**Research Administrative Management System (RAMS)**

R1605

Master Test Plan



**October 2016**

Department of Veterans Affairs

Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
| 10/03/2016 | 1.0 | R1605 Master Test Plan Baseline | Booz Allen Hamilton |

Table of Contents

[1. Introduction 5](#_Toc463626293)

[1.1. Purpose 5](#_Toc463626294)

[1.2. Test Objectives 6](#_Toc463626307)

[1.3. Roles and Responsibilities 6](#_Toc463626308)

[1.4. Processes and References 7](#_Toc463626309)

[2. Items to Be Tested 8](#_Toc463626310)

[2.1. Overview of Test Inclusions 10](#_Toc463626311)

[2.2. Overview of Test Exclusions 11](#_Toc463626312)

[3. Test Approach 11](#_Toc463626313)

[3.1. Product Component Test 1](#_Toc463626316)

[3.2. Component Integration Test 2](#_Toc463626317)

[3.3. System Tests 2](#_Toc463626318)

[3.4. User Functionality Test 2](#_Toc463626319)

[3.5. Enterprise System Engineering Testing 3](#_Toc463626320)

[3.6. Initial Operating Capability Evaluation 3](#_Toc463626321)

[4. Testing Techniques 4](#_Toc463626322)

[4.1. Risk-based Testing 4](#_Toc463626323)

[4.2. Enterprise Testing 4](#_Toc463626324)

[4.2.1. Security Testing 4](#_Toc463626325)

[4.2.2. Privacy Testing 4](#_Toc463626326)

[4.2.3. Section 508 Compliance Testing 4](#_Toc463626327)

[4.2.4. Multi-Divisional Testing 4](#_Toc463626328)

[4.3. Performance and Capacity Testing 5](#_Toc463626329)

[4.4. Test Types 5](#_Toc463626330)

[4.5. Productivity and Support Tools 6](#_Toc463626331)

[5. Test Criteria 7](#_Toc463626332)

[5.1. Process Reviews 7](#_Toc463626333)

[5.2. Pass/Fail Criteria 8](#_Toc463626334)

[5.3. Suspension and Resumption Criteria 8](#_Toc463626335)

[6. Test Deliverables 9](#_Toc463626336)

[7. Test Schedule 10](#_Toc463626337)

[8. Test Environments 10](#_Toc463626338)

[8.1. Test Environment Configurations 10](#_Toc463626339)

[8.2. Base System Hardware 11](#_Toc463626340)

[8.3. Base Software Elements in the Test Environments 12](#_Toc463626341)

[9. Staffing and Training Needs 12](#_Toc463626342)

[10. Risks and Constraints 13](#_Toc463626343)

[11. Test Metrics 13](#_Toc463626344)

[Attachment A – Approval Signatures 14](#_Toc463626345)

[Appendix A - Test Type Definitions 15](#_Toc463626346)

# Introduction

This Master Test Plan defines the test strategy and outlines the approach that is taken to perform tests on the Department of Veterans Affairs (VA) Research Administrative Management System (RAMS). This document adheres to the ProPath template and includes the test objectives, roles and responsibilities, test approach, testing techniques, criteria, deliverables, schedule, environment, training, risks, and test metrics. The business rules stated in the Requirements Specification Document (RSD) for the current production release of RAMS will be referenced throughout the testing process. The scope of requirements approved for production use may increase over time, as additional development initiatives are completed, and will subsequently be reflected in MTP for each revision of the RAMS system.

## Purpose

The Veterans Health Administration (VHA) Office of Research and Development (ORD) is responsible for the management and administration of the national VA research program. ORD aims to improve the performance of the national VA research program by implementing a research management system that provides a common database for tracking and reporting administrative research program data throughout the VA.

RAMS is a single web-based front-end data management, reporting application framework, and centralized back-end database system, based on a collection of technologies that include Java and SharePoint 2013. The purpose of RAMS testing is to ensure consistency and functional quality of the software that is deployed in the production environment. The testing will cover major business functions for approval from both single site and multi-site human research projects, and will be extended over time to include the functionality released through subsequent new development initiatives. Testing will also provide the necessary traceability to the design and requirements artifacts to ensure that developed components are properly maintained and are fully operational. Table 1 depicts the five major functional processes that were included in Increment 1 and Increment 2.

Table 1: Six Major Functional Processes of RAMS

| Functional Area | Functional Description |
| --- | --- |
| Personnel Management | The RAMS Personnel process manages data about personnel engaged in VA research. |
| Project Management | The RAMS Project processes allow authorized users, based on their role, to enter, update, track and manage research projects/protocols and upload associated documents. |
| Committee Management | The RAMS Committee processes automate the review, communications and request for amendments processes that are initiated by the IRB committee. |
| Document Management | The RAMS Document process includes uploading, reviewing and submitting documents as part of the IRB application. |
| Communications Management | The RAMS Communications processes include creating task lists, generating notifications and providing report generation tools. |
| RAMS Data Bridge | RAMS Data Bridge, which allows RAMS users to import archived study data from a spreadsheet into the RAMS Application |

## Test Objectives

The objective of the Master Test Plan is to identify the approach, resources, and schedule for RAMS testing activities. This includes high level items to be tested, features to be tested, testing tasks to be performed, personnel responsible for each task, and risks associated with this plan. Testing, at a high level, will ensure the following:

* An application that is centrally managed and accessible to all authorized staff for creating, editing, modifying, searching, and reporting on VA research studies.
* Support for key business process workflows that are defined during the requirements validation process.
* Standards for capturing structured data, which will support improvements in central reporting and management of performance metrics.
* Capturing of all required data that will enable the Institutional Review Board (IRB) to make informed decisions for approval of VA research projects/protocols.
* Required security rules are integrated to ensure an authority to operate can be achieved.

## Roles and Responsibilities

Table 2 lists the key roles and responsibilities associated with RAMS test activities.

