# Rx Refill\_MTP

**Department of Veterans Affairs**   
**Mobile Applications (Apps) Phase Two (MAP2)**  
**Rx Refill - Prescription Refill** **– Master Test Plan**   
**April** **2016**  
**Version 0.6**

# Introduction

Veterans Health Administration (VHA), Patient Care Services (PCS), Pharmacy Benefits Management (PBM) is requesting modification to the exiting Rx Refill – Prescription Refill (Rx Refill) mobile application.  
The current Hypertext Mark-up Language five (5) (HTML5) version of the Rx Refill app., which is not in production due to compliance and 508 issues. The Application must first be updated to meet compliance requirements and then have the agreed upon enhancements updated to include self-service Rx medication tracking. The VA now sends the majority of outpatient Rx's to Veteran patients via mail. In order to track the delivery of mailed Rx medications, VA patients must call their local VAMC directly and wait for a representative to look up their order and connect them to the carrier responsible for delivering the shipment.   
The current Rx Refill mobile application shall be enhanced to allow Veterans to track Rx medication shipments to include the status of their mailed Rx medication and expected delivery date and time. The Rx Refill mobile app shall be modified to display relevant mail tracking information, with links to send the user or display status information from common delivery services such as the United States Postal Service (USPS), FedEx and United Parcel Service. The content displayed on the enhanced Rx Refill app shall mirror the information in My Health**e**Vet (MHV), making use of existing MHV services, where appropriate.  
Additionally, the existing Rx Refill application shall be enhanced to include a picture of the medication including the name of the medication. VA will provide a web service link to the Medication Image Library (MIL) system in the VA Corporate Data Warehouse (CDW).

## Test Plan Purpose

The purpose of this Master Test Plan (MTP) for the RX Refill application is to define all testing that must be performed on this mobile applications before the application can be deemed ready for submission to the VA Compliance Review and Verification and Validation (V&V) testing processes. An Agile testing approach will be followed by the MAP2 Hewlett Packard Enterprise (HPE) Test teams (i.e., Developers and System Quality Assurance (SQA) Testers for the testing of the functionalities as defined in the various program documents including the Performance Work Statement (PWS) for the Mobile Applications (Apps) Phase Two (MAP2), requirements documentation, and system design specifications. This MTP is designed to ensure that the objectives of testing are successfully met. It serves as a guideline for MAP2 test development and test execution.   
Where applicable, this test plan will reference other existing program documentation. The purpose of referencing existing documentation is to minimize information redundancy and also to minimize the associated maintenance efforts required to keep multiple documents synchronized. Supporting documentation for the Rx Refill application can be found on the Rx Refill App wiki page (<https://DNS/display/RXR/P2+Documentation>). Table 1 provides a list of program documentation referenced within this test plan.

**Table 1: Reference Documents**

|  |  |  |  |
| --- | --- | --- | --- |
| **Document** | **Author** | **Status** | **Version** |
| **MAP2 – Performance Work Statement – TAC-14-16925** | **VHA** | **Client Approved** | **1.0** |
| **MAP2 – Program Master Test Plan** | **Ephraim Blevins** | **Baseline** | **1.0** |
| **MAP2 – Contract Project Management Plan** | **Eddie Urso** | **Baseline** | **1.0** |
| **Rx Refill MAP2 Master Schedule** | **Sandi Winters, Debra Wolf** | **Baseline** | **TBD** |
| **MAP2 Issues and Risk Logs/DAIR** | **Eddie Urso, Dave Getman** | **Work in Progress** | **TBD** |
| **Rx Refill\_BRD** | **David Getman** | **Work in Progress** | **1.0 2/17/2015** |
| **Rx Refill\_RTM** | **Megan Lucas** | **Work in Progress** | **V6** |
| **Rx Refill\_User Stories – Sprint Planning** | **Laurie Kirby, Megan Lucas** | **Work in Progress** | **TBD** |
| **RxRefill\_Wireframes\_Mobile** | **Laurie Kirby, Christine Durose** | **Work in Progress** | **V6** |
| **RxRefill\_Wireframes\_Tablet** | **Laurie Kirby, Christine Durose** | **Work in Progress** | **V6** |
| **Rx Refill (RXR) SDD Addendum** | **Aaron Lucas** | **Work in Progress** | **2.0** |
| **Rx Refill\_RSD Addendum** | **Megan Lucas** | **Work in Progress** | **V6** |
| **Rx Refill (RXR) Service Interface Control Document (ICD)** | **Aaron Lucas** | **Work in Progress** | **2.16** |

This Test Plan also supports the following specific objectives:

* Identifies the items that should be targeted by the tests
* Identifies the motivation for, and ideas behind, the test areas to be covered
* Outlines the testing approaches that will be used
* Identifies the required resources and provides an estimate of the test efforts
* Describes the testing environment
* Lists the deliverable elements of the test project

## Test Objectives

This Master Test Plan supports the following objectives:

* To provide test coverage for 100% of the documented requirements – functional and non-functional contained in applicable application requirement documentation functional contained in applicable requirement documentation (i.e., the Business Requirements Documents (BRD), the Requirement Specifications Document (RSD), and User Stories)
* To validate the Acceptance Criteria of each User Story for the Rx Refill application has been met
* Provide coverage for System Design Document (SDD) addendum elements
* To execute 100% of the test cases during all testing levels
* To perform handoffs with V&V, business sponsors, and Compliance bodies for independent testing, compliance reviews, and User Functionality testing
* To create, maintain, and control the test environment

## Acronyms

This section contains a list of acronyms used in this document.

|  |  |
| --- | --- |
| **Acronym** | **Description** |
| ADM | Agile Development Methodology |
| ADR | Administrative Data Repository |
| BRD | Business Requirements Document |
| CCB | Change Control Board |
| CDW | Corporate Data Warehouse |
| COR | Contracting Officer's Representative |
| EAS | Enterprise Application Security |
| EHR | Electronic Health Record |
| ESE | Enterprise Systems Engineering |
| HPE | Hewlett Packard Enterprise |
| IAM | Identity and Access Management |
| ID | Identification |
| IOC | Initial Operating Capability |
| MA | Mobile Application |
| MADW | Mobile Application Development Workflows |
| MAE | Mobile Application Environment |
| MDWS | Medical Domain Web Services |
| MHV | My Health**e**Vet |
| MIL | Medication Image Library |
| MTP | Master Test Plan |
| MVI | Master Veteran Index |
| NFR | Non-Functional Requirement |
| NIST | National Institute of Standards and Technology |
| NSOC | Network Security Operations Center |
| OIA | Office of Information Assurance or Office of Informatics and Analytics |
| OI&T | Office of Information and Technology |
| O&M | Operations and Maintenance |
| Rx Refill | Rx Refill Delivery Tracking and Image Enhancements |
| PHI | Private Health Information |
| PII | Personally Identifiable Information |
| PMAS | Project Management and Accountability System |
| RSD | Requirements Specification Document |
| RTM | Requirements Traceability Matrix |
| SDD | System Design Document |
| SQA | Software Quality Assurance |
| SSP | System Security Plan |
| UFT | User Functionality Test |
| URL | Uniform Resource Locator |
| VA | Department of Veterans Affairs |
| VAHA | VA Health Adapter |
| VAMC | VA Medical Center |
| VAMF | VA Mobile Framework |
| VDD | Version Description Document |
| VHA | Veterans Health Administration |
| VistA | Veterans Health Information Systems and Technology Architecture |
| VP | Verification Point |
| V&V | Verification and Validation |
| WMS | Web and Mobile Solutions |
| WASAs | Web Application Security Assessments |

## Roles and Responsibilities

Table 2 lists the key roles and their responsibilities for this Master Test Plan. The roles defined below are members of the MAP2 HPE Program team.

