



حکومت سندھ
Government of Sindh

TEST PLAN

E-Portal

11/24/2017

Revision History

Date	Version	Description	Author
24/November/2017	1.0 version	E-Portal Test Plan	Umair Saleem

Table of Contents

1. INTRODUCTION.....	5
1.1 PURPOSE.....	5
1.2 AUDIENCE	5
1.3 PROJECT OVERVIEW	6
1.4 SCOPE.....	7
2. TEST METHODOLOGY	9
2.1 OVERVIEW.....	9
2.2 UNIT TESTING (MULTIPLE).....	9
2.3 ITERATION/REGRESSION TESTING	9
2.4 FINAL RELEASE TESTING	10
2.5 TEST LEVELS	10
2.6 BUILD TESTS	10
2.6.1 <i>Build Acceptance Tests</i>	10
2.6.2 <i>Smoke Tests</i>	10
2.6.3 <i>Bug Regression Testing</i>	10
2.6.4 <i>Functionality Testing</i>	11
2.6.5 <i>System Testing</i>	11
2.6.6 <i>Performance Testing / Pen Testing</i>	11
2.6.7 <i>Security Testing / Validation</i>	11
2.6.8 <i>User Interface Testing</i>	11
3. TEST STRATEGY.....	12
3.1 TEST ASSUMPTIONS.....	12
3.2 TEST PRINCIPLES	12
3.3 DATA APPROACH.....	13
3.4 ENTRY AND EXIT CRITERIA	13
3.5 BUG SEVERITY AND PRIORITY	13
3.5.1 <i>Severity List</i>	14
3.5.2 <i>Priority List</i>	14
3.5.3 <i>Test Metrics</i>	14
3.6 BUG REPORTING.....	15
3.7 TOOLS	17
4. RESOURCE & ENVIRONMENT	ERROR! BOOKMARK NOT DEFINED.
4.1 ROLE EXPECTATIONS	17
4.2 TEST ENVIRONMENT.....	18
4.2.1 <i>Hardware</i>	18
4.2.2 <i>Software</i>	19
5. PROJECT MILESTONES.....	19
6. DELIVERABLES	19

7. PROJECT TASKS	20
8. APPROVALS	22

1. INTRODUCTION

1.1 Purpose

The purpose of this document is to outline the test strategy and overall test approach for the E-Portal site. This includes test methodologies, requirements, resources, testing deliverables and milestones.

This Test Plan document supports the following objectives:

- Identify existing project information and the software that should be tested.
- List the recommended test requirements (high level).
- Recommend and describe the testing strategies to be employed.
- Identify the required resources and provide an estimate of the test efforts.
- List the deliverable elements of the test activities.

The document introduces:

- **Test Strategy:** rules the test will be based on, including the givens of the project (e.g.: start / end dates, objectives, assumptions); description of the process to set up a valid test (e.g.: entry / exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).
- **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 (individuals as identified by the PMO Leads) may take part in the UAT test to ensure the business 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.

1.3 Project Overview

The main objective of E-portal Site is that the **Information, Science & Technology Department, Government of Sindh** would like to achieve by developing “**E-PORTAL**” is to provide improved services delivery to citizens to dissemination by giving access to the information. The following are the high level objectives that the **Information, Science & Technology Department, Government of Sindh** would like to cover from this project.

- Improved and more modern User Interface for all the departments of Government of Sindh.
- Enhance and create structured User Experience for the website visitors to retrieve available information like Jobs, Tenders, Official Gazette, Notifications, etc.
- The new E-Portal have extended capabilities to accommodate and handle higher web traffic on the individual portals.
- The visitors of the site will be able to view the site in either English, Urdu or Sindhi.
- The website for departments to be responsive to support multiple devices.
- The visitors on the portals to enter / submit forms online.
- The website visitors to communicate with departments via the ticketing system which will improve the response time.
- The information available on all the portals to be searchable and retrieval by the website visitors.
- The E-Portal will provide easy to use Content Management System to update the relevant information on their respective portals.
- The E-Portal will also provide cross departmental communication platform and will also enable the information to be syndicated on all portals from the centralized location.