Table 2: Roles and Responsibilities

| Role/Descriptions | Responsibilities |
| --- | --- |
| **Project and Technical Management Team**  Technical support for project management, testing, quality assurance and deployment of the RAMS product (includes Project Manager, Development Team and Test Team) | * Develop front-end web-based data management and reporting application and centralized back-end database system * Verify implementation and functionality of the system * Create the Master Test Plan * Create Test Cases/Test Scripts * Support the VHA sites throughout the testing process * Ensure full execution of the test process to include the verification of technical requirements and the validation of business requirements * Establish, maintain, and control test environments * Provide support for any additional repositories and tools for managing the configuration control and testing of the system * Create Test Evaluation Report |
| **VHA Users and Stakeholders**  Program oversight and approval of all project artifacts; performs UAT | * Provide input and agreement on the goals for the testing by research staff * Review test results and provide feedback * Review the delivered project test artifacts prior to acceptance and provide feedback * Perform validation of the system to ensure requirements were successfully implemented * Approve User Acceptance Test (UAT) Test Scripts and prioritize any defects discovered for correction within the agreed upon timeline |

## Processes and References

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

* Test Preparation
* Account Set Up and Training
* Test and Certification

The initial process for preparing the test environment requires all VHA testing sites to identify their own respective testers and to request test accounts, ensuring that there are testers available for each role in the system. All test cases shall be derived using the current approved Requirements Specification Document (RSD) and associated User Stories for the production instance of RAMS.

Instructions for establishing test accounts will be provided to every site, as well as initial training that is needed to complete User Acceptance Test (UAT). To ensure standardization, training materials will be developed for Local System Administrators. Throughout RAMS sustainment, virtual (online) training sessions will be available for end users who are designated as RAMS Local System Administrators.

Timeline will be provided for each tester to complete a series of test scripts, after which, each tester will record their results in a standard template. Any issues or defects found during UAT will also be recorded in a Test Defect Log. Items identified as change requests will be presented to the Change Control Board (CCB) for adjudication and prioritization. Any changes identified as new feature requests will be collected and will be provided periodically to the RAMS stakeholders for review and consideration as future development outside of the sustainment scope. The test progress and results reporting will take place during the daily test status conference call. Test result compilation will be managed by the Project and Technical Management Team, and will be presented back to the VA Project Manager using the standard ProPath templates.

VA will be responsible for Testing and Certification processes, which will adhere to standard VA processes. Specific reference documents are included in Table 3.

Table 3: Referenced Documents

| **Document Name** | **Version Number** | **Date** | **Description** |
| --- | --- | --- | --- |
| Contractor Project Management Plan (CPMP) | 5.0 | December 2015 | Describes the processes and procedures that will be used to manage all aspects of the RAMS Sustainment efforts |
| Research Administrative Management System Work Effort, Unique Identifying #01-02-03-08-05-004 | 2.1 | April 2014 | Business Requirements Document that describes the business needs of the customer |
| T4 Performance Work Statement (PWS), TAC Number: TAC-14-12171 | 13.4 | March 2014 | The performance work statement that describes the task order (TO) effort for RAMS Increment 1 and RAMS Increment 2 |
| T4 Performance Work Statement (PWS) for RAMS Sustainment, TAC Number: TAC-15-19116 | 4.0 | May 2015 | The performance work statement that describes the task order (TO) effort for RAMS Sustainment |
| Requirements Specification Document (RSD) Research Administrative Management System (RAMS) | 4.3 | November 2015 | Documents the requirements of the RAMS solution proposed through an acquisition contract |
| Software/System Design Document (SDD) | 1.8 | April 2015 | Signed SDD |
| Test Defect Log and Test Execution Log | N/A | Latest Version from ProPath | Document for recording, analyzing, tracking and closing defects. |
| Section 508 Office Web Page | N/A | N/A | <http://www.section508.gov> |
| Test Case/Test Script Checklist | N/A | Latest Version from ProPath | Provides a checklist for all possible test cases identified during test. |
| RAMS Increment 1 Test Evaluation Report | 1.0 | December 2014 | Document is the primary output of the test and evaluation process which identifies levels of performance for Increment 1. |
| RAMS Increment 2 Test Evaluation Report | 1.1 | May 2015 | Document is the primary output of the test and evaluation process which identifies levels of performance for Increment 2. |

# Items to Be Tested

The focus of the RAMS Sustainment task order is to conduct functional tests of remediated defects to ensure that requirements are satisfied, and to conduct regression tests to ensure that system changes do not inadvertently introduce new defects. Test execution will be carried out in multiple iterations for RAMS Sustainment Release R1605. The work items in the following Table 2‑1 will be tested for this release:

Table 4: Work Items in Scope

| **Work Item Id** | **Work Item Type** | **Summary** |
| --- | --- | --- |
| 365579 | User Story | Insert some UI indicator that the page is loading/refreshing |
| 374242 | User Story | As a RAMS user I want to be able to upload required documentation to the IRB Application, so that my submission may be reviewed and approved. |
| **Work Item Id** | **Work Item Type** | **Summary** |
| 351844 | User Story | Prevent the Save Progress button from being clicked twice without making a change on the form. |
| 226942 | User Story | True electronic submission system (can take all new protocols, not just IRB): Updates to IRB Application Questions: Section D – Question 30. |
| 230493 | User Story | Updates to IRB Application Questions: I6158919FY16: Section A number 4. Asking where co pi is located |
| 365546 | User Story | Enable the IRB Application to allow upload of attachments in a post-submitted state. |
| 365548 | User Story | Enable the Assigned IRB Admin for a study to view all user entered comments in one selection. |
| 365559 | User Story | Enable a [Print] button feature within the toolbar functionality |
| 368587 | Defect | Clicking in the 'blank space' of the IRB Application form automatically selects answers to radio buttons and checkboxes. |
| 368607 | Defect | The Disclaimer messages for Sections B, C, F, G, H, and L are difficult to read with the color contrast |
| 370161 | Defect | The left side of the comments window for the IRB Admin, Reviewer, and PI toolbar is longer (not even) with the right side. |
| 368609 | Defect | Remove the light grey vertical bar next to each question in a post submitted application |
| 226891 | Defect | The Expedited and Exempt category questions are missing from the RAMS toolbar during the review process |
| 264393 | Defect | The IRB Admin/Reviewer is unable to navigate to the breadcrumbs links if there is a toolbar, rather they have to expand the toolbar and click [Cancel] button. |