**Table 2: Roles and Descriptions**

|  |  |
| --- | --- |
| **Role** | **Description** |
| Developers | Persons that build or construct the product/product component and complete the required testing. |
| Development Manager | Person responsible for ensuring mobile application components have gone through adequate unit and component-integration testing by development team before turning over to the System Quality Assurance (SQA) Test Team for system testing. Responsible for approving the MTP. |
| Project Manager | Person responsible for overseeing the development and testing of mobile applications under the MAP2 program. Responsible for approving the MTP. |
| Program Manager | Person that has overall responsibility for the successful planning and execution of the MAP2 program. The Program Manager will be responsible for approving the MTP. |
| Stakeholders | Persons that hold a stake in a situation in which they may affect or be affected by the outcome. |
| SQA Test Planner/Engineer | An experienced test analyst or member of the Test Team that leads and coordinates activities (i.e., ensure test environment setup, support Initial Operating Capability (IOC) and V&V testing activities, etc.) related to all aspects of testing for an individual mobile application. Responsible for developing a MTP for each application. |
| SQA Test Manager | Person that has overall responsibility for creation and implementation of the program-level MTP. Person responsible for leading and coordinating activities related to all aspects of testing based on an approved application-level MTPs and schedules. |
| SQA Test Team | Persons that execute tests and ensure the test environment will adequately support planned test activities. |
| Test Environment Team | Persons that establish, maintain, and control test environments. |

## Processes and References

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

* VA's Mobile Application Development Life Cycle
* VA's Mobile Application Project Management Accountability System
* MADW-3 Application Development document
* MADW-4 Verification and Validation
* MADW-5 Compliance Review
* MADW-6 User Acceptance Test
* MADW-7 Field Test (IOC)

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

* Mobile Development Application Life Cycle (<http://DNS/content/developing-va-apps>)
* ProPath

(<http://DNS/process/propath>)

* Section 508 Office Web Page

(<http://DNS/508workgroup>)

* Privacy Impact Assessment - Privacy Service

(<http://DNS/Privacy_Impact_Assessment.asp>)   
Note: For 508 Compliance, copy and paste the above URLs into your browser to reach each site. Also, the above hyperlinks/URLs prefaced with DNS are only accessible with access to VA network.

# Items to Be Tested

During design of the Rx Refill mobile application, the MAP2 HPE Development and SQA Test teams will define the changes required to deploy and test the Rx Refill application into the Mobile Application Environment (MAE). After receipt of the SDD Addendum and/or the Interface Control Document (ICD) for the Rx Refill application, the MAP2 HPE SQA Test team will determine the kind of integration testing that needs to occur for the Rx Refill application in addition to the other system testing to occur on this application.

## Overview of Test Exclusions

Note: Additional information for the Overview of Test Exclusions section will be added/updated in future iterations, as needed.  
The following components and features and combinations of components and features will not be tested:

* Non-functional requirements that are not applicable to the Rx Refill application developed by the MAP2 HPE Development team
* Testing on Android mobile devices (i.e. phone, tablet, etc.)

|  |  |  |
| --- | --- | --- |
| **Epic/Story #** | **JIRA Story #** | **Epic/Story Requirement** |
| RXR-528 | RXR-538 | As a Rx Refill team member, I want to create the single page application template so that the RXR application has a defined structure. |
| RXR-534 | RXR-556 | As a Rx Refill developer, I want to create the project structure and shell along with build scripts for the VAMF Rx Refill Service so that a web service project can be initiated to provide Rx Refill and Tracking information for the Rx Refill application. |
| RXR-534 | RXR-557 | As a Rx Refill developer, I want to create an active prescriptions endpoint within the VAMF Rx Refill Service so that active prescriptions from the Enterprise Tracking and Refill service can be provided to the Rx Refill Application. |
| RXR-534 | RXR-558 | As a Rx Refill developer, I want to create a historical prescriptions endpoint within the VAMF Rx Refill Service so that historical prescriptions from the Enterprise Tracking and Refill service can be provided to the Rx Refill Application. |
| RXR-534 | RXR-559 | As a Rx Refill developer, I want to create a prescription refill endpoint within the VAMF Rx Refill Service so that a prescription refill request can be submitted to the Enterprise Tracking and Refill service via the Rx Refill Application. |
| RXR-534 | RXR-560 | As a Rx Refill developer, I want to create a trackable prescriptions endpoint within the VAMF Rx Refill Service so that trackable prescriptions from the Enterprise Tracking and Refill service can be provided to the Rx Refill Application. |
| RXR-534 | RXR-561 | As a Rx Refill developer, I want to create a trackable prescription detail endpoint within the VAMF Rx Refill Service so that detailed tracking information on a prescription from the Enterprise Tracking and Refill service can be provided to the Rx Refill App. |
| RXR-534 | RXR-562 | As a Rx Refill developer, I want to create a medication refill request endpoint within the VAMF Rx Refill Service so that details on prescriptions that have been successfully refilled by the VAMF Rx Refill app. |
| RXR-535 | RXR-564 | As a Rx Refill developer, I want to create a medication image metadata endpoint within the VAMF MIL Service so that medication image metadata from the CDW MIL database can be provide to VAMF applications. |
| RXR-535 | RXR-565 | As a Rx Refill developer, I want to create a MIL image endpoint within the VAMF MIL Service so that medication image from the CDW MIL database can be provide to VAMF applications. |
| RXR-535 | RXR-566 | As a Rx Refill developer, I want to create a MIL images endpoint within the VAMF MIL Service so that a list of medication images from the CDW MIL database can be provide to VAMF applications. |
| RXR-535 | RXR-567 | As a Rx Refill developer, I want to create a MIL images endpoint within the VAMF MIL Service so that a list of medication images from the CDW MIL database can be provide to VAMF applications. |
| RXR-535 | RXR-563 | As a Rx Refill developer, I want to create the project structure and shell along with build scripts for the VAMF MIL Service so that a web service project can be initiated to provide medication images from the MIL for VAMF applications. |
| RXR-534 | RXR-973 | As a Rx Refill developer, I want to develop a mechanism to lookup a Veteran's mhvCorrelationId, so that I can implement business logic within the RXR Session endpoint. |
| RXR-534 | RXR-974 | As a Rx Refill developer, I want to implement oAuth security for all non-public webservice endpoints, so that endpoints are protected. |
| RXR-1010 | RXR-1012 | As an RxR Developer, I need to incorporate the user friendly error description that were provided by the Business Owner into the exception handling of each webservice endpoint, so that user friendly error messages can be provided to a calling application. |
| RXR-529 | RXR-1017 | As a Rx Refill developer, I want to create a refillable prescription endpoint within the VAMF RxRefill Service, so that only refillable prescriptions can be provided to the RxRefill application. |
| RXR-1010 | RXR-1009 | As an Rx Refill Business Owner, I need to provide user friendly error descriptions, so that Veterans have meaningful error descriptions to allow them to resolve any potential errors. |
| RXR-1010 | RXR-1013 | As an Rx Refill developer, I need to incorporate the user friendly error description that were provided by the Business Owner into the exception handling of each webservice endpoint, so that user friendly error messages can be provided to a calling application. |
| RXR-1133 | RXR-1130 | As a RXR developer, I would like to update/verify that the login functionality is consistent with other MAP2 apps and to integrate it with ROA, so that a common workflow for login is achieved. |
| RXR-1206 | RXR-1188 | As an RxR developer, I would like to upgrade Handlebars from version 1.0.12 to version 2.0.0, so that the environment is running on the most up to date TRM approved version. |
| RXR-1206 | RXR-1189 | As an RxR developer, I would like to upgrade Jasmine from version 2.1 to version 2.3, so that the environment is running on the most up to date TRM approved version. |
| RXR-1206 | RXR-1190 | As an RxR developer, I would like to upgrade SASS, so that the environment is running on the most up to date TRM approved version. |
| RXR-1206 | RXR-1191 | As an RxR developer, I would like to upgrade the RXR Resource service to align Jersey libraries with those running in the Health Adapter, so that the libraries running behave as expected without library collisions. |
| RXR-1206 | RXR-1193 | As an RxR developer I would like to upgrade the MIL Resource service to align Jersey libraries with those running in the Health Adapter, so that the libraries running behave as expected without library collisions. |
| RXR-1235 | RXR-1236 | As an RXR (MT) Developer, I need to incorporate the oAuth Lib Spring MVC Controller into my application so that the Front End can query MT for oAuth client details and can request MT to broker calls for obtaining oauth tokens given the oAuth code. |
| RXR-1235 | RXR-1237 | As a UI Developer, I want to utilize the Auth Functions wrapper to initialize the application, so that user can see if they are authenticated properly. |

## Overview of Test Inclusions

Note: Additional information for the Overview of Test Inclusions section will be added/updated in future iterations, as needed.  
The following components, features, and combinations of components and features will be tested:

* Mobile Web site (HTML5)
* Testing on IOS mobile devices, e.g., GFE iPad using Safari web browser.
* Testing on Windows laptop using IE-11, Chrome, Firefox, web browsers.
* For phone view only, emulators are used as no GFE device is available.

