NINDS Web Application Testing

Version 1.0

Master Test Plan

December 2019

NIH / NINDS

Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
|  |  |  |  |
| 12/13/2019 | V1 | Master Test Plan Document | Venkat Shanmugam |

Table of Contents

[1. Introduction 1](#_Toc433725062)

[1.1. Purpose 1](#_Toc433725063)

[1.2. Test Objectives 1](#_Toc433725064)

[1.3. Roles and Responsibilities 2](#_Toc433725065)

[1.4. Processes and References 2](#_Toc433725066)

[2. Items To Be Tested 3](#_Toc433725067)

[2.1. Overview of Test Inclusions 3](#_Toc433725068)

[2.2. Overview of Test Exclusions 3](#_Toc433725069)

[3. Test Approach 3](#_Toc433725070)

[3.1. Product Component Test 3](#_Toc433725071)

[3.2. Component Integration Test 3](#_Toc433725072)

[3.3. System Tests 4](#_Toc433725073)

[3.4. User Functionality Test 4](#_Toc433725074)

[3.5. Enterprise System Engineering Testing 4](#_Toc433725075)

[3.6. Initial Operating Capability Evaluation 4](#_Toc433725076)

[4. Testing Techniques 4](#_Toc433725077)

[4.1. Risk-based Testing 4](#_Toc433725078)

[4.2. Enterprise Testing 4](#_Toc433725079)

[4.2.1. Section 508 Compliance Testing 5](#_Toc433725082)

[4.3. Performance and Capacity Testing 6](#_Toc433725084)

[4.4. Test Types 6](#_Toc433725085)

[4.5. Productivity and Support Tools 8](#_Toc433725086)

[5. Test Criteria 8](#_Toc433725087)

[5.1. Process Reviews 8](#_Toc433725088)

[5.2. Pass/Fail Criteria 9](#_Toc433725089)

[5.3. Suspension and Resumption Criteria 9](#_Toc433725090)

[6. Test Deliverables 9](#_Toc433725091)

[7. Test Schedule 10](#_Toc433725092)

[8. Test Environments 10](#_Toc433725093)

[8.1. Test Environment Configurations 10](#_Toc433725094)

[8.2. Base System Hardware 11](#_Toc433725095)

[8.3. Base Software Elements in the Test Environments 12](#_Toc433725096)

[9. Staffing and Training Needs 12](#_Toc433725097)

[10. Risks and Constraints 13](#_Toc433725098)

[11. Test Metrics 13](#_Toc433725099)

[Attachment A – Approval Signatures 14](#_Toc433725100)

# Introduction

To start a software testing project. it’s good practice to go through a standard testing process to lay the foundation for success and make sure we don’t forget anything.

This standard testing process and planned checklist should consider doing from beginning to the end of the project

This document is a comprehensive checklist of executing a software testing project.

## Purpose

This is a Master Test Plan document aimed for NINDS FASTR Applications in mind. The goals of FASTR Applications is to merge the functionality and features of Brainex (Brain Grants) and Extramural Financial Applications (FINeX). Rebuild the whole new FASTR systems and eventually Brainex and FINeX would get retired down the line. FASTR will help users to consolidate the features in one application than working on two other systems.

Include that this test plan will:

* Document the overall testing process.
* Describe the Test Strategy including defining the test levels and types of tests planned.
* Include testing activities to be performed.
* Document who will perform the test activities.

For projects which include New Software, Software upgrades or Patches:

The patch identifier for this Test plan is <Patch Number e.g. FB\*3.5\*124>.

## Test Objectives

Tailor the test objectives as appropriate.

This Master Test Plan supports the following objectives:

* To provide test coverage for 100% of the documented requirements
* To provide coverage for System / Software Design Document elements
* To execute 100% of the test cases during User Functionality Testing
* To create, maintain and control the test environment
* Add other objectives as needed

## Roles and Responsibilities

Customize the table below according to the roles that support the execution of the Master Test Plan.

Table 1 lists the key roles and their responsibilities for this Master Test Plan.