In addition to the defect remediation testing for the work items listed in Table 2‑1, the following major functionalities will be tested as part of the regression testing for this release:

* + Add/edit a Studies
  + Upload documents related to the studies
  + Create Consent and Health Insurance Portability and Accountability Act (HIPAA) letters from templates
  + Manage Study data provided by affiliate IRB
* Associate personnel to Study
  + - Add project staff and assign role and permission
    - Add/edit user profile

1. User Name, Contact, and Affiliations

2. Scope of Practice

3. User Education, Licenses and Credentials

4. Add Bio sketch or CV

5. User Research Appointment

6. User Employment data

7. Project Investigator Profile

* Preliminary Study Review
* Online communication between IRB Administrator and PI (Principal Investigator)
* Add project to selected agenda
* Assign type of review
* Manage Committee Processes
* Create and publish agenda
* Role based notification system
* Assign and receive reviews online
* Upload and manage documents with version control
* Create and publish minutes
* Track unexpected events and deviations
* Generate revision and rebuttal letters
* Generate renewal notifications
* Monitor calendar and continuing review deadlines for each project
* Manage processes unique to Central IRB (IRB)

1. Local site applications

2. Expanded notification structure

* Manage Study Status
* Track project status
* Develop Study status reports
* Update the Study status dashboard
* Generate data submission in MS Word format
* Export data for local reports

The following is a list of additional focus areas to consider during application testing:

* Graphical User Interface (GUI) functionality.
* Software interface implementation using SOAP/HTTP(S), XML/HTTP(S), SOAP/JAVA Message Service (JMS) or XML/JMS. Developed so that it is independent of the hardware operation system (e.g., Windows, LINUX, or UNIX).
* Hardware interface requirements such that RAMS shall be capable of running on virtual server(s).
* Privacy and Security specifications including VA’s Active Directory (LDAP) for user authentication, encryption, providing single sign-on (SSO) and complying with all applicable VA security standards and best practices.
* Section 508 accessibility compliance.
* Security Standards Compliance (Fortify, WASA, NESSUS, etc.)

## Overview of Test Inclusions

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

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

High-level functional testing includes:

RAMS software promotes interoperability by allowing access with other systems without requiring use of a specific language (e.g. Java only) to access the service, enabling to add new users and edit/update existing user profiles.

High-level security testing:

* Vulnerability testing.
* Security testing as part of the process for achieving and sustaining an authority to operate (ATO).

Release and operational readiness testing:

* Tests for completeness of all user and system documentation.

## Overview of Test Exclusions

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

* + Multi-divisional specifications **will not be** tested. From the perspective of the end user, there is no functional difference between user location and the functionalities provided by the RAMS.

# Test Approach

During sustainment, the RAMS Project and Technical Management Team will take a regression testing approach and focus on defect remediation. The following outlines the three testing phases for unit, integration, and system testing, which will focus on the five major functional areas of Personnel, Project, Committee, Document and Communications Management. Utilizing RAMS functionality that was developed and tested in Increment 1 and Increment 2, the Project and Technical Management Team will maintain the quality and the operational status of approved RAMS functionality in the production environment.

**Table 5: Three Phases of the Testing Approach**

| **Phase** | **Description** |
| --- | --- |
| Phase 1: Test Planning | * Participate in discussions to ensure prioritization, assignment to sprint/release, and split into testable features * Review the Master Test Plan and make updates, as necessary * Identify items to be tested, test strategy, test criteria, test deliverables, test schedule, test environments, staffing and training needs, risks and constraints and test metrics * Provide Test Cases, including test scripts and test data to drive testing * Submit Customer Acceptance Form for RAMS Test Cases to obtain VA sign-off prior to test execution |
| Phase 2: Test Execution | * Conduct release-level testing that includes integration and regression testing for each increment using automated tools wherever possible * Confirm integration with other applications such as Computer Associates SiteMinder (SSOi) and Microsoft Active Directory. * Assess current state of the test environment * Establish/configure the test environment * Execute test cases * Capture test results |
| Phase 3: Test Reporting | * Provide summary of testing activities, in the bi-weekly product development status report * Upon completion of acceptance testing, produce RAMS Test Evaluation report based on activities performed during the testing cycle for each increment and summarize the test activities with the results achieved * Provide confirmation that tests have been successfully executed * Cross-check results with requirements to ensure testing coverage of all requirements * Provide end-to-end requirements traceability, mapping each business requirement to test cases/scenarios |

## Product Component Test

Before code development, the development team will analyze the existing requirements for any failed components of RAMS and then develop a test which uses automated testing tools. This test will validate that the component is in a failed state, and will also form the basis to ensure that the corrective maintenance successfully corrects the defect. Afterwards, code development will produce a passing test of the previously failed component, along with the revised code. Automated test cases will be promoted to the test environment for additional testing. This approach is known as Test Driven Development (TDD). An embedded Quality Assurance (QA) tester will also perform unit testing to provide independent verification of software functionality in the Software Quality Assurance (SQA) environment. Unit testing includes internal technical and functional testing of the RAMS software code, which includes successful application of requirements and design specifications to the code in production. Steps include analyzing requirements to understand the application functionality and dependencies, executing tests with different combinations of data, and recording the actual test results. Unit test cases shall be designed to test the proper operation of a single program or component.

## Component Integration Test

The Project Team will perform component integration testing of the RAMS software using a combination of standalone and automated tools. The test team will utilize the test scripts developed by the developers during product component testing. The development team will also use a separate environment during the product component integration testing to confirm integration with other applications, such as Microsoft Active Directory. This test will aim to expose defects in the interfaces and interaction between integrated components, as well as, verifying installation instructions.