1.4 Scope

The overall purpose of testing is to ensure the application meets all of its technical, functional and business requirements. The purpose of this document is to describe the overall test plan and strategy for testing the E-Portal application. The approach described in this document provides the framework for all testing related to this application. Individual test cases

will be written for each version of the application that is released. This Test Plan describes the integration and system tests that will be conducted on the E-Portal site following integration of the subsystems and components identified.

In Scope

The Test Plan defines the unit, integration, system, regression, and Client Acceptance testing approach. The test scope includes the following:

- Testing of all functional, application performance, security and use cases requirements
- Quality requirements
- End-to-end testing and testing of interfaces of all systems that interact with the E-Portal System.

Out of Scope

The following are considered out of scope for system Test Plan and testing scope:

- Functional requirements testing for systems outside E-Portal
- Testing of Business SOPs, disaster recovery and Business Continuity Plan.

The purpose of E-Portal site is to present detailed description of all Sindh government departments and feature of content management and publishing system. It is critical that all system and subsystem interfaces be tested as well as system performance at this early stage. Testing of system functionality and features will not be conducted on the prototypes.

The E-Portal site will have the following features that will be tested:

- Departments
 - Home
 - Pages
 - Announcements
 - Modules
 - Pictures
 - Videos
 - Projects
 - Suggestions/Complaints
 - Contact
- Provincial Announcements
- Tenders
- Jobs
- Pages
- News & Events
- Quick Links
- Complaint / Support Ticketing
- Admin
 - Menu
 - Pages
 - Content Revision Management
 - Media
 - Modules
 - Jobs
 - Tenders
 - Bid Evaluation Reports
 - News
 - Forms
 - User Management
 - Roles
 - Activity Logging
 - Communication Engine
 - Departments
 - Provincial Announcements
 - System Parameters
 - Basic Template Customization
 - Department Domain Mapping

Back-end Admin Portal

The Back-End Admin Portal will contain the following sections:

Menu

Admin user will be able to change and the menu on any given page using this section of the Admin Portal.

Pages

All the pages on the portal, content related to those pages, styling will be manageable from this section of the back-end.

Content Revision Management

Super Admin and Departmental Admin will be able to manage the following from this module on the back-end:

- Add, Edit, Remove Videos
- Publish or Un Publish Page/s
- Multi Level Menu Management
- Meta Data Management on Page Level
- Search Management

2. TEST METHODOLOGY

2.1 Overview

The below list is not intended to limit the extent of the test plan and can be modified to become suitable for the particular project.

The listing below identifies those items (use cases, functional requirements, and non-functional requirements) that have been identified as targets for testing. This list represents what will be tested.

2.2 Unit Testing (Multiple)

Unit testing will be conducted by the Developer during code development process to ensure that proper functionality and code coverage have been achieved by each developer both during coding and in preparation for acceptance into iterations testing.

2.3 Iteration/Regression Testing

During the repeated cycles of identifying bugs and taking receipt of new builds (containing bug fix code changes), there are several processes which are common to this phase across all projects. These include the various types of tests: functionality, performance, stress, configuration, etc. There is also the process of communicating results from testing and ensuring that new drops/iterations contain stable fixes (regression). The project should plan for a minimum of 2-3 cycles of testing (drops/iterations of new builds).

At each iteration, a debriefing should be held. Specifically, the report must show that to the best degree achievable during the iteration testing phase

2.4 Final release Testing

Testing team with end-users participates in this milestone process as well by providing confirmation feedback on new issues uncovered, and input based on identical or similar issues detected earlier. The intention is to verify that the product is ready for distribution, acceptable to the customer and iron out potential operational issues.

