# General Information

|  |  |
| --- | --- |
| **Project Name/Project #:** |  |
| **Business Process Owner/Sponsor:** |  |
| **Author:** |  |
| **Project End Date:** |  |

# Revision / Change History

|  |  |  |
| --- | --- | --- |
| **Revision Level** | **Revision Date** | **Description of Changes** |
| 01 | 4/28/2011 | 1. Template Version |
| 01 |  | 1. Initial Project Version |

The purpose of the Testing Strategy is to fully define the testing approach. The strategy will define which tests will be performed, to what extent the system will be tested, the high level plan or sequence of the testing and the key roles involved in fully testing the system.

The template includes instructions to the author, boilerplate text, and fields that should be replaced with the values specific to the particular project.

* Blue italicized text enclosed in square brackets (i.e., [text]) provides instructions to the document author, or describes the intent, assumptions and context for content included in this document.
* Blue text enclosed in angle brackets (i.e., <text>) indicates a field that should be replaced with information specific to the particular project.
* Text and tables in black are provided as boilerplate examples of wording and formats that may be used or modified as appropriate.

Contents

[General Information 1](#_Toc291753711)

[Revision / Change History 1](#_Toc291753712)

[1 OVERVIEW 3](#_Toc291753713)

[1.1 System Description 3](#_Toc291753714)

[1.2 System Organization 3](#_Toc291753715)

[2 ASSUMPTIONS/CONSTRAINTS/RISKS 3](#_Toc291753716)

[2.1 Assumptions 3](#_Toc291753717)

[2.2 Constraints 3](#_Toc291753718)

[2.3 Risks 3](#_Toc291753719)

[3 TESTING APPROACH/STRATEGY 3](#_Toc291753720)

[4 PLANNED TESTS 4](#_Toc291753721)

[5 TEST PROGRESSION 5](#_Toc291753722)

[6 FEATURE TESTING 5](#_Toc291753723)

[6.1 Features To Be Tested 5](#_Toc291753724)

[6.2 Features Not To Be Tested 5](#_Toc291753725)

[7 TEST CASES 6](#_Toc291753726)

[8 DEFECT TRACKING & REPORTING 6](#_Toc291753727)

[9 ENVIRONMENT 6](#_Toc291753728)

[9.1 Hardware 6](#_Toc291753729)

[9.2 Software 7](#_Toc291753730)

[9.3 Test Data 7](#_Toc291753731)

[9.4 Other Materials 7](#_Toc291753732)

[9.5 Installation, Testing, & Control 7](#_Toc291753733)

[9.6 Security 8](#_Toc291753734)

[10 SCHEDULES & MILESTONES 8](#_Toc291753735)

[11 ROLES & RESPONSIBILITIES 8](#_Toc291753736)

[11.1 Resources 8](#_Toc291753737)

[11.2 Orientation Plan 9](#_Toc291753738)

# OVERVIEW

*[Briefly describe the purpose and context for the system, and summarize the history of its development.]*

## System Description

*[Provide an overview of the processes that the system is intended to support. If applicable, provide a general description of the type of data maintained and the operational sources and uses of those data.**]*

## System Organization

*[Provide a brief description of the system architecture and the major system components essential to the testing. Describe hardware, software, and communications, as appropriate. Include any charts, diagrams, and/or graphics as necessary, with corresponding textual descriptions]*

# ASSUMPTIONS/CONSTRAINTS/RISKS

## Assumptions

*[If the testing approach/strategy is based upon any assumptions, list and describe them. For example, identify dependencies with other systems and the assumption that they will be ready to test when needed; assumptions regarding availability of defined test environments, etc.]*

## Constraints

*[Describe any limitations or constraints that have a significant impact on the testing of the system, application, or situation. Such constraints may be imposed by any of the following (the list is not exhaustive):*

1. *Hardware or software environment*
2. *End-user environment*
3. *Availability of resources*
4. *Interoperability requirements*
5. *Interface/protocol requirements*
6. *Data repository and data distribution requirements]*