## System Tests

The Project and Technical Management Team will perform System Tests on the RAMS software interface. The goal of system testing is to detect faults that can only be exposed by testing the entire integrated system or major part of it. System testing is primarily concerned with areas such as functionality, performance, security, validation, load/stress, and configuration sensitivity. The focus for the RAMS Sustainment Team will be on performance and system maintenance and defect remediation.

## User Functionality Test

User Functionality Testing/User Acceptance Testing (UAT), also known as End-User Testing, will be performed by designated sponsors to verify that RAMS functionality satisfies user requirements. UAT participants will include representatives from the VHA ORD Leadership team, as well as subject matter experts and workgroup members who can provide insight to business requirements. UAT Test Team membership should represent the key users of the RAMS application. The UAT Test Team will work closely with the project and technical teams to ensure that resource availability for testing is consistent with the integrated project schedule. The UAT Test Team, as prescribed by the Agile Methodology, will be involved throughout all corrective maintenance release periods. This approach provides invaluable assistance in validating the user test scenarios, test scripts and user documentation.

An end-user training manual, user guide, and system administrator training manual will be provided in print and in online formats, to assist users involved with testing activities. The updated RAMS User Guide will provide description and instruction for all user accessible functionality by role. The RAMS System Administrators Guide will provide instruction on system installation, configuration, maintenance and security.

## Enterprise System Engineering Testing

The Project Team will work with the RAMS Project Manager to support Service Delivery and Engineering (SDE) to execute all required performance testing. The software will adhere to all VA Security and Privacy requirements. Based on Federal Information Processing Standard (FIPS) 199 and National Institute of Standards and Technology (NIST) SP 800-60, the recommended Security Categorization is Moderate. The Security Engineer (SE) assigned to the project will assist the stakeholders to determine the Security Categorization. This will drive the initial set of minimal security controls that is required for the information system. Minimum-security control requirements are addressed in NIST SP 800-53 and VA Handbook 6500. Efforts that involve collection and maintenance of individually identifiable information must be covered by the Privacy Act system of records notice. The software will also adhere to all Section 508 requirements, executive order requirements, and Enterprise Identity Management requirements. These requirements apply to any application that adds, updates, or performs searches on any personnel.

## Initial Operating Capability Evaluation

If required, the RAMS Project Manager will coordinate with the Project Team to conduct the Initial Operating Capability (IOC) evaluation process (This type of testing may be combined with User Acceptance Test). The software will be deployed to a single test server and will be accessed from three to five different test sites that will be involved in validating the software. Sites that are currently identified as test sites include:

* Miami
* New Orleans
* St. Louis
* Washington, D.C.

The Project and Technical Management Team shall ensure that the facilities involved in UAT understand the test processes and what is expected at the duration and at the end of the test. Test teams at these locations will run the test cases and ascertain whether the features and functionality perform as expected, and do not adversely affect the existing functionality of the product and the system.

# Testing Techniques

## Risk-based Testing

There are potential risks that may cause the system to produce dissatisfactory quality in user expectations. Therefore, a risk-based testing technique must be utilized to prioritize testing by starting with the highest risk items first and continuing to test down the risk prioritization ladder (as the schedule permits). The system must serve users who are located at VA Centers, as well as, those who access it remotely via the VA CAG from affiliates and universities. However, providing real time secure access could be a major constraint, and should be considered in determining the risk factors for testing the RAMS system.

## Enterprise Testing

### Security Testing

The Project and Technical Management Team will conduct a comprehensive assessment of the management, operational, and technical security controls for the RAMS interface. Based on the assessment, the team will determine whether the documented controls are operating as intended, and are producing the desired outcome with respect to meeting the security requirements based on the specified level of sensitivity. The system shall provide a process for end user account management (i.e., create account, inactivate account) and shall reside within VA firewalls. The team will review all audit logs, locally stored and captured data, as well as, data uploaded via the interface. Testing will include verifying user access controls and relevant security tests.

### Privacy Testing

Based on results of a Privacy Impact Assessment (PIA), any required privacy tests for RAMS are to ensure that (1) user and research data are adequately protected; and (2) the system complies with U.S. Government requirements for storing and handling Personally Identifiable Information (PII) at both the application and the database level. RAMS is designed to encrypt PII stored data. This test type validates the requirements specified in “Privacy and Security Specifications” in the Requirements Specification Document (RSD).

### Section 508 Compliance Testing

Accessibility specifications for RAMS include compliance with Section 508 requirements covered in http://www.section508.gov. The Development Team is responsible for ensuring that product functionality is usable from the keyboard. This test type validates the requirements specified in “Accessibility Specifications” in the Requirements Specification Document. Section 508 testing will be performed in coordination with the VA Section 508 Office.

### Multi-Divisional Testing

N/A – Please refer to Section 2.2 Overview of Test Exclusions.

## Performance and Capacity Testing

N/A

## Test Types

The relevant test types for this project are included in Table 5 below.

**Table 6: Relevant Test Types and Responsible Party**

| **Test Types** | **Party Responsible** |
| --- | --- |
| Functional Test | Project and Technical Management Team |
| Database Testing | Project and Technical Management Team |
| Integration Test | Technical Management Team |
| Regression Test | Technical Management Team |
| Section 508 Compliance Test | Technical Management Team |
| Security Test | Research Personnel/Security Office |
| Unit Test | Technical Management Team |
| User Acceptance Test | VHA Users and Stakeholders |

## Productivity and Support Tools

Table 7 identifies the test tools that may be utilized by the Test Team in evaluation and validation of RAMS. Table 7 also lists other tools that may be used to support this Master Test Plan. IBM Rational products are the standard for VA requirements management, configuration management, and test management.