|  |  |  |
| --- | --- | --- |
| **Epic/Story #** | **JIRA Story #** | **Epic/Story Requirement** |
| RXR-528 |  | Web Application Template |
| RXR-528 | RXR-539 | As a Rx Refill team member, I want to create the footer area of the single page application so that the RXR application displays general information about the application and the user. |
| RXR-528 | RXR-690 | As a Rx Refill team member, I want to create the home/landing page for the RXR application so that the user has quick access to the major elements provided by the application. |
| RXR-528 | RXR-691 | As a Rx Refill team member, I want to create a quick access navigation element in the banner of the RXR application so that the user has quick access to the major elements provided by the application. |
| RXR-528 | RXR-692 | As a Rx Refill team member, I want to create a help page so that the user is provided with help information relating to the application by the application. |
| RXR-528 | RXR-693 | As a Rx Refill team member, I want to create an about page so that the user is provided with help information relating to the application by the application. |
| RXR-528 | RXR-694 | As a Rx Refill team member, I want to integrate with the VAMF authorization mechanism for login so that the user can be authenticated prior to entering the application. |
| RXR-528 | RXR-696 | As a Rx Refill project team, we must ensure that following 15 minutes of inactivity by the current user our application time-outs so that our application efficiently used the limited resources that maybe available to the user. |
| RXR-529 |  | Refillable VA Medications |
| RXR-529 | RXR-697 | As a Veteran, I want to see my active/refillable VA prescriptions, so that I can view detailed information on a prescription and refill it when appropriate. |
| RXR-529 | RXR-698 | As a Veteran, I want to sort my active/refillable VA prescriptions, so that I can view detailed information on a prescription and refill it when appropriate. |
| RXR-529 | RXR-699 | As a Veteran, I want to see medication detail on my active/refillable VA prescriptions, so that I can easily view detailed information on a prescription. |
| RXR-529 | RXR-700 | As a Veteran, I want to request a refill on my active/refillable VA prescriptions, so that I can refill my VA medications. |
| RXR-530 |  | Track Delivery |
| RXR-530 | RXR-545 | As a Veteran I want the ability to see a list of my VA prescriptions that have been shipped so I can determine when I will receive my VA medication by mail. |
| RXR-530 | RXR-546 | As a Veteran, I want to sort my trackable prescriptions, so that I can ultimately view detailed information on tracking and shipment. |
| RXR-530 | RXR-547 | As a Veteran, I want the option to see an important information screen when I select tracking delivery details so that I can be informed on tracking detail information. |
| RXR-530 | RXR-548 | As a Veteran, I want to see detail tracking information for my trackable prescriptions, so that I track and determine where my medication is. |
| RXR-530 | RXR-549 | As a Veteran, I want to see an image of my medication for trackable prescriptions, so that I can compare what was shipped to what I received |
| RXR-530 | RXR-550 | As a Veteran, I want to track my prescription, so that I can determine where it is. |
| RXR-531 |  | Prescription History |
| RXR-531 | RXR-551 | As a Veteran, I want to see my VA prescriptions, so that I can view detailed information on a prescription |
| RXR-531 | RXR-552 | As a Veteran, I want to sort my VA prescriptions on the prescription history screen, so that I can view detailed information on a prescription |
| RXR-531 | RXR-553 | As a Veteran, I want to see medication detail on a prescription selected from the prescription history screen, so that I can easily view detailed information on a prescription. |
| RXR-532 |  | Medication Image |
| RXR-532 | RXR-554 | As a Rx Refill team member, I want to create a mechanism for resizing MIL images, so that the images can be used on a mobile platform. |
| RXR-533 |  | Disclaimer |
| RXR-533 | RXR-555 | As a Rx Refill team member, I want to provide a disclaimer for the Veteran, so that the application informs the user to take medication as prescribed. |
| RXR-536 |  | Medication List on MyHealtheVet |
| RXR-536 | RXR-568 | As a Veteran using the RxRefill application, I need access to the My Complete Medication List via a link to MHV, so that I can view all my medications. |
| RXR-536 | RXR-569 | As a Veteran using the RxRefill application, I need access to my Blue Button Download via a link to MHV, so that I access my Blue Button data. |
| RXR-536 | RXR-570 | As a Veteran using the RxRefill application, I need access to my Health Summary (VA CCD) via a link to MHV, so that I can view all my CCD data. |
| RXR-537 |  | Icebox |
| RXR-537 | RXR-571 | As a Veteran, I want a way to track delivery of a refill for my prescription directly from the refill details. |
| RXR-537 | RXR-572 | As a Veteran, I want the ability to call my pharmacy/provider so that I can talk to them directly if I have questions regarding my prescription. |
| RXR-537 | RXR-573 | As a Veteran, I want the ability to send a secure message to my pharmacist or provider so I can talk to them directly if I have questions regarding my prescription. |
| RXR-537 | RXR-574 | As a Veteran, I need access to an authenticated user's COMPLETE medication list as defined by the Essential Med Directive (EMD). |
| RXR-537 | RXR-575 | As a Veteran, I want to be able to understand the meaning of a term or status and how it relates to similar Rx Refill interfaces such as MHV Rx Refill and Audiocare. |
| RXR-537 | RXR-576 | As a Veteran, I want to be able to replace the MIL image with a picture of my VA Pharmacy/CMOP dispensed medication. |
| RXR-537 | RXR-577 | As a Veteran, I want to be prompted to notify my VA Pharmacy/provider of any discrepancies I have found regarding the medication images. |
| RXR-537 | RXR-578 | As a Veteran, I want a warning of a chronic medication prescription that is about to expire so that that the Veteran can take action before the medication runs out. |
| RXR-537 | RXR-579 | As a Veteran, I want to be able to submit a refill request by using my mobile device to scan a prescription label bar code. |
| RXR-537 | RXR-580 | As a Veteran, I want the ability to get an automated Renewal request for non-refillable prescriptions similar to Audio Care. |
| RXR-537 | RXR-581 | As a Veteran, I want a refill reminder sent to my mobile device or email account. |
| RXR-537 | RXR-582 | As a Veteran, I want to be alerted on my mobile device when a medications is about to be sent so that (if needed) I have plenty of time to stop a refill from being sent (i.e. alert me 10 days prior to the next fill date. |
| RXR-537 | RXR-583 | As a Veteran, I want a pill reminder to remind me when I need to take my pill based on the frequency I enter OR the frequency contained in my VA prescription. |
| RXR-537 | RXR-584 | As a Veteran, I want the ability to utilize a "Facility Locator" to find my preferred VA Healthcare facility or facilities closest to my mobile device. |
| RXR-537 | RXR-585 | As a Veteran, I want links to future/existing education on medication information. |
| RXR-537 | RXR-586 | As a Veteran, I want Drug Information lookup specific to the prescribed medication using the VA Health Library and the VA National Formulary. |
| RXR-537 | RXR-587 | Additional requirements as requested by HFE contained within "Dec2014\_Interviews\_Rx Refill\_ABInterviews\_Report" and agreed upon by the BO, PBM: |
| RXR-1010 | RXR-1014 | As a Veteran, I want to see user friendly error messages when an error occurs, so that I understand how to correct the error or be provided with a mechanism for assistance. |
| RXR-529 | RXR-1018 | As a Veteran, I want to see only refillable prescriptions in the Refillable VA Medications screen, so that I can quickly review and refill my refillable medications. |
| RXR-1134 | RXR-1107 | As a Veteran, I want to approve the release of my health information, so that I may be granted access to my information via the RXR mobile application. |
| RXR-530 | RXR-1015 | As a Veteran, I want to see "Image not available" when an image for medication is selected and is not available for trackable medication, so that I see an image is not available in clear, readable language. |
| RXR-530 | RXR-1016 | As a Veteran, I want to see an image disclaimer before viewing the selected medication for trackable medication, so that I am aware the image is for identification purposes only. |
| RXR-1133 | RXR-1131 | As a Veteran, I want the application to integrate with the VAMF authorization mechanism for logout so that the user can see they are logged out of the application. |
| RXR-1235 | RXR-1238 | As a SQA Tester, I need to regression test the use of Token validation services with Authorization Services 4.4 to ensure that the service protection continues to work |
| RXR-1235 | RXR-1239 | As a SQA Tester, I need to ensure that the Inactivity timeout functionality continues to function as before with Authorization Services 4.4, so that I can verify the upgrade did not impact timeout. |
| RXR-1235 | RXR-1240 | As a SQA Tester, I need to ensure that the Logout functions continue to perform as before with Authorization Services 4.4, so that I can verify that I can logout of the system. |

For a list of non-functional requirements to be tested, refer to the Rx Refill RSD addendum document located on the VA Rx Refill App wiki page: (\_Toc402796789" class="external-link" rel="nofollow"> DNS /display/RXR/PWS+Deliverables).