## Risks

*[Identify and describe the potential problems or risk areas of the project and/or issues which may have an impact upon the testing effort. Some examples might include: system interfaces, highly complex software, system load issues, security, performance, and reliability. If any issues arise during the prescribed testing activities that lead to new risks, they should be documented with the overall project’s Risks.]*

# TESTING APPROACH/STRATEGY

*[Describe the overall approach that will be used to test all functions, features, and requirements of the automated system or application for which the Test Plan applies. As applicable to this Test Plan, describe the measures to be taken to ensure all aspects of the system are successfully tested and can be implemented. Document key aspects of the testing approach, such as content, methodology, prioritization, and progression of development, validation, implementation and operational testing activities to be performed during the corresponding lifecycle phases.*

*Describe the Testing Framework that will be applied to the project. For example, will some unit and application integration testing be done, and then some more development, and so on? Is a prototype being built that will be usability tested before the releasable software is developed? Also include plans for testing related documentation (e.g., installation instructions, User Manual, Operations & Maintenance (O&M) Manual, Training Artifacts, etc.) and for conducting applicable readiness reviews. Also if applicable, describe how reuse will be applied to the testing effort to make testing more efficient and less costly. Finally discuss the defect issue resolution process and how retesting of issues will occur.*

# PLANNED TESTS

*[As applicable to the scope of the Test Plan being prepared, describe the various types of testing (test functions) to be performed for the system, application or situation during the life cycle, taking into consideration the system development methodology that is being employed for the project (e.g., waterfall or iterative).*

*Modify the list below to indicate which tests will be performed during the course of the project.]*

During the Unit Tests, System Tests, Integration Tests and User Acceptance Tests, the following are the various types of testing that will be performed:

* Functionality Testing - ensures that the system operates as specified in the Functional Requirements Specification.
* Interface Testing - ensures that the interfaces to other systems or subsystems are working correctly
* Process Testing - exercises the processing logic of the system to ensure that edits, calculations, and database updates are performed correctly.
* Scalability/Volume Testing - ensures that the system operates smoothly when subjected to production volumes of data over long periods of time.
* Stress Testing - ensures that the system operates smoothly when subjected to the maximum load expected in production, all at one time. A good rule of thumb is to subject the system to 25 percent more data and processing than is expected during peak loads.
* Load/Storage Testing - to allocate production size data stores to verify job run procedures.
* Recovery Testing - ensures that backup and recovery procedures are working properly. Typically, a base test of the recovery procedures is completed at the beginning of System Test and further recovery testing is scheduled towards the end of the System Test.
* Security Testing - verifies that system security meets requirements. As security is fundamental to the system, security testing is usually scheduled near the beginning of the System Test.
* Conversion Testing- verifies that existing data is converted correctly. Normally the converted data is used, along with new data, during process testing to verify the data conversion.
* Human Interface/GUI Testing - verifies that the human interface is consistent and adheres to standards. The amount of testing required depends on the tools used during development to ensure consistency.
* Usability Testing - verifies that the human interface is compatible with the customer's requirements for being understandable to the user.
* Documentation Testing - validates the user aids prepared for the system.
* Forms Testing - verifies the interface between the system and the manual forms and supporting procedures. Any forms used to gather data for the system are used during part of the system test to uncover changes required to the forms or the application.
* Performance Testing - verifies that performance criteria are met, (e.g., response time)
* Compatibility testing - evaluates the application's compatibility with the computing environment which could include: capacity of hardware platform (IBM 360, HP 9000, etc.), bandwidth handling capacity of networking hardware, compatibility of peripherals (Printer, DVD drive, etc.), operating systems (MVS, UNIX, Windows, etc.), database (Oracle, Sybase, DB2, etc.), other system software (Web server, networking/ messaging tool, etc.), or browser compatibility (Firefox, Netscape, Internet Explorer, Safari, etc.)
* Error handling testing – validate the error handling capabilities of system software components
* Penetration testing – evaluate the security of the system by simulating a malicious attack and checking for vulnerabilities
* Exploratory testing - testing that emphasizes the personal freedom and responsibility of the individual tester to continually optimize the quality of his/her work by treating test-related learning, test design, test execution, and test result interpretation as mutually supportive activities that run in parallel throughout the project
* Regression testing - rerunning previously run tests and checking whether program behavior has changed and whether previously fixed faults have re-emerged.
* Installation testing – validates that the installation of the hardware or software components at operating according to expectations.
* Maintenance testing - identify equipment problems, diagnose equipment problems or to confirm that repair measures have been effective.