Table 1: Roles and Descriptions

| Role | Description |
| --- | --- |
| Development Team | Persons that build or construct the product/product component. |
| Development Manager | Person responsible for assisting with the creation and implementation of the Master Test Plan. |
| Program Manager | Person that has overall responsibility for the successful planning and execution of a project; person responsible for creating the Master Test Plan in collaboration with the Development Manager. |
| Stakeholders | Persons that hold a stake in a situation in which they may affect or be affected by the outcome. |
| Test Lead | An experienced Test Analyst or member of the Test Team that leads and coordinates activities related to all aspects of testing based on an approved Master Test Plan and schedule. |
| Test Team/Testers | Persons that execute tests and ensure the test environment will adequately support planned test activities. |

## Processes and References

The processes that guide the implementation of this Master Test Plan are:

* Test Preparation
* Product or Application New Features Build on existing Product
* Independent Test and Evaluation

The references that support the implementation of this Master Test Plan are:

* Business Requirement Document (BRD) Version <#.#>, Date <Month, Year>
* Requirements Specification Document (RSD) Version <#.#>, Date <Month, Year>
* System Design Document (SDD) Version <#.#>, Date <Month, Year>
* Requirements Traceability Matrix (RTM) Version <#.#>, Date <Month, Year>
* Defect Log Tracking Version <#.#>, Date <Month, Year>

# Items to Be Tested

List those test items - software and supporting product elements that serve as targets for testing. A test item may include source code, control data, documentation, or a collection of these.

## Overview of Test Inclusions

Provide a high-level list of the major target test items. This list should include both items produced directly by the project Development Team and, if applicable, as well as vendor-supplied products being integrated into the information system or application. Refer to the Requirements Specification Document (RSD) to identify the requirements needed for testing items. Consider grouping the list by category and assigning relative importance to each motivator.

The following components and features and combinations of components and features will be tested:

List of Test cases related to user stories are listed here for testing.

## Overview of Test Exclusions

Identify any items specifically excluded from testing.

The following components and features and combinations of components and features will not be tested:

If the features of the applications are not qualified for testing. That would be listed here as follows.

# Test Approach

The Test Approach is the implementation of the Test Strategy.

The Software Test Plan

1. Confirm the workspace and objectives

Testing Scope: OS (Mobile, Client Server, Web application), browsers, tools and

functionality to be tested and type of the testing.

For Example.

Performance,

Usability,

Automation,

Functional,

Security,

System Test,

Installation / Upgrade / Patching,

Recovery,

Beta Test,

User Acceptance Testing.

2) Resources needed and available (labor, skills, tools) etc.:

Depending on the scope you will need to get the expertise to execute that type of testing, along with the right tools, you will need a budget for time and tools for themselves.

Schedule and milestones:

Discuss with the development team what their schedule is and how they will deliver the product. If doing the agile, they will have a small working prototype, and minimal documentation on requirements, so you will have to work closely with them to flush out what the requirements for finished are for each sprint.

Do not forget to establish a change control procedure.

1. Risk Planning:

Besides conforming to the schedule and figuring out meaning milestones, think of risk management and contingency – Thinks about what you can go wrong and plan for it.

For example:

1. Resources not enough or not the right skills or need more time to learn.
2. Tools do not work correctly.
3. Software not delivered until late.

Document Dependencies:

Personal Dependencies

Software Dependencies

Hardware Dependencies

Test Data & Database

Document Risks

Schedule

Technical

Management

Personnel

Requirements

Determine Suspension, Exit and Resumption Criteria based on Document Dependencies and Document Risks

Written tests artifacts and deliverables.?

(for example, What metrics and measurements? daily? weekly? doc report? online report?)

So that you can look for measurements and learn from mistakes, lesson learned is must.

Create Test Project Management

It’s important for collaboration not only among the test team but for both test and development team to share ideas and solutions.

Create Configuration Management

This is especially with test case management, automation and performance scripts.

Determine access rights, naming conventions and best practices.

Communicate to the team.

NINDS is a Microsoft favored technology shop, Azure Dev Ops would provide more value to our development cycle by understanding the core concepts of Azure Dev Ops

This helps in integration of TFS, PBI, CI/CD, Microsoft Test Manager with Test Plans, Test cases, Attachment of Test plans, Test cases to each PBIs and Development Sprint Iterations and many more to achieve fullest CI/CD.

Test cases design and review

Different test case design should be investigated and designed depending on scope, time and skill of the test team.