Table 7: Tool Category or Types

| **Tool Brand Name** | **Description** | **Vendor or In-house** | **Version** |
| --- | --- | --- | --- |
| Rational Quality Manager (RQM) | Quality management tool that coordinates and manages quality assurance plans, processes, and resources across the project lifecycle. Lifecycle test plans define the roles, processes and deliverable ownership, and automate the flow of work and assets. RQM also provides test lab management capabilities to help users track, schedule, and execute tests on physical and virtual lab assets while acting as a central repository for test assets and templates for re-use. | IBM | 6.0.1 |
| Rational RequisitePro and/or Requirements Composer | Requirements management tool that allows project teams to manage their requirements, cases, and traceability. It integrates with MS Word and uses document-based methods while taking advantage of database-enabled capabilities such as requirements traceability and impact analysis. | IBM | 6.0.1 |
| ClearQuest (IBM Rational)  Test Execution Log (TEL) | A tool which will provide a repository for Defect Tracking, where issues may be entered and categorized according to functional area of impact, severity, and business priority | IBM  In-House (ProPath) | 7.1.2.3 |
| Requirements Traceability Matrix (RTM) | A tracking document which will ensure accountability between approved system requirements and test cases. Provides a monitoring tool to evaluate the completeness of test activities against expected functionality. This will be created by exporting data from Rational Quality Manager, referenced above | IBM | 6.0.1 |
| Primavera | PD standard tool for development of Project Management Plans. Also used in some cases for tracking actual time spent working on project related activities. | Oracle | P6 |
| Change & Configuration Management (CCM) | Configuration Management | IBM | 6.0.1 |
| Microsoft SQL Server | DBMS Tools for connecting to the RAMS database and validating data structures before and after test execution | Microsoft | 2012 |
| Window-Eyes  FireEyes | Section 508 Testing | GW Micro  Deque | 8.4  2.1.6 |
| Rational Functional Tester | Functional Test Automation Tools | IBM | 8.6.0.3 |

# Test Criteria

## Process Reviews

The Master Test Plan under goes 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

The following categories will be used to assess the potential impact of test incidents. A test incident is a deviation found during test execution, or a test execution that fails to meet expected criteria. The impact is the classification of the test incidents according to their potential damage to the software, system, personnel, or operating systems.

Pass Criteria:

A test is considered to have passed when both of these conditions are met:

* The functionality meets the requirements as defined
* Any test incident that fails to meet the requirement during the course of testing, is addressed by successful correction and testing

Fail Criteria:

A test is considered to have failed when either of these conditions is met:

* The functionality does not meet the requirement as defined
* A test incident that is found during the course of testing is not fixed and is not successfully tested

## Suspension and Resumption Criteria

Every effort will be made to ensure continued testing when failures are found. However, testing will be suspended if:

* Test accounts or variable items are unavailable or become unstable
* A failure that prevents the test from continuing or invalidates any additional testing to be performed, or corrupts the database
* A defect is discovered that corrupts the data within the database in such a way that proceeding would cause severe damage to the test environment
* A failure of major component. A major component failure is one that can reasonably be assumed will result in other test case failures

Testing will resume when the component or code is repaired, rebuilt and versioned, unit tested, and pass thorough system and regression testing. If the cause of the suspension is due to either an unstable or unavailable test environment, testing will resume when that test environment becomes stable and/or available.

RAMS is considered to have satisfied the acceptance criteria when:

* All high and medium priority test cases and scripts have been executed and the results recorded
* All high impact (Severity 1 or Severity 2) test incidents/defects have been resolved
* Any low impact (Severity 3 or Severity 4) test incidents/defects have been resolved or have been accepted by the business owner for future resolution.

# Test Deliverables

Table 8 lists the test deliverables for the RAMS Sustainment project.

Table 8: Test Deliverables

| **Test Deliverables** | **Responsible Party** |
| --- | --- |
| Test Cases/Scripts | Due 3 days prior to UAT |
| Software Quality Assurance Review Checklist | Due 2 days before deployment into SQA environment |
| Test Evaluation Report | Due 1 week after completion of test |
| UAT/Formal Test Results Report | Due 1 week after completion of test |
| Test Certifications | Due 1 week after completion of test |
| A&A Documentation | Due 3 days after implementation |
| Updated RAMS Source Code Set | Due 3 days after test and/or defect remediation |
| Updated RAMS ProPath Documentation Set | Due 3 days after implementation |
| Updated RAMS Product Release Package | Due 5 days after test and/or defect corrections |
| Defect Resolution Plan | Due 2 days after detection of defect |
| Weekly Defect/Fix Status Report | Due every Monday by noon, and weekly thereafter |
| RAMS Implementation and Operations Plan | Due 1 week before deployment into production |
| Repair Plan | Due 2 business days after trouble ticket log entry |

# Test Schedule

The milestones listed in Table 9 will serve as a guideline for establishment of a RAMS Sustainment Release R1605 test schedule. (See integrated master schedule for a detailed timeline.)

Table 9: Testing Milestones

|  |  |
| --- | --- |
| **Testing Milestones** | **Timeline** |
| Detailed Analysis Report | Due 3 days prior to normal/emergency corrective maintenance releases |
| Requirements Traceability Matrix | Due 3 days after normal/emergency corrective maintenance releases |
| Updated RAMS RSD | Due 5 days after normal/emergency corrective maintenance releases |
| Updated SDD | Due 5 days after normal/emergency corrective maintenance releases |
| Updated Master Test Plan | Due August 4, 2016, and updated with each functional module release |
| Functional Testing | Starts on August 5, 2016 |
| User Acceptance Testing | UAT Session starting from  August 10, 2016 |

# Test Environments

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

## Test Environment Configurations

The Project Team will work with all test sites to configure and maintain the test environments.

1. Product Component/Unit Tests will be performed by the Development team and embedded QA Tester in the development environment: “DEV” Enterprise Development Environment (EDE).

2. Integration/System Tests will be performed by the Project Team in the integrated test environment: “SQA”.

3. Section 508 Audit and Security Testing will be performed by the Section 508 Office and the Network Security Operations Center (NSOC) using the Philadelphia Information Technology Center (PITC) “SQA” environment.