# TEST PROGRESSION

***[As applicable to this Test Plan, explain the planned sequence or progression of the prescribed tests.*** *Identify existing dependencies that affect the conduct and progression of test activities. Also identify any regularly held meetings or reports that provide information on or that may affect testing (e.g., Change Control Board (CCB) meetings, status reports, etc.).****]***

# FEATURE TESTING

## Features To Be Tested

*[List and describe system functions/features that are to be tested. The table below provides an example that identifies features, software items, criticality to deployment success, testing priority, and notes. An explanation should be provided for the methodology used to define the criticality to deployment success and testing priority levels established for the given project.]*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Feature | Software | Criticality to Deployment Success | Testing Priority | Notes |
| <Name of function/feature to be tested> | <Software item(s) with version number if appropriate> | <High, Medium, or Low> | <Testing Priority Level> | <Comments or additional notes> |
| Access Enrollment Application | Program A | High | 6 |  |
| Complete Enrollment Application | Program B | High | 5 |  |
| Submit Application | Program C | High | 5 |  |
| Receive Receipt & Acknowledgement | Program D | Medium | 4 |  |
| Queue Application | Program E | Low | 4 |  |
| Submit to Available Reviewer | Program E | High | 4 |  |

## Features Not To Be Tested

*[List and describe the system functions/features not planned to be tested and explain why. The table below provides an example that identifies features, software items, criticality to deployment success, testing priority, and notes. Generally, only identify non-critical functions/features with a designated low priority as features not to be tested due to resource or other identified constraints.]*

| Feature | Software | Notes |
| --- | --- | --- |
| <Name of function/feature not to be tested> | <Software item(s) with version number if appropriate> | <Comments or additional notes identifying reason(s) why feature will not be tested> |
| Validate Enrollment Eligibility | Program F |  |
| Validate Plan Selection | Program F |  |

# TEST CASES

*[Describe the measures taken to document and prioritize test cases, the controls applied to them, and how/where they are stored (e.g. eRoom, CD, library, etc.). Generally, test cases are traditionally documented in a separate Test Specification that should be referenced within this section.]*

# DEFECT TRACKING & REPORTING

*[Describe the defect resolution process to be implemented during testing, including the operational definition and assignment of appropriate impact/severity levels. For example, identify the reporting or defect tracking system that will be used to document unexpected results, problems, or defects that occur during the prescribed testing and their resolutions. If multiple phases are involved, there may be a defect resolution meeting to keep track of the inter-dependencies. If any third party is involved in the verification, then their roles should also be included here.]*

# ENVIRONMENT

***[Provide details and a graphical presentation of the environmental components required to test the system to include:*** *hardware, software, communications, and any other resources used to configure the test environment(s), as well as any security considerations. If multiple test sites will be used, each test site should be explicitly identified and the test environment for each test site appropriately described. If multiple test sites use the same or similar test environments, they may be discussed together with the differences clearly identified. The test environment(s) should reflect the planned production environment as closely as possible or with capabilities proportional to the level of testing. Also provide details where the test environment(s) does not mirror the production environment.]*