# Testing Approach

Note: Additional information for the Test Approach section will be added/updated in future iterations, as needed.  
The MAP2 Program team will be utilizing an Agile Development Methodology (ADM) as well as using best practices in agile testing techniques for all applications developed as part of the MAP2 program.   
Developers assigned to a Scrum team will perform product component or unit testing on their local development workstation and component-integration testing for each Sprint iteration using the MAE Development environment, next the MAP2 HPE SQA tester assigned to the Scrum team will perform feature (aka functional) verification testing using the MAE Development environment, which involves testing against the user story's acceptance criteria. In addition, a parallel process of regression testing will occur throughout the Sprint iteration. This involves re-running the automated and/or manual product-component tests and feature verification tests from the current Sprint iteration and previous iterations via a continuous integration framework. This level of testing will be completed in the MAE Development environment.   
During the application development stage, the MAP2 HPE Development team will perform a static code analysis on the Rx Refill application using the HPE Fortify tool. Any issues found will be remediated. MAP2 HPE SQA Test team members will be engaged in this process to review static analysis output results as well as verifying the code has been remediated.   
Members of the MAP2 HPE SQA Test team will perform the System Test level, which starts once the first user story is ready for this test level. System testing will consist of executing functional, end-to-end, system-integration, full regression, performance, etc. using the MAE Development-Test environment.   
System testing will begin at the start of Sprint 2, but no later than Sprint 4. The timing of the System Test level, including what test types will be performed, will be addressed in [Section 2.2](#RxRefill_MTP-_Overview_of_Test) of this document in conjunction with the applicable Sprint planning meeting where the System Test level is expected to begin. After completion of system testing, the final build will be submitted for VA Compliance Reviews and V&V Test processes.

## Testing Processes

The following testing processes will be employed by the MAP2 HPE SQA Test team:

* Develop the Master Test Plan (this document)
* Establish the Test Environments
  + MAE Development
  + MAE Development-Test
* Prepare Test Scenarios, Cases, Scripts and Data
* Conduct the Tests
* Verify Test Results
* Document Test Results
* Update the Test Status
* Identify and Resolve Discrepancies
* Retest remediated defects

Representatives of the user base perform user acceptance testing and independent teams perform verification and validation testing and compliance body reviews. While the strategy is standard to best practices in testing, there are some unique aspects to the mobile and MAP2 test approach.

1. The MAP2 program will be following an Agile Development Methodology (ADM), which integrates testing into the entire development process versus a separate test phase at the end of the development cycle as well as promotes constant communication with the business owner's team. This yields application software that implements the intended requirements and will not yield many requirements surprises in the downstream testing levels.
2. The ADM followed by the MAP2 program team promotes the following Agile test-related strategies:

* MAP2 HPE SQA tester involvement in user story refinement and specification of acceptance criteria as part of story definition
* Automate product component, component-integration, system and acceptance testing as much as possible
* Incorporation of automated tests into the continuous build environment
* Small (minimal) hand-offs between Scrum team (developers, business analyst, testers) members (they work together)
* Lightweight (fast-turnaround) defect tracking and management (defects are identified and fixed within the Sprint)
* Exploratory testing
* Test in the same Sprint in which the feature is developed
* Track test coverage
* Incorporation of testing into the "Definition of Done"
* Consideration of how various facets of testing will be performed during the Release Cycle (e.g., stronger emphasis on performance testing during 'hardening Sprint iterations')
* Separate Black-box (behavioral) unit testing from white-box (code-structure based) unit testing.
* Product component or unit testing is typically the responsibility of developers. However, for agile projects, it is recommended that the responsibility for white-box testing be assigned to developers, while the testers can focus on black-box (feature-driven) unit testing concurrently. This provides complementary but comprehensive focus to unit testing while also enabling early detection of defects that would typically be found in later test levels (e.g., system testing) and reduces cycle time for defect fixes and rework.
* Integrate Automated Tests with the Continuous Integration Build Environment
  + Automated tests will run against the continuous build environment

1. The compliance reviews that must be performed on a mobile application are defined based on the level of risk to VA. The Rx Refill application has been assessed as a category 4 application. The risk level is defined as one of the four levels shown in Table 3. The compliance bodies that must review those applications are also shown.

**Table 3: Risk Level Definition**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Mobile Application Classification** | **1a - Very Low: Non-OI&T**\*\* | **1 - Very Low** | **2 - Low** | **3 - Medium** | **4 - High** |
| **Compliance Review Body** | **Does not utilize VA resource or OI&T Funding** | **Does not utilize VA resources** | **Read only access to VA resources** | **Write access to VA or external resources** | **Read and/or write access to VA or external sensitive resources** |
| V&V | Required | Required | Required | Required | Required |
| Business Owner Acceptance | Required | Required | Required | Required | Required |
| Patient Safety Assessment (OIA) | Required | Required | Required | Required | Required |
| 508 Accessibility (OI&T)\* | Required | Required | Required | Required | Required |
| Code Review | Required | Required | Required | Required | Required |
| Usability Testing (OIA) | Required | Required | Required | Required | Required |
| **Mobile Application Classification** | **1a - Very Low: Non-OI&T**\*\* | **1 - Very Low** | **2 - Low** | **3 - Medium** | **4 - High** |
| **Compliance Review Body** | **Does not utilize VA resource or OI&T Funding** | **Does not utilize VA resources** | **Read only access to VA resources** | **Write access to VA or external resources** | **Read and/or write access to VA or external sensitive resources** |
| User Interface (OIA) | Required | Required | Required | Required | Required |
| VA Branding (OPIA) | Required | Required | Required | Required | Required |
| Privacy and Application Security (OIA) | Required | Required | Required | Required | Required |
| Sustainment Plan\* | Required | Required | Required | Required |  |
| System Performance Impact Assessment (OI&T)\* | Required | Required | Required |  |  |
| Enterprise Security | Required | Required | Required |  |  |
| Data and Terminology Standards Compliance (OIA)\* | Required | Required |  |  |  |
| * Not Required for Field Test |  |  |  |  |  |

* In order to qualify for this category, the Mobile Application (MA) must meet the following criteria:
* Does not access Personally Identifiable Information (PII) or Personal Health Identifiers (PHI) data
* Does not access the VA network
* Does not read and/or write to VA or external database
* Does not store data in any VA database
* Does not utilize the VA Mobile Framework
* Does not receive funding (including sustainment and future release funding) from the Office of Information and Technology (OI&T)

Note: The VAMF is considered a VA resource. With this definition, the following is implied:

* If an application is hosted in the VA Mobile Framework (VAMF), it cannot be a Very Low app. It is at least a Low if it only reads pages within the VAMF or reads data within the VAMF
* If an application uses a service in the VAMF, but is not installed in the VAMF, then it must be at least a Low app

1. The MAE provides development, test, integration, and other environments that provide standardized mobile technologies and toolsets. The Web and Mobile Solutions (WMS) team maintains the MAE. The WMS team has defined and continues to define methods for using the tools to streamline the processes and make the compliance steps and documentation easier to find and read. These processes are intended to speed up the testing process but still maintain quality in order to ensure the products work as intended and pass government regulations, such as National Institute of Standards and Technology (NIST) security and privacy guidelines, 508 compliance, and VA patient safety regulations. Tools, such as JIRA and Confluence (wiki), provide the tracking mechanisms and document management that allow for these streamlined processes.
2. Each mobile application will document test scripts and test cases in a Microsoft Excel workbook and include the workbook as an attachment to the application's Master Test Plan (MTP). The Requirements Traceability Matrix (RTM) for the mobile application will be updated to map the applicable test case identification (ID) for the test cases that provides test coverage for a user story. The MTP will document the overarching approach and test environment such that the test cases and scripts contained as an attachment to the MTP need to only address specifics of the mobile application being tested. The HPE SQA Test Team will complete the test cases and test scripts for all applicable functional and non-functional requirements in scope for the mobile application being tested.

Currently, the MAP2 program team is forecasting that it will take five Sprint iterations to complete all development and testing activities (up through the System Test level), for each new application. With that in mind, system testing will be a parallel testing activity that will begin once the user stories from Sprint 1 for a given application have completed feature verification testing and met the definition of "done" by the VA Business Owner of that application. At the earliest, system testing for the Rx Refill application will begin no earlier than Sprint 2 and will be completed by Sprint 5.