4. UAT/IOC tests will be performed from pre-identified VHA field locations using the Philadelphia Information Technology Center (PITC) “SQA” environment.

## Base System Hardware

The test system will simulate the production environment as closely as possible. The project hardware requirements are identified in Table 10 below.

Table 10: System Hardware Resources

| **Resource** | **Quantity** | **Name and Type** |
| --- | --- | --- |
| * Client Test PCs | 8-10 | VA-issued laptop |
| RAMS Database | 1 | Microsoft SQL Server |
| Application | 1 | Microsoft Active Directory |
| Network | 1 | VA Intranet |
| Test Development PCs | 8-10 | Contractor issued laptops with Citrix Access Gateway (CAG) |
| Test Repository | 1 | IBM Rational Suite |
| Special Configuration Requirements | N/A | N/A |

## Base Software Elements in the Test Environments

Table 11 identifies base software components that are required in the test environment.

Table 11: Software Elements

| **Software Element Name** | **Type and Other Notes** | | **Version** |
| --- | --- | --- | --- |
| Document Repository | | Microsoft SharePoint 2013 | 2013 |
| RAMS IRB Application Logic | | Apache Tomcat 7 | 7.0.57 |
| RAMS Software | | Microsoft SQL Server 2012 | 2012 |
| RAMS Software Interfaces | | JSON  HTTP RESTful services | 1st Edition 1.1 |
| RAMS Solution | | J2EE | Java Enterprise Edition 7 |

# Staffing and Training Needs

Table 12 describes the personnel resources needed to plan, prepare, and execute this Master Test Plan.

Table 12: Staffing Resources

|  |  |  |  |
| --- | --- | --- | --- |
| **Testing Task** | **Quantity of Personnel Needed** | **Test Process** | **Duration/Days** |
| Create the Master Test Plan | 1 | Test Preparation | 10 days |
| Develop Test Cases | 2 | Test Preparation | 15 days |
| Establish the Test Environment | 1 | Test Preparation | 5 days |
| Perform System Tests | 2-3 | Product Build | 3-4 days per Sprint |
| UAT Testers | 8-10 | User Acceptance Testing | 5 days |

Table 13 lists the personnel that require training.

Table 13: Training Needs

| **Role** | **Training Need** | **Training Option** | **Estimated Training Hours** |
| --- | --- | --- | --- |
| Principal Investigator (PI) | All functions related to adding/editing/submitting projects/protocol data and supporting documentation. | A train the trainer approach is recommended where Local Site Administrators (LSA) or coordinators will participate in training for PIs. | Training will be conducted by the VA |
| Designees | All functions related to editing projects/protocol data and supporting documentation. | Represented site personnel will attend face-to-face training regarding their role | Training will be conducted by the VA |
| System Administrators (OI&T and LSA) | All system administration functions related to managing user accounts and system controls | Face to face training will be provided for OI&T and local site system administrators. | Training will be conducted by the VA |
| Reviewers | All functions related to committee review, documentation and communications | Face to face training will be provided for Central IRB staff as well as selected field IRB staff. | Training will be conducted by the VA |
| Management | All functions related to reporting. | Face to face training will be provided for managers to understand reporting capabilities. | Training will be conducted by the VA |

# Risks and Constraints

The system must serve users who are located at VA affiliates and universities via the VA Citrix Access Gateway or VPN software. Providing secure access in real time mode will be a major constraint. The system must also comply with VHA Enterprise Architecture and security requirements. Due to compressed timeframes, there is an inherent risk in inability to provide stable operational functionality, regression testing, and defect remediation in accordance to the project schedule. In addition, because the system will serve as a communication portal that is accessible to ORD and research offices, it will require sufficient server space to accommodate numerous tools and resources (i.e. SharePoint 2013) related to personnel, document and data management, many of which require large amounts of storage capacity.

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 (total executed, total passed after review, failed, total not run)
* Percentage of test cases (executed, passed, failed)
* Number of defects attributed to test case/test script creation
* Number of defects identified, listed by severity and priority
* Estimated time and actual time required to test

Attachment A – Approval Signatures

REVIEW DATE:

Signed: Date:

< Program/Project Manager >

Signed: Date:

< Business Sponsor Representative >

Signed: Date:

<Project Team Test Manager>

Appendix A - Test Type Definitions

| Test Type | Definition |
| --- | --- |
| Access Control Testing | A type of testing that attests that the target-of-test data (or systems) are accessible only to those actors for which they are intended, as defined by use cases. Access Control Testing verifies that access to the system is controlled and that unwanted or unauthorized access is prohibited. This test is implemented and executed on various targets-of-test. |
| Benchmark Testing: | A type of performance testing that compares the performance of new or unknown functionality to a known reference standard (e.g., existing software or measurements). For example, benchmark testing may compare the performance of current systems with the performance of the Linux/Oracle system. |
| Build Verification Testing  (Prerequisite: Smoke Test) | A type of testing performed for each new build, comparing the baseline with the actual object properties in the current build. The output from this test indicates what object properties have changed or don’t meet the requirements. Together with the Smoke test, the Build Verification test may be utilized by projects to determine if additional functional testing is appropriate for a given build or if a build is ready for production. |
| Business Cycle Testing | A type of testing that focuses upon activities and transactions performed end to end over time. This test type executes the functionality associated with a period of time (e.g., one-week, month, or year). These tests include all daily, weekly, and monthly cycles, and events that are date-sensitive (e.g., end of the month management reports, monthly reports, quarterly reports, and year-end reports). |
| Capacity Testing | [Capacity](http://www.geekinterview.com/question_details/48768) testing occurs when you simulate the number of users in order to stress an application's hardware and/or network infrastructure. Capacity testing is done to determine the capacity (CPU, Data Storage, LAN, WAN, etc.) of the system and/or network under test. |
| Compliance Testing | A type of testing that verifies that a collection of software and hardware fulfills given specifications. For example, these tests will minimally include: “core specifications for rehosting – ver.1.5-draft 3.doc”, Section 508 of The Rehabilitation Act Amendments of 1998, Race and Ethnicity Test, and VA Directive 6102 Compliance. It does not exclude any other tests that may also come up. |
| Component Integration Testing | Testing performed to expose defects in the interfaces and interaction between integrated components as well as verifying installation instructions. |
| Configuration Testing | A type of testing concerned with checking the programs compatibility with as many possible configurations of hardware and system software. In most production environments, the particular hardware specifications for the client workstations, network connections, and database servers vary. Client workstations may have different software loaded, for example, applications, drivers, and so on hand, at any one time; many different combinations may be active using different resources. The goal of the configuration test is finding a hardware combination that should be, but is not, compatible with the program. |
| Contention Testing | A type of performance testing that executes tests that cause the application to fail with regard to actual or simulated concurrency. Contention testing identifies failures associated with locking, deadlock, livelock, starvation, race conditions, priority inversion, data loss, loss of memory, and lack of thread safety in shared software components or data. |
| Data and Database Integrity Testing | A type of testing that verifies that data is being stored by the system in a manner where the data is not compromised by the initial storage, updating, restoration, or retrieval processing. This type of testing is intended to uncover design flaws that may result in data corruption, unauthorized data access, lack of data integrity across multiple tables, and lack of adequate transaction performance. The databases, data files, and the database or data file processes should be tested as a subsystem within the application. |
| Documentation Testing | Documentation testing is a type of testing that should validate the information contained within the software documentation set for the following qualities: compliance to accepted standards and conventions, accuracy, completeness, and usability. The documentation testing should verify that all of the required information is provided in order for the appropriate user to be able to properly install, implement, operate, and maintain the software application. The current VistA documentation set can consist of any of the following manual types:  Release Notes, Installation Guide, User Manuals, Technical Manual, and Security Guide. |
| Error Analysis Testing | This type of testing verifies that the application checks for input, detects invalid data, and prevents invalid data from being entered into the application. This type of testing also includes the verification of error logs and error messages that are displayed to the user. |
| Exploratory Testing | A technique for testing computer software that requires minimal planning and tolerates limited documentation for the target-of-test in advance of test execution, relying on the skill and knowledge of the tester and feedback from test results to guide the ongoing test effort. Exploratory testing is often conducted in short sessions in which feedback gained from one session is used to dynamically plan subsequent sessions. |
| Failover Testing | A type of testing test that ensures an alternate or backup system properly “takes over” (i.e., a backup system functions when the primary system fails). Failover Testing also tests that a system continually runs when the failover occurs, and that the failover happens without any loss of data or transactions. Failover Testing should be combined with Recovery Testing. |
| Installation Testing | A type of testing that verifies that the application or system installs as intended on different hardware and software configurations, and under different conditions (e.g., a new installation, an upgrade, and a complete or custom installation). Installation testing may also measure the ease with which an application or system can be successfully installed, typically measured in terms of the average amount of person-hours required for a trained operator or hardware engineer to perform the installation. Part of this installation test is to perform an uninstall. As a result of this uninstall, the system, application and database should return to the state prior to the install. |
| Integration Testing | An incremental series of tests of combinations or sub-assemblies of selected components in an overall system. Integration testing is incremental in a successively larger and more complex combinations of components tested in sequence, proceeding from the unit level (0% integration) to eventually the full system test (100% integration). |
| Load Testing | A performance test that subjects the system to varying workloads in order to measure and evaluate the performance behaviors and abilities of the system to continue to function properly under these different workloads. Load testing determines and ensures that the system functions properly beyond the expected maximum workload. Additionally, load testing evaluates the performance characteristics (e.g., response times, transaction rates, and other time-sensitive issues). |
| Migration Testing | A type of testing that follows standard VistA and Health*e*Vet (H*e*V)-VistA operating procedures and loads the latest .jar version onto a live copy of VistA and H*e*V-VistA. The following are examples of the types of tests that can be performed as part of migration testing:  Data conversion has been completed  Data tables are successfully created  Parallel test for confirmation of data integrity  Review output report, before and after migration, to confirm data integrity  Run equivalent process, before and after migration |
| Multi-Divisional Testing | A type of testing that ensures that all applications will operate in a multi-division or multi-site environment recognizing that an enterprise perspective while fully supporting local health care delivery. |
| Parallel Testing | The same internal processes are run on the existing system and the new system. The existing system is considered the “gold standard”, unless proven otherwise. The feedback (expected results, defined time limits, data extracts, etc.) from processes from the new system are compared to the existing system. Parallel testing is performed before the new system is put into a production environment. |
| Performance Monitoring Testing | Performance profiling assesses how a system is spending its time and consuming resources. This type of performance testing optimizes the performance of a system by measuring how much time and resources the system is spending in each function. These tests identify performance limitations in the code and specify which sections of the code would benefit most from optimization work. The goal of performance profiling is to optimize the feature and application performance. |
| Performance Testing | Performance Testing assesses how a system is spending its time and consuming resources. Performance testing optimizes a system by measuring how much time and resources the system is spending in each function. These tests identify performance limitations in the code and specify which sections of the code would benefit most from optimization work. Performance testing may be further refined by the use of specific types of performance tests, such as, benchmark test, load test, stress test, performance monitoring test, and contention test. |
| Performance – Benchmark Testing | A type of performance testing that compares the performance of new or unknown functionality to a known reference standard (e.g., existing software or measurements). For example, benchmark testing may compare the performance of current systems with the performance of the Linux/Oracle system. |
| Performance – Contention Testing | A type of performance testing that executes tests that cause the application to fail with regard to actual or simulated concurrency. Contention testing identifies failures associated with locking, deadlock, livelock, starvation, race conditions, priority inversion, data loss, loss of memory, and lack of thread safety in shared software components or data. |
| Performance – Endurance Testing | Endurance testing, also known as Soak testing, is usually done to determine if the system can sustain the continuous expected load. During soak tests, memory utilization is monitored to detect potential leaks. |
| Performance – Load Testing | A performance test that subjects the system to varying workloads in order to measure and evaluate the performance behaviors and abilities of the system to continue to function properly under these different workloads. Load testing determines and ensures that the system functions properly beyond the expected maximum workload. Additionally, load testing evaluates the performance characteristics (e.g., response times, transaction rates, and other time-sensitive issues). |
| Performance - ProfilingTesting | Performance profiling assesses how a system is spending its time and consuming resources. This type of performance testing optimizes the performance of a system by measuring how much time and resources the system is spending in each function. These tests identify performance limitations in the code and specify which sections of the code would benefit most from optimization work. The goal of performance profiling is to optimize the feature and application performance. |
| Performance – Spike Testing | A performance test in which an application is tested with sudden increment and decrements in the load. The focus is on system behavior during dramatic changes in load. |
| Privacy Testing | A type of testing that ensures that (1) veteran and employee data are adequately protected and (2) systems and applications comply with the Privacy and Security Rule provisions of the Health Insurance Portability and Accountability Act (HIPAA). |
| Product Component Testing | Product Component Testing (aka Unit Testing) is the internal technical and functional testing of a module/component of code. Product Component Testing verifies that the requirements defined in the detail design specification have been successfully applied to the module/component under test. |
| Recovery Testing | A type of testing that causes an application or system to fail in a controlled environment. Recovery processes are invoked while an application or system is monitored. Recovery testing verifies that application or system, and data recovery is achieved. Recovery Testing should be combined with Failover Testing. |
| Regression Test | A type of testing that validates existing functionality still performs as expected when new functionality is introduced into the system under test. |
| Risk Based Testing | A type of testing based on a defined list of project risks. It is designed to explore and/or uncover potential system failures by using the list of risks to select and prioritize testing. |
| Section 508 Compliance Testing | A type of test that (1) ensures that persons with disabilities have access to and are able to interact with graphical user interfaces and (2) verifies that the application or system meets the specified Section 508 Compliance standards. |
| Security Testing | A type of test that validates the security requirements and to ensure readiness for the independent testing performed by the Security Assessment Team as used by the Assessment and Authorization Process. |
| Smoke Test | A type of testing that ensures that an application or system is stable enough to enter testing in the currently active test phase. It is usually a subset of the overall set of tests, preferably automated, that touches parts of the system in at least a cursory way. |
| Stress Testing | A performance test implemented and executed to understand how a system fails due to conditions at the boundary, or outside of, the expected tolerances. This failure typically involves low resources or competition for resources. Low resource conditions reveal how the target-of-test fails that is not apparent under normal conditions. Other defects might result from competition for shared resources (e.g., database locks or network bandwidth), although some of these tests are usually addressed under functional and load testing. Stress Testing verifies the acceptability of the systems performance behavior when abnormal or extreme conditions are encountered (e.g., diminished resources or extremely high number of users). |
| System Testing | System testing is the testing of all parts of an integrated system, including interfaces to external systems. Both functional and structural types of testing are performed to verify that the system performance, operation and functionality are sound. End to end testing with all interfacing systems is the ultimate version. |
| Usability Testing | Usability testing identifies problems in the ease-of-use and ease-of-learning of a product. Usability tests may focus upon, and are not limited to: human factors, aesthetics, consistency in the user interface, online and context-sensitive help, wizards and agents, user documentation. |
| User Functionality Test | User Functionality Test (UAT) is a type of Acceptance Test that involves end-users testing the functionality of the application using test data in a controlled test environment. |
| User Interface Testing | User-interface (UI) testing exercises the user interfaces to ensure that the interfaces follow accepted standards and meet requirements. User-interface testing is often referred to as GUI testing. UI testing provides tools and services for driving the user interface of an application from a test. |