1. Test case template – test cases should be standardized in the format and minimal content, otherwise they become individualized thus increasing risky
2. Test case analysis – Test case need to be examined relative to the software coverage and risk.
3. Defect logging and review – when logging defects, its important to characterize the defect when developers are trying to fix them, they will know exactly in what instances it occurs
4. Defect template: Defects should be reported in a standard manner in a concise way yet containing all the information needed.
5. Defect Analysis: After going through the week or two, you will pick up the trends in the software quality and in the performance of your team.
6. Confirm defect/questions with development team: Do triage with the development team, working with them, to see how you can become more efficient and work better together. What is critical and what is not may depend on point of view, the current release, timing and moods of the individuals.

## Product Component Test

Briefly describe how the Developers perform Product Component Test, also known as Unit Test. Identify the responsible roles.

Component Integration Test

Briefly describe how the Developers perform Component Integration Test. Identify the responsible roles.

## System Tests

Briefly describe how the system or application will be tested during System Tests. At a high level specify any testing requirements, such as, test environment, hardware, test data, or dependencies.

For more information will go under the section after knowing the NINDS Grant Applications

## User Functionality Test

Briefly describe how the system or application will be tested during User Functionality Test. At a high level specify any testing requirements, such as, point of contact, test environment, test data, hardware, or dependencies

This section will cover the Functional testing part. Requires the understanding of the various features across the NINDS Grant applications.

## Enterprise System Engineering Testing

Specify how the Development Team will support Enterprise System Engineering (ESE) testing, the development team point of contact, and any special testing requirements and dependencies, including Performance Testing. Include intended testing process, plans for test scripts, and likely test scenarios. For more information on ESE testing

I do not think any special testing requirements are needed to cover this portion. This is completely kept open for NINDS Grants Applications.

## Initial Operating Capability Evaluation

Briefly describe how the Development Team will support the Test Sites during Initial Operating Capability Evaluation. Initial Operating Capability Evaluation was formerly known as Field Testing.

# Testing Techniques

Testing Techniques describes the approach to risk-based testing, requirements for enterprise testing, test types, iterations, and tools that are used to test the designated test items as applicable.

## Risk-based Testing

Describe the potential risks that may cause the system to not meet reasonable user and customer expectations of quality. Risk-based testing is a technique for prioritizing testing based on testing the highest risk items first and continuing to test down the risk prioritization ladder as the testing schedule permits. Describe how the identified risks have been covered in the testing effort. For example, a table may be created to identify which test type or which test cases will be executed to address the identified risks.

## Enterprise Testing

Cite how the project testing covers the enterprise requirements. Enterprise requirements include security, privacy, Section 508 Compliance requirements, and Multi-divisional requirements.

### WCAG Accessibility / Section 508 Compliance Testing

Section 508 Compliance Testing is required for all applications.

Development Team is responsible for ensuring that product functionality is accessible and works with adaptive technology. Section 508 Program Office provides consultation on how to implement and test Section 508 compliant solutions, tools to conduct the testing, and training on how to use the tools and other aspects of Section 508.

The project must submit proof of compliance to Section 508 Office personal.

## Performance and Capacity Testing

Develop tests to ensure the application will perform as expected under anticipated user loads, and typical business transactions respond in a timely manner. During the test execution, the System Under Test (SUT) is actively monitored for any issues that could affect application performance, and to verify the hardware environment is adequately sized.

This type of testing covers the requirements specified in the “Performance Specifications” in the Requirements Specification Document found in the Requirements process of NINDS Grant Application.

## Test Types

Test types are a group of test activities aimed at testing a component or system regarding one or more interrelated quality attributes. A test type is focused on a specific test objective, i.e., reliability test, usability test, regression test etc., and may take place on one or more test levels or test phases. Specify the Test Types to be performed and the party responsible for performing the test. Delete from the table any test type that does not apply.

Table 2: Test Types

| Test Types | Party Responsible |
| --- | --- |
| Access control testing |  |
| Build verification testing |  |
| Compliance testing |  |
| Component integration testing |  |
| Configuration testing |  |
| Data and database integrity testing |  |
| Documentation testing |  |
| Error analysis testing |  |
| Exploratory testing |  |
| Failover testing |  |
| Installation testing |  |
| Integration testing |  |
| Migration testing |  |
| Performance monitoring testing |  |
| Performance testing |  |
| Product component testing |  |
| Regression test |  |
| Section 508 compliance testing |  |
| Security testing |  |
| Smoke testing |  |
| System testing |  |
| Usability testing |  |
| User Functionality Testing |  |
| User interface testing |  |

## Productivity and Support Tools

Add or delete tools as appropriate.

