Toc352754606)

[Types of testing 10](#_Toc352754607)

[7. Tools 11](#_Toc352754608)

[8. Resource Management 12](#_Toc352754609)

[8.1 Roles and Responsibilities 12](#_Toc352754610)

[8.2 Test Schedule 12](#_Toc352754611)

[9. Assumptions 13](#_Toc352754612)

[10. Risks & Issues 14](#_Toc352754613)

[11. Test Deliverables 15](#_Toc352754614)

[12. Appendix 16](#_Toc352754615)

[12.1 Defect Classification 16](#_Toc352754616)

Version

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Description |
| November 23, 2012 | 0.1 | QA CoE | Initial Version |
| November 24, 2012 | 1.0 | QA CoE | Final Version after review |

Reviewed By

|  |  |
| --- | --- |
| Date | Name |
| November 24, 2012 | Jaideep S |

# Introduction

This document is a procedural guide for listing the testing activities that needed to be carried out for the <<PROJECT NAME>> Project. It describes the software test environment to be used for the testing, identifies the tests to be performed, the deliverables and provides schedules for the test activities.

## Purpose The System Test Plan (STP) covers the following points:

* + Identify all the activities involved in testing
  + Training of resources required for executing test activities and monitoring mechanisms
  + The strategy and approach of testing the software components along with risk and mitigation plan
  + The defect tracking system to be used with the help of various tools

# Acronyms

This section provides the description of acronym that is required to interpret the STP properly.

| S. No. | Acronym | Description |
| --- | --- | --- |
|  | STP | System Test Plan |
|  | CCB | Change Control Board |
|  | QA | Quality Assurance |

Add all acronyms to be used in the document.

# Scope of Testing

## Features to be tested

Module 1: <<Module Name>>

| S. No. | Features to be tested |
| --- | --- |
|  |  |
|  |  |
|  |  |

Module 2: <<Module Name>>

| S. No. | Features to be tested |
| --- | --- |
|  |  |
|  |  |

Module 3: <<Module Name>>

| S. No. | Features to be tested |
| --- | --- |
| 1. |  |
| 2. |  |

## Features not to be tested

Module 1: <<Module Name>>

| S. No. | Features not to be tested |
| --- | --- |
| 1. |  |

Module 2: <<Module Name>>

| S. No. | Features not to be tested |
| --- | --- |
| 1. |  |

Miscellaneous:

| S. No. | Activities |
| --- | --- |
| 1. |  |

# Entry and Exit Criteria

<<Entry and Exit criteria should be defined here

Entry Criteria Example:

1. Test case preparation to start after business requirement sign off.
2. Release notes to be published by development team before test execution.

Exit Criteria Example:

1. All high priority and severity defects are closed.
2. All Test cases are executed at least once.

>>

# Test Environment

## Hardware Requirements

| S. No. | Purpose | Configuration | Quantity |
| --- | --- | --- | --- |
|  | Client | IBM Compatible PC with at least 10 GB HDD, 512 MB RAM | 1 |
|  | Server | IBM Compatible PC with at least 4 GB HDD, 256 MB RAM | 1 |

## Software Requirements

| S. No. | Name of the Software | Version/Release details | No. Of licenses |
| --- | --- | --- | --- |
| Server Side | | | |
|  | SQL Server with SQL Server Reporting Service | 2000, with latest service pack | NA |
|  | Windows OS | 2000 professional, Latest SP | 1 |
|  | MS. NET framework | 1.1 | NA |
|  | MQ Series client |  |  |
| Client Side | | | |
| 1. | Windows OS | XP | 1 |
| 2. | MS .NET framework without Visual Studio | 1.1 | NA |
| 3. | MS Excel | 2000/XP | 1 |
| 4. | MS Access | 97 | 1 |

# Test Strategy

Testing Process  
This process details the testing process that will be followed during the course of the project execution. The testing process is broken down into test planning, test design, test execution and test reporting. The testing process to be followed for <<PROJECT NAME>> is as follows:

Build Process  
<<Define the Build process to be followed during the project>>

Types of testing  
Different types of testing that should be carried out in the project are as follows:

* **Smoke testing:** A set of test cases/ checklist will be identified and documented as smoke test cases. With every build, the Tester will execute these test cases to determine if the build is stable for further testing.
* **Integration testing:** Ensure that different modules function appropriately with the integrated build.
* **System Testing:** All the test cases documented will be executed for each build to ensure the system as a whole has been tested.
* **Regression Testing:** Performed whenever there is a change in the code to ensure that the new code does not impact the existing functionality.
* **Recovery Testing:** In case of abrupt system failure, verify that the recovery is properly performed. It is essential that data is recovered after a system failure & data corruption is not reported.
* **Security Testing:** Determine how secure the system is from unauthorized user access or hacking.

# Tools

<<This section will provide details about the various tools (Test Management Tools, Automation Tools, Defect Tracking Tool, etc.) used for Testing.>>

|  |  |  |
| --- | --- | --- |
| S. No. | Name of tool | Usage |
|  |  |  |
|  |  |  |

# Resource Management

## Roles and Responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| S. No. | Roles | Name of the Resource | Responsibilities |
|  | Project Manager |  | Responsible for all Project activities and managing project team. |
|  | Project Lead |  | Serve as a primary contact/liaison between the development team and the project test team. |
|  | Test Lead |  | -Responsible for all Testing activities  -Coordinate weekly meetings and will communicate the testing status to the project team  -Create and update the STP |
|  | Testers |  | Responsible for writing, executing test cases and reporting defects. |
|  | Developers |  | Develop the application and perform unit testing. |
|  | Business Analyst |  | Responsible for requirement gathering and related query resolution. |

## Test Schedule

<<Mention the planned Test Schedule here>>

# Assumptions

<<Mention project Assumptions and Dependencies here

Example:

1. Business Requirements are signed off before Test Case preparation start date
2. Test Environment should be ready and available prior to test execution activity.>>

# Risks & Issues

<<Mention project Risks and Issues here

Example:

1. Hardware not yet procured

2. Unavailability of domain specific resources >>

|  |  |  |  |
| --- | --- | --- | --- |
| Risk & Issue | Probability | Impact | Mitigation Plan |
|  |  |  |  |

# Test Deliverables

| S. No. | Artifact | Delivery Date | Owner | Frequency |
| --- | --- | --- | --- | --- |
|  | System Test Plan |  |  |  |
|  | Requirements Traceability Matrix |  |  |  |
|  | Test Design |  |  |  |
|  | Test Cases |  |  |  |
|  | Build Acceptance Report… |  |  |  |

# Appendix

## Defect Classification

Following are the Defect Priority levels:

1. **High:** The defect must be resolved as soon as possible preferably before next build because it is impairing development and / or testing activities. System use will be severely affected until the defect is fixed.
2. **Medium:** The defect should be resolved in the normal course of development activities. It can wait until a new build or version is created.
3. **Low:** The defect repair can be put of indefinitely. It can be resolved in a future major system revision or not resolved at all.

Following are the Defect Severity levels:

1. **Show Stopper:** The defect results in crash or shut down of the application. The user cannot use the application until the defect is resolved.
2. **Critical:** The defect results in the failure of the complete software system, of a subsystem, or of a software unit (program or module) within the system. There is no workaround.
3. **Major:** The defect does not result in a failure, but causes the system to produce incorrect, incomplete, or inconsistent results, or the defect impairs the systems usability. There is a workaround.
4. **Cosmetic:** The defect is the result of non-conformance to a standard, is related to the aesthetics of the system, or is a request for an enhancement. Mostly user interface issues. Defects at this level may be deferred.