Assuming critical bugs are resolved during previous iterations testing- Throughout the Final Release test cycle, bug fixes will be focused on minor and trivial bugs. Testing will continue its process of verifying the stability of the application through regression testing (existing known bugs, as well as existing test cases).

2.5 Test Levels

Testing of an application can be broken down into three primary categories and several sub-levels. The three primary categories include tests conducted every build (Build Tests), tests conducted every major milestone (Milestone Tests), and tests conducted at least once every project release cycle (Release Tests). The test categories and test levels are defined below:

2.6 Build Tests

2.6.1 Build Acceptance Tests

Build Acceptance Tests should take less than 2-3 hours to complete (15 minutes is typical). These test cases simply ensure that the application can be built and installed successfully. Other related test cases ensure that adopters received the proper Development Release Document plus other build related information (drop point, etc.). The objective is to determine if further testing is possible. If any Level 1 test case fails, the build is returned to developers un-tested.

2.6.2 Smoke Tests

Smoke Tests should be automated and take less than 2-3 hours (20 minutes is typical). These tests cases verify the major functionality a high level.

The objective is to determine if further testing is possible. These test cases should emphasize breadth more than depth. All components should be touched, and every major feature should be tested briefly by the Smoke Test. If any Level 2 test case fails, the build is returned to developers un-tested.

2.6.3 Bug Regression Testing

Every bug that was “Open” during the previous build, but marked as “Fixed, Needs Re-Testing” for the current build under test, will need to be regressed, or re-tested. Once the smoke test is completed, all resolved bugs need to be regressed. It should take between 5 minutes to 1 hour to regress most bugs.

2.6.4 Functionality Testing

Functionality testing will be performed by the QA team to verify that a web application performs and functions correctly according to design specifications. During functionality testing we check the core application functions, text input, menu functions and installation and setup on localized machines, etc. The objective of this test is to ensure that each element of the application meets the functional requirements of the business as outlined in the: Description of Software Requirement Program.

2.6.5 System Testing

System Testing (ST) is a black box testing technique performed to evaluate the complete system the system's compliance against specified requirements. In System testing, the functionalities of the system are tested from an end-to- end perspective.

2.6.6 Performance Testing / Pen Testing

Performance testing, a non-functional testing technique performed to determine the system parameters in terms of responsiveness and stability under various workload. Performance testing measures the quality attributes of the system, such as scalability, reliability and resource usage. Load testing - It is the simplest form of testing conducted to understand the behavior of the system under a specific load. Load testing will result in measuring important business critical transactions and load on the database, application server, etc., are also monitored. Stress testing - It is performed to find the upper limit capacity of the system and also to determine how the system performs if the current load goes well above the expected maximum.

2.6.7 Security Testing / Validation

The purpose of the security test is to discover the vulnerabilities of the web application so that the developers can then remove these vulnerabilities from the application and make the web application and data safe from unauthorized actions.

2.6.8 User Interface Testing

User Interface testing is the process of ensuring proper functionality of the Graphical user interface (GUI) for a given application and making sure it Conforms to its written specifications. Following are aspects that will be checked during GUI testing.

- Verify ease of navigation through a sample set of screens.
- Verify sample screens conform to GUI standards.
- The System shall be easy-to-use and shall be appropriate for the anyone.
- The desktop user-interface shall be Windows 7 to 10 etc compliant.
- The System operation and actions shall be easy to understand and user friendly.
- The user interface of the System shall be designed for ease-of-use.

3. TEST STRATEGY

The following test strategy is generic in nature and is meant to apply to the requirements.