Table 3 describes the tools that will be employed to support this Master Test Plan.

Table 3: Tool Category or Types

| Tool Category or Type | Tool Brand Name | Vendor or In-house | Version |
| --- | --- | --- | --- |
| Test Management |  |  |  |
| Defect Tracking |  |  |  |
| Test Coverage Monitor or Profiler |  |  |  |
| Project Management |  |  |  |
| Performance Testing |  |  |  |
| Configuration Management |  |  |  |
| DBMS tools |  |  |  |
| Functional Test Automation |  |  |  |
| Other |  |  |  |

# Test Criteria

## Process Reviews

The Master Test Plan undergoes two reviews:

* Peer Review – upon completion of the Master Test Plan
* Formal Review – after the Development Manager approves the Master Test Plan

For more information on the reviews associated with testing, see the Product Build, Test Preparation, and Independent Test and Evaluation processes.

## Pass/Fail Criteria

Pass/Fail criteria are decision rules used to determine whether a test item (function) or feature has passed or failed a test.

Specify the criteria to be used to determine whether the test items have passed or failed testing.

## Suspension and Resumption Criteria

Suspension Criteria are the criteria used to (temporarily) stop all or a portion of the testing activities on the test items. Resumption Criteria are the testing activities that must be repeated when testing is re-started after a suspension.

Specify the suspension and resumption criteria that will guide test execution.

# Test Deliverables

The Test Deliverables listed below represent some possible deliverables for a testing project. The Test Deliverables table may be tailored to meet project needs. Do not include Delete any listed test deliverable that is not used by the Product Build, Test Management, and Independent Test and Evaluation processes.

Table 4 lists the test deliverables for the NINDS Grant project.

Table 4: Test Deliverables

| Test Deliverables | Responsible Party |
| --- | --- |
| Master Test Plan | QA |
| Performance Test Plan | QA |
| Iteration Test Plans *(when appropriate)* | QA |
| Test Execution Risks | QA |
| Test Schedule | QA |
| Test Cases/Test Scripts | QA |
| Test Data | QA |
| Test Environment | QA |
| Test Evaluation (including performance test results) | QA |
| Traceability Report or Matrix | QA |

# Test Schedule

Integration testing – Individual software modules are combined and tester as a group.

System Testing – Conducted on a complete, integrated system to evaluate the system’s compliance with its specific requirements

Suspension Criteria and Resumption Requirements

If the team member reports that there are 40% test case are failed, suspend the testing event until all the test case are fixed.

Test Completeness.

List the major testing milestones. When appropriate, reference other workflow documentation or tools, such as the Project Management Plan, or Work Breakdown Structure (WBS.) Put a minimum amount of process and planning information within the Master Test Plan in order to facilitate ongoing maintenance of the test schedule.

Table 5: Testing Milestones

| Testing Milestones | Responsible Party |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Test Environments

A test environment is an environment containing hardware, software, tools and techniques, and other support elements needed to conduct a test.

## Test Environment Configurations

Successful testing requires control of the test environment. Unplanned changes to the test environment may introduce new defects, alter the expected test results, and thus invalidate the test cases. Successful testing requires controlled access to the test environment, an environment that replicates the environment as closely as possible.

In order to ensure the verification and validation of applications and systems requiring multi-divisional capabilities, be sure to configure the test environments as multi-divisional environments. For more information, see section 3.3.4 in this document.

The party or parties responsible for configuring and maintaining the test environments are: {Operations Team providing Infrastructure }.

## Base System Hardware

Table 6 sets forth the system resources for the test effort presented in this Master Test Plan.

The specific elements of the test system may not be fully understood in early iterations, so this section may be completed over time. The test system should simulate the production environment as closely as possible, scaling down the concurrent access and database size, and so forth, if and where appropriate. Tailor the System Hardware Resources table as required.