## Product Component Test

All components of the Rx Refill application will go through thorough product component (aka unit) testing by the MAP2 HPE Development team. The product component testing will follow processes as specified in the VA's Mobile Application Development Life Cycle and documented using the approved templates. Product component testing will be performed using the developer's local workstation computer. The steps included in the process include:

* Analyze requirements and design model to understand the application functionality and dependencies
* Identify all the services and interfaces affected by the application
* Specify all the routines or services that are called from various locations
* Execute tests on prioritized options
* Execute tests with different combinations of options and data (e.g., test with minimal data entered and test with maximal data entered)
* Perform exploratory testing (i.e., randomly exercise the module, object, and options based upon domain knowledge, past performance, and expertise)
* Record the actual test results

## Component Integration Test

As application components are completed by the Rx Refill MAP2 HPE Development team and integrated into the Rx Refill application, the application will be tested to ensure functionality and requirement satisfaction. Component-Integration testing is performed to expose defects in the interfaces and interaction between integrated components.  
The MAP2 HPE Development team will execute the code analysis tool Fortify on the mobile application for their mobile application. This tool will analyze the code and provide any errors or warnings that would be identified later during V&V or security testing. The MAP 2 HPE Development team will fix the critical and high items identified prior to turning the application over for V&V testing.  
Developers are expected to test the components of the application that they are responsible for and developers assigned to the Scrum team need to test the application components as a whole before turning the application over to the Scrum MAP2 HPE SQA tester team member for each sprint.  
The Component-Integration Test will be known as developer led testing and will occur in the MAE Development environment. This environment contains mock services for system interfaces.   
After component-integration testing is completed, the developer responsible for the application will provide a report listing unit test results indicating that all unit test cases have passed and the application is ready for feature verification testing. The MAP2 HPE SQA Test team member assigned to the Scrum team will perform feature verification testing and work with the other Scrum team members to ensure all defects identified are fixed and retested or resolved.  
It is expected that manual testing will occur during this level, but that automated test scripts will be written by the MAP2 HPE Development SQA team members to facilitate this Test level. The MAP2 HPE Development and Test Managers will help determine the scope of automated test script development for this Test level.

## System Test

Members of the MAP2 HPE SQA Test team will perform the System Test level, which starts once the first user story is ready for this test level. System testing will consist of executing functional, end-to-end, system-integration, full regression, performance, etc. test cases in the MAE Development-Test environment. Test coverage will include all in-scope functional and non-functional requirements.   
The MAE Development-Test environment will contain live interfaces with as many VA enterprise systems and/or services as possible to perform valid System Tests. Mock services will only be used where absolutely required (for example, if a given VA system does not have a test environment that can be connected to the MAE). The MAP2 HPE SQA Test team will be in control of the test data, and developers should not have access to this environment to make changes. It may be required for developers to have read access such that they can debug issues when they arise in System Test.  
Specifically, the MAP2 HPE SQA Test team will perform the following functions:

1. Write the test scripts including:

* Test Case Descriptions
* Pre-conditions/Post-conditions
* Test Cases
* Test Script specifications including:
* Procedure steps
* Verification Points (VP)
* Expected Results (ER)
* Comments

1. Update the Requirements Traceability Matrix created earlier by an Analyst adding the Test Case ID to the matrix
2. Execute the test scripts
3. Report results based on the Actual Results compared to the expected results
4. Create bug issues in JIRA to document defects
5. Perform testing of remediated defects

* As fixes to the defects are planned (and put into the backlog in JIRA), the Test Team will track them
* When fixed, the defects will be retested by the Test Team, closed, or reopened depending on the outcome of test execution
* Regression testing will occur on new builds to ensure that functionality that was working is still working
* Update the test scripts with possible new tests or new steps to the existing test scripts as the software is modified

1. Create artifacts needed for release of the software, such as:

* Defect Log (including Software Anomalies report for issues still present in the software) – the defect log can be a report output from JIRA
* Test Evaluation Summary that provides overall statistics and results about the System Test level

## Performance and Load Test

Note: Additional information for this test type will be added/updated in a future Sprint iteration as needed.  
Performance and load testing will be performed by members of the MAP2 HPE Development and SQA Test teams as required by the MAP2 Performance Work Statement (PWS) document, dated 8/29/2014. The MAP2 program team will review all applicable VA Enterprise NFRs related to performance and load test requirements for the Rx Refill application. If it is determined that performance and/or load testing is required, performance and/or load test cases and scripts will be developed. The determination to execute a Performance and/or Load Test will be based on factors, such as the number of users to be using the new application, the types of services that will be executed within the environment, and the status of prior performance tests (i.e., have the functions been tested before). If it is determined that a Performance and Load Test is required, the following are high-level goals to be met for the Performance and Load Test:

* Validate that documented Critical Success Criteria around performance of the integrated system and infrastructure, which map to Critical Success Factors that are visible to and agreed with the customer, have been satisfied
* Assure the system will perform as required under expected usage levels
* Identify points of system degradation or bottlenecks
* Identify system capacity or limitations on specific components
* Reduce implementation risk
* Identify causes of poor performance to the business functions

A System Performance Impact Assessment will be completed based on the performance test results OR the fact that no performance impact is expected based on the SDD Addendum review.

## User Functionality Test

Note: Additional information for this test type will be added/updated in a future Sprint iteration as needed.  
After successful completion of the System Test level to include functional, non-functional, full regression, integration, end-to-end, performance and load testing (if applicable), the application code will be promoted to the MAE Integration test environment for further user acceptance testing. All test documents /testing package will be presented to the User Functionality Testing (UFT) team testers along with a kick off call to formalize the beginning of the UFT Test level. Daily status calls will also be coordinated to discuss issues encountered during testing and resolution of the issues will be documented. At the end of the UFT Test level, a UFT test approval will be completed by the UFT team.  
Testing at this level is typically performed by members of the business sponsor organization that have the knowledge of the functional requirements, but they can be performed by actual end users. The objective is to confirm that the product meets the business need or business acceptance criteria. It is the responsibility of the business sponsor to define the contents of the UFT and the methods for performing it. The MAP2 HPE SQA team will support this Test level as required.  
Since the Rx Refill application is developed using an Agile process, business owner representatives should have been involved in the development process, and it is more likely that the UFT will be less intense and a short duration.   
It is expected that user functionality testing will be done using the user stories developed and refined early in the development phase of the project along with the same test data used during earlier feature verification and system testing.

## Enterprise System Engineering Testing

Note: Information for the Enterprise System Engineering (ESE) testing section will be added/updated in future iterations as needed.

### V&V Test

The V&V Test will be used to validate the features or functional and non-functional testing completed by MAP2 HPE SQA Test team. Members of the MAP2 HPE SQA Test team will work with the MAP2 HPE Release Manager to ensure the application code and any required artifacts (i.e., Test Cases, Test Scripts, updated RTM, etc.) are ready for release to the V&V team.

### ESE Release Process

Members of the MAP2 HPE SQA team will coordinate with the MAP2 HPE Release Manager to ensure required documentation is submitted to the ESE Compliance body responsible for completing this type testing on the Rx Refill application.

## Initial Operating Capability Evaluation

Note: Information for the Initial Operating Capability (IOC) Evaluation section will be added/updated in future iterations.  
The Initial Operation Capability (IOC) Evaluation will be accomplished through limited release of applications to real users in the production environment. The Rx Refill application must pass all required compliance body reviews and the V&V Test in order to be released to the real end users. The definition of the required compliance bodies is dependent on the mobile application. For example, mobile applications that are "Very Low" systems do not have to be assessed by the Privacy compliance team. The MAP2 Release Management processes will be followed before releasing to production for IOC Evaluation.

# Testing Techniques

Note: Additional information for this section will be added/updated in future iterations as needed.

## Risk-based Testing

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. In an Agile process, it is typical to implement high-risk items in early sprints. The MAP2 Scrum teams (i.e., Developers) will perform their internal product component and component-integration testing on these early sprints and continue testing as the agile cycles continue. The MAP2 Scrum teams will identify high-risk areas and the MAP2 Scrum teams (i.e., HPE SQA Test Testers) will be involved early in the process to test these items prior to the final development Sprint.   
For the System Test, V&V Test, and Compliance Review Test, all functional, non-functional, and compliance requirements must be met. While risk based testing can be performed to identify high-risk issues first, there is not as much value in this approach as with the user story features testing done by the MAP2 Scrum teams. In the end, all requirements must be tested in some manner even if it is through inspection versus actual testing.