3.1 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
- All the defects would come along with a snapshot JPEG format
- The Test Team will be provided with access to Test environment
- The Test Team assumes all necessary inputs required during Test design and execution will be supported by Development/BUSINESS ANALYSTs appropriately.
- Test case design activities will be performed by QA Group
- Test environment and preparation activities will be owned by Dev Team
- BUSINESS ANALYST will review and sign-off all Test cases prepared by Test Team prior to start of Test execution
- The defects will be tracked through JIRA Tool only. Any defect fixes planned will be shared with Test Team prior to applying the fixes on the Test environment
- Project Manager/BUSINESS ANALYST will review and sign-off all test deliverables
- The project will provide test planning, test design and test execution support
- Test team will manage the testing effort with close coordination with Project PM/BUSINESS ANALYST
- Project team has the knowledge and experience necessary, or has received adequate training in the system, the project and the testing processes.

3.2 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.

3.3 Data Approach

- In functional testing, E-Portal site will contain pre-loaded test data and which is used for testing activities.

3.4 Entry and Exit Criteria

- The entry criteria refer to the desirable conditions in order to start test execution; only the migration of the code and fixes need to be assessed at the end of each cycle.
- The exit criteria are the desirable conditions that need to be met in order proceed with the implementation.
- Entry and exit criteria are flexible benchmarks. If they are not met, the test team will assess the risk, identify mitigation actions and provide a recommendation. All this is input to the project manager for a final “go-no go” decision.
- Entry criteria to start the execution phase of the test: the activities listed in the Test Planning section of the schedule are 100% completed.
- Entry criteria to start each cycle: the activities listed in the Test Execution section of the schedule are 100% completed at each cycle.

3.5 Bug Severity and Priority

Bug Severity and Priority fields are both very important for categorizing bugs and prioritizing if and when the bugs will be fixed. The bug Severity and Priority levels will be defined as outlined in the following tables below. Testing will assign a severity level to all bugs. The Test Lead will be responsible to see that a correct severity level is assigned to each bug.

The Test Lead, Development Lead and Program Manager will participate in bug review meetings to assign the priority of all currently active bugs. This meeting will be known as “Bug Triage Meetings”. The Test Lead is responsible for setting up these meetings on a routine basis to address the current set of new and existing but unresolved bugs.

3.5.1 Severity List

The tester entering a bug into JIRA is also responsible for entering the bug Severity.

Severity ID	Severity	Severity Description
1	Critical	The module/product crashes or the bug causes non-recoverable conditions. System crashes, GP Faults, or database or file corruption, or potential data loss.
2	High	Major system component unusable due to failure or incorrect functionality. They cause serious problems such as a lack of functionality, or insufficient or unclear error messages that can have a major impact to the user, prevents other areas of the app from being tested.
3	Medium	Incorrect functionality of component or process. There is a simple work around for the bug.
4	Minor	Documentation errors or signed off severity 3 bugs.

3.5.2 Priority List

Priority	Priority Level	Priority Description
5	Blocker	We cannot move forward until this task completed or this bug fixed.
4	Critical	this task or issue will most likely be resolved for release
3	Major	Default Issue should be resolved for release.
2	Minor	Issue might be resolved for release.
1	Trivial	Issue may or may not be resolved.

3.5.3 Test Metrics

Test metrics to measure the progress and level of success of the test will be developed and shared with the project manager for approval. The below are some of the metrics

Report	Description	Frequency
Test preparation & Execution Status	To report on % complete, % WIP, % Pass, % Fail Defects severity wise Status – Open, closed, any other Status	Weekly / Daily (optional)
Daily execution status	To report on Pass, Fail, Total defects, highlight Showstopper/ Critical defects	Daily
Project Weekly Status report	Project driven reporting (As requested by PM)	Weekly – If project team needs weekly update apart from daily and there is