Template Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
| November 2015 | 1.18 | Expanded Section 4.3 to better describe responsibilities for 508 compliance. | Channing Jonker |
| October 2015 | 1.17 | Corrected broken link to 508 URL. | Channing Jonker |
| June 2015 | 1.16 | Updated metadata to show record retention information and required by PMAS, VHA Release Management, Enterprise Operations, and VistA Intake Program | Process Management |
| May 2015 | 1.15 | Reordered cover sheet to enhance SharePoint search results | Process Management |
| March 2015 | 1.14 | Miscellaneous updates including the addition of Performance testing. | Channing Jonker |
| November 2014 | 1.13 | Updated to latest Section 508 conformance guidelines and remediated with Common Look Office Tool | Process Management |
| August 2014 | 1.12 | Removed requirements for ESE Approval Signature | Process Management |
| October 2013 | 1.11 | Converted to Microsoft Office 2007-2010 format | Process Management |
| July 09, 2012 | 1.10 | Added System Design Document to Section 1.2 -Test Objectives as an example | Process Management |
| January 03, 2012 | 1.9 | Updated Approval Signatures for Master Test Plan in Appendix a | Process Management |
| October 13, 2011 | 1.8 | Replaced references to Test and Certification with Independent Test and Evaluation. Replaced references to Certification and Accreditation with Assessment and Authorization. | Process Management |
| October 4, 2011 | 1.7 | Repaired link to Privacy Impact Assessment | Process Management |
| August 23, 2011 | 1.6 | Changed Operational Readiness Testing (ORT) to Operational Readiness Review (ORR) | Process Management |
| April 12, 2011 | 1.5 | Updated the Signatory Authorities in Appendix A in light of organizational changes | Process Management |
| February 2011 | 1.4 | Removed Testing Service Testing and Operational Readiness Testing; added Enterprise System Engineering Testing.  Changed Initial Operating Capability Testing to Initial Operating Capability Evaluation | Process Management |
| January 2011 | 1.3 | Repaired broken link in section 1.4 | Process Management Service |
| August 2010 | 1.2 | Removed OED from template | Process Management Service |
| December 2009 | 1.1 | Removed “This Page Intentionally Left Blank” pages. | OED Process Management Service |
| July 2009 | 1.0 | Initial ProPath release | OED Process Management Service |