## Enterprise Testing

Note: Additional information for the Enterprise Testing section will be added in future iterations as needed.

### Security Testing

The MAP2 HPE Development and SQA Test team will develop tests to validate the security requirements and to ensure readiness for the independent testing performed by the EAS team. The EAS team perform Web Application Security Assessments (WASAs) that are an in-depth penetration test for common vulnerabilities, such as SQL Injection, Authorization Bypass and Cross-Site-Scripting (CSS). WASAs cannot be started until the EAS team receives a completed questionnaire with working test accounts and a full directory listing. The MAP2 HPE SQA Test team will ensure applicable application Uniform Resource Locators (URLs) and test accounts work before the questionnaire is submitted to the EAS team.  
In addition to the EAS evaluation, the Network Security Operations Center (NSOC) will scan the application and the code for vulnerabilities. The MAP2 HPE Development team needs to complete the NSOC Intake Questionnaire prior to submitting the application for the EAS evaluation. This penetration test is particularly important for those mobile applications that read and write data to VA systems.

### Privacy Testing

Note: Information for the Privacy Testing section will be added/updated in future iterations as needed.  
VHA Privacy and Application Security offices ensure that mobile applications adhere to Privacy regulations and statutes as well as VA policy. They use the Privacy and Security checklist to determine if data is stored, transmitted, or entered by the user, provider, or employee. They also determine if there is sensitive information like Private Health Information (PHI) or Personally Identifiable Information (PII) stored on the application. If it is, the Data Security branch will determine how the Rx Refill application protects the information.   
A Privacy checklist will be completed by the MAP2 HPE Development team. The MAP2 HPE SQA Testers will work with the MAP2 HPE Release Manager to ensure the Privacy checklist is completed for the Rx Refill application.

### Section 508 Compliance Testing

The Office of Information and Technology (OIT) Section 508 tests mobile content to verify that it complies with Section 508 standards including 1194.21 (Software and Platforms), 1194.22 Web, 1194.24 (Multimedia), and 1194.31 Functional Requirements. They work to facilitate access to mobile technologies for anyone with disabilities.  
The Section 508 Office will manually test VA platform-specific applications directly on the device using its built-in accessibility features and supported methods. For example, they test with VoiceOver on iOS and Talkback on Android.  
The Section 508 Office will perform some automated testing on web-based content, but they also perform manual tests to ensure the (mobile application) properly implements accessibility techniques on the devices browsers.  
The MAP2 HPE Development and SQA Test teams will perform 508 compliance testing on the Rx Refill mobile application prior to initiating the Section 508 Program Office to perform their independent testing.

### Multi-Divisional Testing

Note: Information for the Multi-Divisional Testing section will be added/updated in future iterations as needed.  
Multi-Divisional testing is required to ensure that all applications will operate in a multi-division or multi-site environment recognizing that an enterprise perspective while full supporting local health care delivery. If multi-divisional requirements are present in an application's BRD, the MAP2 HPE Development and SQA Test teams will verify and validate that the application under test complies with the multi-divisional requirements stated prior to initiating the applicable Compliance body to perform their independent testing.

## Test Types

Note: additional information for the Test Types section will be added/updated in future iterations as needed.  
Table 4 lists the following test types that will be performed under MAP2. While all of these test types do not list the MAP2 HPE Development or SQA Test teams, the MAP2 HPE Development Team and the corresponding MAP2 HPE SQA Test team must ensure that the mobile application is compliant in the particular area before moving the mobile application past the System Test level.

**Table 4: Test Types**

|  |  |
| --- | --- |
| **Test Types** | **Party Responsible** |
| Access Control Testing | MAP2 HPE Development and SQA Teams, VA V&V Team |
| Build Verification Testing | MAP2 HPE Development Team |
| Code Coverage | MAP2 HPE Development Team, VA Compliance Body |
| Compliance Testing | VA V&V Team and Compliance Bodies |
| Component Integration Testing | MAP2 HPE Development Team |
| Data and Database Integrity Testing | MAP2 HPE Development and SQA Teams, VA V&V Team |
| Documentation Testing | VA V&V Test Team |
| Error Analysis Testing | MAP2 HPE Development and SQA Teams, VA V&V Team |
| End-to-End Testing | MAP2 HPE Development and SQA Test Team, VA V&V Team |
| Functional Testing | MAP2 HPE Development and SQA Test Team, VA V&V Team |
| Information Assurance Testing | MAP2 HPE Development and SQA Test Team, VA V&V Team |
| Installation Testing | MAE Maintenance Team |
| Integration Testing | MAP2 HPE Development and SQA Teams, VA V&V Team |
| Integrated Performance and Load Testing | MAP2 HPE Development and SQA Teams |
| Privacy Testing | MAP2 HPE Development and SQA Teams, VA V&V Team and Compliance Bodies |
| Product Component Testing | MAP2 HPE Development Team |
| Regression Testing | MAP2 HPE Development and SQA Teams, VA V&V Team |
| Section 508 Compliance Testing | MAP2 HPE Development and SQA Test Team, VA V&V Team and Compliance Bodies |
| Security Testing | MAP2 HPE Development and SQA Test Team, VA V&V Team and Compliance Bodies |
| Smoke Testing | MAP2 HPE SQA Test Team, VA V&V Team |
| Usability Testing | VA V&V Team and Compliance Bodies |
| User Functionality Testing | VA Stakeholders |
| User Interface Testing | MAP2 HPE Development and SQA Test Team, VA V&V Team |

## Productivity and Support Tools

Note: Information for the Productivity and Support Tools section will be added in future iterations.  
Table 5 describes the tools that will be employed to support this Master Test Plan.

**Table 5: Tool Category or Types**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tool Category or Type** | **Tool Brand Name** | **Vendor or In-house** | **Version** |
| Test Management | MS Excel 2013 | Vendor - HPE | N/A |
| Defect Tracking | JIRA – Atlassian 6.2 | Vendor - HPE | N/A |
| Code Analyzer | HP Fortify | Vendor - HPE | N/A |
| Document Management | Confluence (wiki) – Atlassian 6.2 | Vendor - HPE | N/A |
| Performance and Load Testing | Apache JMeter – Open Source | Vendor - HPE | N/A |
| Configuration Management | Stash/Git – Atlassian 6.2 | Vendor - HPE | N/A |
| DBMS tools | Oracle & MongoDB | Vendor - HPE | N/A |
| Functional Test Automation | Selenium 2.5.0 | Vendor - HPE | N/A |
| Web Service Test Automation | SoapUI – Open Source | Vendor - HPE | N/A |
| Database Query Tools | Quest -TOAD, Oracle SQLPlus | Vendor - HPE | N/A |

# Test Criteria

## Process Reviews

The Master Test Plan undergoes two reviews:

* Peer Review – upon completion of the Master Test Plan
* MAP2 HPE Technical Writer Review
* Formal Review – after the HPE MAP2 Program and Project Manager approves the Master Test Plan

The Master Test Plan does serve as an input artifact used for Milestone 1 and Milestone 2 Reviews of the Development phase.  
For more information on the milestone reviews associated with testing, see the required artifacts and templates for PMAS as part of the VA's Mobile Application Development Life Cycle.

## Pass/Fail Criteria

A test will be considered as successful when it passes 100% of the test cases.  
Per the Agile Development Methodology (ADM) followed by the MAP2 Scrum teams, all defect(s) found for a given user story will be fixed and retested during the sprint the feature(s) were developed. If for some reason the defect(s) for a certain user story cannot be fixed and retested prior to Sprint completion, the defect(s) and applicable user story will be addressed in a subsequent sprint.  
For the System Test, pass criteria is defined in each specific test case; but in general, a test is considered as passed when the functionality meets the requirements as defined with no Level 1, Level 2, or Level 3 defects identified. If any test case within the overall test fails, then the whole test fails. It is typical that an application will still be released with some Level 3 and Level 4 defects, but it must be approved by the business owner and the release processes in order to be approved out to production. The open issues are provided as part of the release documentation.  
A test is considered failed when the functionality does not meet the requirements as defined.