Table 6: System Hardware Resources

| Resource | Quantity | Name and Type |
| --- | --- | --- |
| Database Server |  |  |
| Network or Subnet |  | TBD |
| Server Name |  | Fastrtest.ninds.nih.gov |
| Client Test PCs |  |  |
| Include special configuration requirements |  | TBD |
| Test Repository |  |  |
| Network or Subnet |  | TBD |
| Server Name |  | TBD |
| Test Development PCs |  | TBD |

**Test Roles and Responsibilities**

This is a list of the testing roles and responsibilities.

|  |  |  |
| --- | --- | --- |
| ***Resource*** | ***Role*** | ***Responsibility*** |
| *NINDS Test Lead* | *Test Manager/Project Manager* | * *Provides technical direction.* * *Acquires testing resources.* * *Assigns tasks to testing resources.* * *Communicates test results to management.* * *Documents test plan.* * *Identifies, documents and prioritizes test cases.* * *Evaluates effectiveness of test effort.* * *Ensures test environment and assets are installed, managed and maintained.* * *Includes administration of testing ids* |
| *NINDS Tester* | *Tester* | * *Executes test cases, logs defects, and documents test results.* |

## Base Software Elements in the Test Environments

Add or delete Software Elements as appropriate. If necessary, specify software patches referenced and/or required here.

Table 7 describes the base software elements that are required in the test environment for this Master Test Plan.

Table 7: Software Elements

| Software Element Name | Version | Type and Other Notes |
| --- | --- | --- |
| Windows/Unix/ Server |  | Operating System |
| Internet Explorer, Chrome, Firefox |  | Top Browser |
| Oracle, SQL Server DB |  | RDBMS |
|  |  |  |

# Staffing and Training Needs

Table 8 describes the personnel resources needed to plan, prepare, and execute this Master Test Plan.

Table 8: Staffing Resources

| Testing Task | Quantity of Personnel Needed | Test Process | Duration/ Days |
| --- | --- | --- | --- |
| Create the Master Test Plan |  | Test Preparation | 3 Weeks |
| Establish the Test Environment |  | Test Preparation | 3 Weeks |
| Perform System Tests |  | Product Build | 1 weeks |
| Etc. |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Identify training options for providing necessary skills and the estimated number of hours necessary to complete the training.

Table 9 lists the personnel that require training.

Table 9: Training Needs

| **Name** | **Training Need** | **Training Option** | **Estimated Training Hours** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Risks and Constraints

The Test Preparation process requires the performance of a risk assessment for test execution. Risks associated with the testing project are potential problems/events that may cause damage to the software, systems, patient, personnel, operating systems, schedule, scope, budget or resources. The risks, listed in the risk log, may impact scope and schedule, necessitating a deviation from this Master Test Plan.

The risk log was taken into consideration in the development of this test plan.

The risks identified in this Master Test Plan can be found in the risk log and may be recorded and tracked in an automated tool, such as, IBM Rational ClearQuest®.

# Test Metrics

Metrics are a system of parameters or methods for quantitative and periodic assessment of a process that is to be measured.

Test metrics may include, but are not limited to:

* Number of test cases (pass/fail)
* Percentage of test cases executed
* Number of requirements and percentage tested
* Percentage of test cases resulting in defect detection
* Number of defects attributed to test case/test script creation
* Percentage of defects identified; listed by cause and severity
* Time to re-test

Attachment A – Approval Signatures

The Master Test Plan documents the project’s overall approach to testing and includes:

* Items to be tested
* Test strategy
* Test criteria
* Test deliverables
* Test schedule
* Test environments
* Staffing and training needs
* Risks and constraints
* Test Metrics

This section is used to document the approval of the Master Test Plan during the Formal Review. The review should be ideally conducted face to face where signatures can be obtained ‘live’ during the review however the following forms of approval are acceptable:

1. Physical signatures obtained face to face or via fax
2. Digital signatures tied cryptographically to the signer
3. /es/ in the signature block provided that a separate digitally signed e-mail indicating the signer’s approval is provided and kept with the document.

NOTE: Delete the entire section above prior to final submission.

REVIEW DATE: <Date>

Signed: Date:

< Program/Project Manager >

Signed: Date:

<Project Team Test Manager>

Template Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
|  |  |  |  |
| December 2019 | 1.0 | Master Test Plan Template | QA |