## Hardware

*[Identify by name, number, and version, as applicable, all computer hardware, interfacing equipment, communications equipment, peripherals, etc. that will be required at each test site. Describe the purpose of each item, and state the period of usage and the number of each item needed.]*

| Hardware Item | Purpose | Period of Usage | Number Needed | Issues | Type of Test |
| --- | --- | --- | --- | --- | --- |
| <Identifying information for hardware item (e.g., name, number, version)> | <Role of hardware item in the testing> | <Period of usage> | <Number of the item needed> | <Issues associated with item> | <Unit, System, Integration, UAT> |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Software

*[Identify by name, number, and version, as applicable, all software items (e.g., operating systems, compilers, communications software, related applications software, databases, input files, code auditors, dynamic path analyzers, test drivers, pre-processors, test data generators, test control software, other special test software, post-processors, etc.) that will be required at each test site. Describe the purpose of each item, its media, and state the period of usage and the number of each item needed. Also identify the proprietary nature and any licensing issues associated with each item.]*

| Software Item | Purpose | Media | Period of Usage | Number Needed | Issues | Type of Test |
| --- | --- | --- | --- | --- | --- | --- |
| <Identifying information for software item (e.g., name, number, version)> | <Role of software item in the testing> | <Media for providing item> | <Period of usage> | <Number of the item needed> | <Issues associated with item> | <Unit, System, Integration, UAT> |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

## Test Data

*[Provide a detailed description of the test data to be used for the various testing activities. If personally identifiable information (PII) will be used, identify how this test data will be protected and controlled.]*

## Other Materials

*[Identify and describe any other materials needed for the testing at the test site(s). These materials may include manuals or other forms of instruction. Identify the type and quantity of the materials, as applicable.]*

## Installation, Testing, & Control

*[Identify plans for establishing the test environment at each test site and testing each element prior to its use. Also describe how the test environment will be controlled and maintained. The schedule for establishing the test environment should be documented in the Project Schedule and referenced here as appropriate.]*

## Security

*[Identify any security or privacy issues associated with the test environment, including any issues regarding Personally Indentifying Information (PII) not previously addressed in the section above regarding Test Data.]*

# SCHEDULES & MILESTONES

*[List the milestone events and dates for the all testing activities, including each test site as appropriate. The example below provides insight into some of the types of testing activities that should be shown in the project’s testing schedule and within the overall project schedule:]*

| Date | Milestone |
| --- | --- |
| 01/31/2011 | Create Draft Test Plan |
| 02/15/2011 | Final Approval of Test Plan |
| 03/14/2011 | Complete Development Testing |
| 03/31/2011 | Conduct Review |
| 04/07/2011 | Begin Validation Testing |
| 04/30/2011 | Complete Validation Testing |
| 05/05/2011 | Conduct Validation Phase Gate Review/Exit |
| 05/12/2011  05/30/2011 | Begin Implementation Testing  Complete Implementation Testing |
| 06/13/2011 | Conduct Project Closure Meeting |

# ROLES & RESPONSIBILITIES

## Resources

*[Identify the number, type, and skill level of the personnel that will be needed from each organization to participate in the testing activities during each of the prescribed tests at the designated test site(s), and describe the roles and responsibilities of each. Include the names of the individuals, if known.]*

| Skill Type | Skill Level | # Personnel | Assigned Staff | Length of Time Needed | Role | Type of Test |
| --- | --- | --- | --- | --- | --- | --- |
| <skill type (e.g., test coordination)> | <skill level and/or specific expertise (e.g., senior-level)> | <number of personnel needed with associated skill type and skill level> | <name(s) of individuals / organization> | <# of hours, days, weeks, months, etc. or timeframe that the personnel will be needed> | <role in the testing> | <Unit, System, Integration, UAT> |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

## Orientation Plan

*[Describe any orientation and training to be given before and/or during the testing. This training may include user instruction, operator instruction, maintenance and control group instruction, and orientation briefings to test team personnel. If extensive training is anticipated, a separate Training Plan and Training Artifacts may be developed and referenced here.]*