## Suspension and Resumption Criteria

Suspension: Every effort will be made to continue testing when failures are found. However, some failures can impact product functionality significantly, and execution is no longer possible. Testing will be suspended if functionality is significantly impacted until a fix or workaround is provided.  
Resumption: In general, testing will be resumed from the point of suspension once a fix or workaround is provided. If the fix or workaround has implications outside the boundaries of the immediate test case procedure, it will be necessary to execute additional regression test procedures or test cases.  
The Project Manager, at the recommendation of the MAP2 HPE SQA Test team, has the authority to suspend testing. Testing will be suspended if:

* Test accounts or verifiable items are unavailable or become unstable
* A failure occurs 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
* The component being tested fails or a major component fails. A major component failure is one that one can reasonable assume will result in other test case failures. If there are so many defects that continuing testing is a waste of resources, testing will be suspended.
* Testing will resume when the component or code has been repaired, rebuilt and versioned, unit tested, gone through system test, and is installed into the test environment by the MAE maintenance team. If the cause of the suspension is due to either unstable or unavailable test environment, testing will resume when the test environment becomes stable and/or available.

## Acceptance Criteria

The following activities must be performed to complete the test and validation process:

* Updates to test documentation have been completed and are under configuration management control on the mobile application wiki page
* All required test types have been completed (i.e., Functional, 508, Security and Regression, etc.)
* Test cases have been executed according to the test plan, and any deviations are documented and approved
* All compliance bodies have approved the release or stated that their participation is not required for the Rx Refill mobile application
* All defects have been analyzed and changed to the appropriate state
* All closed defects have an assigned defect root cause

After completion of the above activities, the acceptance criteria for release of the application is:

1. For System Test, the Rx Refill application has no Level 1, Level 2, or Level 3 defects except as specifically accepted by the business owner.
2. All defects that are in the application are documented in the Test Evaluation Summary report as known defects and accepted by the business owner.
3. All MAP2 Release Management activities are carried out.
4. The business sponsor or owner and required compliance bodies have signed off on the release.

# Test Deliverables

Table 6 lists the test deliverables for the Rx Refill application.

**Table 6: Test Deliverables**

|  |  |
| --- | --- |
| **Test Deliverables** | **Responsible Party** |
| Master Test Plan (Iterative) | MAP2 HPE SQA Test Planner/Engineer |
| Master Test Plan Checklist | MAP2 HPE SQA Test Manager |
| Test Execution Risks | MAP2 HPE SQA Test Manager and Project Manager |
| Test Schedule | MAP2 HPE SQA Test Manager and Project Manager |
| Test Cases/Test Scripts (Iterative) | MAP2 HPE Developers and SQA Testers |
| Test Data | MAP2 HPE Developers and SQA Testers |
| Test Environment | MAP2 HPE SQA Test Manager |
| Test Evaluation Summary and Defect Log (Iterative) | MAP2 HPE SQA Test Planner/Engineer for Sprint Iterations and System Test  VA V&V Team for V&V Tests and Compliance Reviews |
| Requirements Traceability Matrix | MAP2 HPE SQA Test Planner/Engineer for Sprint Iterations and System Test  VA V&V Team for V&V Tests and Compliance Reviews |

# Test Schedule

Table 7 lists the testing milestones for the Rx Refill application.

**Table 7: Testing Milestones**

|  |  |
| --- | --- |
| **Testing Milestones** | **Responsible Party** |
| Deliver Initial MTP and MTP Checklist for all MAP2 Mobile Applications | MAP2 HPE Test Manager |
| Complete Sprint 1 User Story/Unit Test/SQA Testing | MAP2 HPE Developers and SQA Testers |
| Complete Sprint 2 User Story/Unit Test/SQA Sprint Testing | MAP2 HPE Developers and SQA Testers |
| Complete Sprint 3 User Story/Unit Test/SQA Sprint Testing | MAP2 HPE Developers and SQA Testers |
| Complete Sprint 4 User Story/Unit Test/SQA Sprint Testing | MAP2 HPE Developers and SQA Testers |
| Complete Sprint 5 User Story/Unit Test/SQA Sprint Testing | MAP2 HPE Developers and SQA Testers |
| Complete Sprint 6 User Story/Unit Test/SQA Sprint Testing | MAP2 HPE Developers and SQA Testers |
| Complete System Testing | MAP2 HPE SQA Test Team |
| Complete Compliance Reviews, V&V Testing, IOC | VA Compliance Bodies and V&V Test Team |

# Test Environments

A test environment is an environment containing hardware, instrumentation, simulators, software tools, and other support elements needed to conduct a test. Simulators in a service-oriented architecture, such as used in VA, include "mock services," which simulate the behavior of VA services. The MAE contains mock services in some areas to control the interfaces and data retrieved and simulate test scenarios.  
The test environments for the Rx Refill application will be the MAE Development and Development-Test environments.

## Test Environment Configurations

The MAE includes the following environments and tools that can be utilized by the MAP2 Program team:  
**Development environment** – This includes the following services and tools:

* **Git/Stash** – Code repositories are provided for the mobile application as well as access to the Mobile Adapter (Health Adapter) code repository to provide the development team access to the common services provided
* **Jenkins** – For building the software into deployable units
* **Fortify** – For scanning the software
* **JIRA** – For executing the agile processes, documenting user stories, creating agile boards, documenting software issues, and so on. Compliance issues are included in the JIRA project so the development team can track completion of these requirements and attach the appropriate documents.
* **Confluence** – For documenting the mobile application in a wiki format. The wiki can be used to deliver the documents needed by MAP and the V&V team.
* **Development integration servers** – To include mock services for standard VA enterprise services to be called by the Mobile Adapter services and/or directly from mobile applications

**MAE Development** – Developers use this environment for internal component integration, not external integration. It allows the users to test deployment of multiple developer's code from the code repository. Features of this environment are:

* Mock Services for external systems, such as Identity and Access Management (IAM), Master Veteran Index (MVI), Administrative Data Repository (ADR), VistA (or dev VistA), Medical Domain Web Services (MDWS), Corporate Data Warehouse (CDW), etc.
* All test data and fake patients/users

**MAE Development-Test** – MAP2 HPE SQA Testers will perform system testing in this environment.  
Features of this environment are:

* Real Test Services for external systems, such as IAM, MVI, ADR, VistA (or dev VistA), MDWS, CDW, etc.
* Mock Services used where development integrations are not allowed
* All test data and fake patients/users

**MAE Integration** – This environment will be used for system testing and to date for V&V and Compliance testing. When more resources are obtained, this will be a system testing environment.   
Features of this environment are:

* Real Test Services for external systems, such as IAM, MVI, ADR, VistA (or dev VistA), MDWS, CDW, etc.
* Mock Services used where needed such as CDW
* All test data and fake patients/users

Access to these MAE environments is obtained through the Web and Mobile Solution request process. The MAP2 HPE Development and SQA Test teams can only have access to the MAE Development environments. The MAP2 CM manager must have access to all MAE development environments for auditing purposes. The MAE Maintenance Team maintains all of the MAE environments, and this team should not include any developers from the MAP2 Program team.

## Base System Hardware

Note: Additional information for the Base System Hardware section will be added/updated in future iterations as needed.  
Table 8 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 will simulate the production environment as closely as possible, scaling down the concurrent access and database size, and so forth, if and where appropriate. The System Hardware Resources table will be tailored as required.

**Table 8: System Hardware Resources**

|  |  |  |
| --- | --- | --- |
| **Resource** | **Quantity** | **Name and Type** |
| Test Development PCs | 1 | HPE-issued Windows 7 laptop |
| Mobile Device | 1 | GFE-issued Apple iPad |

## Base Software Elements in the Test Environments

Additional information for the Base Software Elements in the Test Environments section will be added/updated in future iterations as needed.  
Table 9 describes the base software elements that are required in the test environment for this Master Test Plan.

**Table 9: Software Elements**

|  |  |  |
| --- | --- | --- |
| **Software Element Name** | **Version** | **Type and Other Notes** |
| Windows | 7.0 | Desktop Operating System |
| IOS | 7+ | Apple Operating System |
| Internet Explorer | 9 and11 | Laptop Internet Browser |
| Safari | 7+ | Mobile Internet Browser |
| Android Studio | 1.5.1 | Emulator |
| Chrome | 40+ | Laptop Internet Browser |
| Firefox | 36+ | Laptop Internet Browser |

## Staffing and Training Needs

Note: Additional information for the Staffing and Training Needs section will be added/updated in future iterations as needed.  
Table 10 describes the personnel resources needed to plan, prepare, and execute this Master Test Plan.