		template available
--	--	--------------------

3.6 Bug Reporting

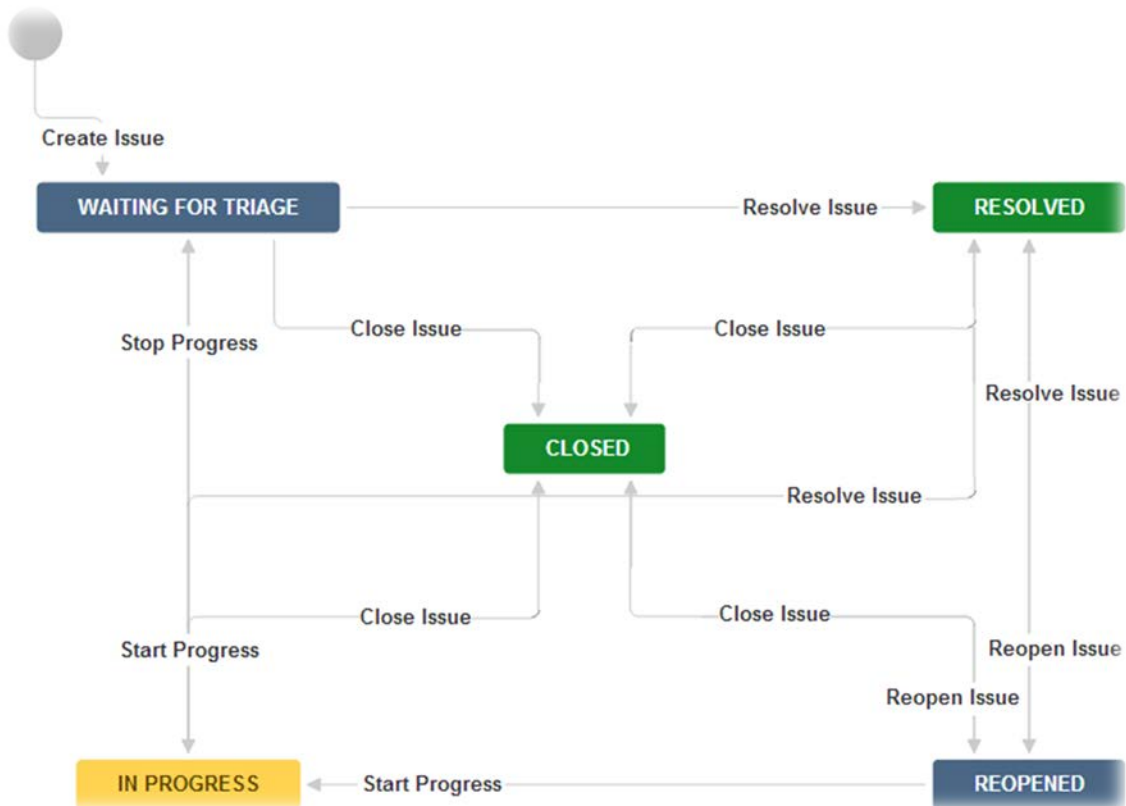
Testing team recognizes that the bug reporting process is a critical communication tool within the testing process. Without effective communication of bug information and other issues, the development and release process will be negatively impacted.

A major component of a bug tracking system is a database that records facts about known bugs. Facts may include the time a bug was reported, its severity, the erroneous program behavior, and details on how to reproduce the bug; as well as the identity of the person who reported it and any programmers who may be working on fixing it. The main benefit of a bug-tracking system is to provide a clear centralized overview of development requests (including both bugs and improvements, the boundary is often fuzzy), and their state. The prioritized list of pending items (often called backlog) provides valuable input when defining the product road map, or maybe just "the next release".

Typical bug tracking systems support the concept of the life cycle for a bug which is tracked through the status assigned to the bug. A bug tracking system should allow administrators to configure permissions based on status, move the bug to another status, or delete the bug. The system should also allow administrators to configure the bug statuses and to what extent a bug in a particular status can be moved.

The Test Lead will be responsible for managing the bug reporting process. Testing's standard bug reporting tools and processes will be used. JIRA is the company-wide standard Bug Logging / Tracking tool. Testing and development will enter their data into the JIRA tool.