**Table 10: Staffing Resources**

|  |  |  |  |
| --- | --- | --- | --- |
| **Testing Task** | **Quantity of Personnel Needed** | **VA MA Development Phase** | **Duration/ Days** |
| Create/Update Master Test Plan – Application-Level | 1 | Planning/Development | TBD |
| Establish the Test Environment | 1 | Planning/Development | TBD |
| Create/Update Test Cases and Scripts | 3 | Development | TBD |
| Perform Feature Verification/System Tests | 3 | Development | TBD |
| Test Management Reporting | 1 | Development | TBD |
| Coordinate with V&V Test team and Compliance Bodies | 3 | Development/Compliance Review | TBD |

Table 11 lists the personnel that require training. This document does not apply to any VA User training needed.

**Table 11: Training Needs**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Training Need** | **Training Option** | **Estimated Training Hours** |
| MAP2 HPE Developers and SQA Testers | AMP Training | Online | 2 |

# Risks and Constraints

Note: Additional information for the Risk and Constraints section will be added/updated in future iterations as needed.  
Risks and constraints will be tracked in JIRA at the MAP2 mobile application-level. JIRA will be updated as risks and constraints are identified.  
Again, risks are tracked in JIRA both at the mobile application-level and the MAP program-level. Risks that affect the completeness or effectiveness are the following:

* Lack of Environments – need to ensure VA will be configuring the MAE Integration and Pre-Production environment to match the configuration of the MAE Development-Test environment. This is to fully support integrated system testing, V&V testing, and performance testing. If the configuration is not the same, the overall test approach is compromised.
* If an integrated MAE Development-Test (i.e., CDW, HDR, Vista, etc.) environment is not available, then there is a risk that testing will not be adequate for VA V&V and production release
* Assuming performance and load testing will be performed in the MAE Development-Test environment, and this environment is not a close match to the production environment, then then there is a higher risk that the migrated application will not perform as expected in the Production environment
* In the case of read-only applications, the pre-Production environment can be integrated with other pre-Production/Production environments for UFT testing, but obviously a pre-Production environment cannot write to a Production environment. This needs to be managed closely on an application by application basis and a method of testing constructed such that it is clear that adequate testing has been performed.

# Test Metrics

Metrics are a system of parameters or methods for quantitative and periodic assessment of a process that is to be measured. Table 12 describes the test metrics to be captured and reported on for the Rx Refill mobile application.

**Table 12: Test Metrics**

|  |  |  |
| --- | --- | --- |
| **Report Name** | **Report Type** | **Report Description** |
| Projected vs. Actual Test Effort | Project Management | The Estimated vs. Actual Test Effort report provides management with a view of how well the project met its agreed upon delivery timeline determined at project initiation. The data used to construct this indicator is as follows:   * Estimated Hours: Estimated hourly effort associated with completing all activities required to meet the exit criteria for a given test phase * Actual Hours: Hourly effort associated with completing all activities required to meet the exit criteria for a given test phase |
| Actual Test Effort by Level | Project Management | The Actual Test Effort by level report provides a breakdown of the testing effort by each level for each application in a MA release. |
| Requirement Test Case Coverage | Coverage | Requirements Test Case Coverage report provides a method to measure the percentage of coverage through test procedures. The data used to construct this indicator is as follows:   * Baseline Requirements: The number of requirements approved by the team and client to be addressed in the MA release, iteration, or build under development * Requirements Mapped to Test Scripts: The number of baseline requirements that have been mapped to test scripts for the specified release |
| Overall Test Case/Script Execution Status | Progression | The Overall Test Procedure Execution Status Report details the progression and extent of the test effort at a given point and time and the percentage of test cases executed in addition to:   * Total Test Scripts – The number of test scripts scheduled to be run in the current test cycle. The test cycle will be denoted in the slide * Scripts to be Executed – The number of scripts yet to be run * Currently Passed – the number of scripts for the current cycle that have passed * First Time Passed – The number of "Currently Passed" scripts that passed the first time they were executed |
| Daily Test Execution Status | Progression | The Daily Test Execution Status reports details the daily progression of the testing effort versus the planned progression by reflecting how many scripts have passed and failed on a daily basis. The report captures the following metrics:   * Test Scripts in Release: Number of scripts that provide coverage for the release * Test Scripts Passed Cumulative Planned: Number of scripts the test team expects to have passed at a given point in time * Test Scripts Passed Cumulative Actual: Number of scripts that have actually passed at a given point in time * Test Scripts Passed Daily: Number of scripts run in a given day that have successfully passed * Test Scripts Failed Daily: Number of scripts run in a given day that failed |
| Open Defects by Severity | Progression | Open Defects by Severity Report captures the ratio of reported defects by the severity level and the:   * Total number of defects identified * Number of defects associated to test case/test script and user stories/requirements * Percentage of defects listed by cause and severity |

# Attachment A - Approval Signatures

The Master Test Plan documents the program's overall approach to testing and includes:

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

Review Date:   
Constance Murphy, VA MAP2 Program Manager   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Signed:                                                                                                                            Date:

Luz A. Cuff, VA MAP2 Contracting Officer's Representative   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Signed:                                                                                                                           Date:

VA MAP2 Business Owner   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Signed:                                                                                                                           Date:

# Attachment B – Defect Level Definitions

**Level 1 Critical** - The defect results in the failure of the complete software system, of a subsystem, or of a software unit (program or module) within the system.

* Any defect that compromises patient safety or system security. Examples of system security defects include breach of confidentiality requirements of the Privacy Act, the Health Insurance Portability and Accountability Act (HIPAA), or Federal Tax Information guidelines.
* Loss of system functionality critical to user operations with no suitable workaround, i.e., there is no way to achieve the expected results using the application
* System crash or hang that prevents further testing or operation of the complete application or a section of the application
* Any defect that causes corruption of data from a result of the system (as opposed to user error)
* Any defect in which inappropriate transmissions are consistently generated or appropriate transmissions of HL7 messages fail to be generated
* Loss of functionality resulting in erroneous eligibility/enrollment determinations or communications not being sent

**Level 2 High**- The defect results in the failure of the complete software system, of a subsystem, or of a software unit (program or module) within the system. There is no way to make the failed component(s) function. However, there are acceptable processing alternatives which will yield the desired result.

* A major defect in the functionality that does not result in corruption of data
* A major defect in the functionality resulting in a failure of all or part of the application, where:
* The expected results can temporarily be achieved by alternate means. The customer indicates the work around is acceptable for the short term.
* Any defect that does not conform to Section 508 standards
* Any defect that results in inaccurate or missing requirements
* Any defect that results in invalid authentication or authentication of an invalid end user

**Level 3 Medium** - The defect does not result in a failure, but causes the system to produce incorrect, incomplete, or inconsistent results, or the defect impairs the systems usability.

* Minor functionality is not working as intended and a workaround exists but is not suitable for long term use
* The inability of a valid user to access the system consistent with granted privileges
* Typographical or grammatical errors in the application, including installation guides, user guides, training manuals, and design documents
* Any defect producing cryptic, incorrect, or inappropriate error messages
* Any defect that results from the use of non-standard data terminology in the application or documentation, as defined by the Department of Veterans Affairs
* Cosmetic issues that are important to the integrity of the product, but do not result in data entry and or data quality problems

**Level 4 Low** - The defect does not cause a failure, does not impair usability, and the desired processing results are easily obtained by working around the defect.

* Minor loss of, or defect in the functionality where a long term use exists
* Low-level cosmetic issues

# Attachment C – Test Case/Script – Coverage

Attach test cases/scripts to be used for the Rx Refill application.  
**Test Case:**  
**Description**  
**TC Pre Conditions:**

|  |  |  |
| --- | --- | --- |
| **Pre-Conditions** | **Post-Conditions** | **Acceptance Criteria** |
|  |  |  |
| The user/tester should have access to all needed areas like the store to download and install the app.  The user/tester should have the app installed on the device.  The user/tester should be familiar with Mobile apps controls, such as slides and spinners. |  |  |

**Test Script**  
**Test Script Description:**  
**Test Procedure**

|  |  |  |
| --- | --- | --- |
| **Pre-Conditions** | **Post-Conditions** | **Acceptance Criteria** |
|  |  |  |
|  |  |  |

**Auto Implementation:**   
**Manual Implementation:**   
**Test Inputs:**   
**Test Design**

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Type** | **Note** | **Description** |
| 1 | Step |  |  |
| 2 | VP | . |  |
| 3 | Step |  |  |
| 4 | VP |  |  |

# Attachment D – 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 do not 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). |
| 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 program's 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 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 HeV-VistA operating procedures and loads the latest .jar version onto a live copy of VistA and HeV-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. |
| 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 required 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. |