Following flowchart depicts Defect Tracking Process:



3.7 Tools

The following tools will be employed for this project:

	Tool	Vendor/In-house
Test Management	JIRA	In-house
Defect Tracking	JIRA	In-house
Performance testing	Jmeter Tool	In-house
Test Coverage Monitor or Profiler	JIRA	In-house
Project Management	JIRA	In-house
DBMS tools	MySql	In-house

4. RESOURCE AND ENVIRONMENT

4.1 Role Expectations

The following list defines in general terms the expectations related to the roles directly involved in the management, planning or execution of the test for the project.

This section presents the recommended resources for the E-Portal application, their main responsibilities, and their knowledge or skill set.

Roles	Name	Contact Info
Project Manager	Agha Zeeshan	-
Test Lead	Umair Saleem	-
Business Analyst	Talha Hashmi	-
Development Lead	Muhammad Irfan	-
Testing Team	Umair Saleem , SM Anas	-
Development Team	Muhammad Irfan, Jerry Kazmi , Danial Hassan	

Project Management

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

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).

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.

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.

Development Team

- Review testing deliverables (test plan, cases, scripts, 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.

4.2 Test Environment**4.2.1 Hardware**

Include the minimum hardware requirements that will be used to test the Application.

Testing will have access control to one or more application/database servers separate from any used by non-test members of the project team. Testing will also have access control to an adequate number of variously configured PC workstations to assure testing a range from the minimum to the recommended client hardware configurations listed in the project’s Requirements, Functional Specification and Design Specification documents.

4.2.2 Software

In addition to the application and any other customer specified software, the following list of software should be considered a minimum:

- Windows 10 , OS X 10.7.
- MS Office 2017 Professional
- MS Exchange
- TCM (Testing Tool Server)
- Task Manager (Testing Tool Server)

5. PROJECT MILESTONES

Testing of E-Portal should incorporate test activities for each of the test efforts identified in the previous sections. Separate project milestones should be identified to communicate project status accomplishments.

Milestone Task	Effort	Start Date	End Date
Plan Test	1	24/11/2017	25/11/2017
Design Test	2	31/11/2017	10/12/2017
Implement Test	2	20/12/2017	28/12/2017
Execute Test	3	2/1/2018	15/1/2018
Evaluate Test	1	17/1/2018	25/1/2018
		-	-

6. DELIVERABLES

E-Portal team has responsibility for the following software testing deliverables:

Phase 1 Testing Deliverables:

- **Test Plan** November 2017

The develop team has responsibility for the following software testing deliverables:

Phase 1 Testing Deliverables:

- **Completion of Software Coding** December 2017
- **Completion of Unit, Integration & System Testing** December 2017
- **Integration Test Results Document** December 2017
- **Completion of Field Acceptance Testing** December 2017

Phase 2 Testing Deliverables:

- **Completion of Software Coding** December 2017
- **Completion of Unit, Integration & System Testing** December 2017
- **Integration Test Results Document** January 2018
- **Completion of Field Acceptance Testing** January 2018

7. PROJECT TASKS

Below are the test related tasks for testing the E-Portal Application:

Plan Test

- Identify Requirements for Test
- Assess Risk
- Develop Test Strategy
- Identify Test Resources
- Create Schedule
- Generate Test Plan

Design Test

- Workload Analysis (not applicable for Prototype)
- Develop Test Suite
- Identify and Describe Test Cases

- Identify and Structure Test Scripts
- Review and Access Test Coverage

Implement Test

- Setup Test Environment
- Identify Test-Specific functionality in the design and implementation model
- Establish External Data sets

Execute Test

- Execute Test Scripts
- Evaluate Execution of Test
- Recover from Halted Test
- Verify the results
- Investigate Unexpected Results
- Log Defects

Evaluate Test

- Evaluate Test-Case Coverage
- Evaluate Code Coverage
- Analyze Defects
- Determine if Test Completion Criteria and Success Criteria have been achieved
- Create Test Evaluation Report

8. APPROVALS

The Names and Titles of all persons who must approve this plan.

Signature:	
Name:	
Role:	
Date	

Signature:	
Name:	
Role:	
Date	

