

# **Department of Veterans Affairs**

## **VistA Scheduling Enhancements (VSE)**

### **Master Test Plan**



**September 2014**

**Version 2.0**

## Revision History

Date	Version	Description	Author
9/26/2014	2.0	Added E1, E2, and E3 in Section 2 and Section 3 TW review performed	[REDACTED]
8/29/2014	1.0	Initial version for submission	[REDACTED]

## Table of Contents

<b>1. Introduction .....</b>	<b>1</b>
1.1. Purpose .....	1
1.2. Test Objectives .....	1
1.3. Roles and Responsibilities .....	2
1.4. Processes and References .....	3
<b>2. Items to be Tested .....</b>	<b>4</b>
2.1. Overview of Test Inclusions .....	4
2.2. Overview of Test Exclusions .....	6
<b>3. Test Approach.....</b>	<b>7</b>
3.1. Product Component Test (Unit Test) .....	8
3.2. Conduct Peer Review of Product Component (Secondary Developer Review) .....	9
3.3. Component Integration .....	9
3.4. System Test.....	10
3.4.1. HPES Testing Reports.....	11
3.5. User Acceptance Test .....	11
3.6. Enterprise System Engineering Testing .....	11
3.7. Initial Operating Capability Evaluation .....	12
<b>4. Testing Techniques .....</b>	<b>14</b>
4.1. Risk Based Testing.....	14
4.2. Enterprise Testing .....	14
4.2.1. Security Testing.....	14
4.2.2. Privacy Testing .....	14
4.2.3. Section 508 Compliance Testing.....	14
4.2.4. Multi-Divisional Testing.....	14
4.3. Test Types .....	15
4.4. Productivity and Support Tools.....	15
<b>5. Test Criteria.....</b>	<b>16</b>
5.1. Process Reviews.....	16
5.2. Pass/Fail Criteria.....	16
5.3. Suspension and Resumption Criteria .....	17
5.4. Acceptance Criteria .....	17
<b>6. Test Deliverables .....</b>	<b>18</b>
<b>7. Test Schedule.....</b>	<b>18</b>
<b>8. Test Environments.....</b>	<b>19</b>
8.1. Test Environment Configurations .....	20

8.2. Base System Hardware .....	20
8.3. Base Software Elements in the Test Environments.....	20
9. Staffing and Training Needs.....	20
10. Risks and Constraints .....	20
11. Test Metrics.....	21
Attachment A - Approval Signatures.....	22
A. Test Type Definitions.....	23
A.1. Test Type Definitions.....	24
B. Acronyms .....	30

## List of Tables

Table 1-1: Roles and Descriptions.....	2
Table 2-1: Test Inclusions.....	4
Table 3-1: Test Environments.....	8
Table 3-2: IOC Test Sites .....	12
Table 4-1: Test Types .....	15
Table 4-2: Productivity and Support Tools.....	15
Table 6-1: Test Deliverables .....	18

# **1. Introduction**

## **1.1. Purpose**

The purpose of this Master Test Plan (MTP) for the VistA Scheduling Enhancements (VSE) project is to provide the planning and control of the test effort for development testing provided by Hewlett-Packard Enterprise Services (HPES). This test plan defines the test approach that will be employed to test the application's software and evaluate the testing results.

## **1.2. Test Objectives**

The object of the VSE MTP is to provide a detailed description of testing processes, defect correction and reporting, verification that the code meets the business and functional requirements, and determination of areas of risk.

This MTP supports the following objectives:

- Identifies the items that should be targeted by the tests
- Identifies the motivation for and strategies behind the test areas that will be covered
- Outlines and documents testing approaches
- Identifies testing resources and needs
- Lists the testing deliverables

## 1.3. Roles and Responsibilities

The following table lists the Key roles and their responsibilities for this MTP

**Table 1-1: Roles and Descriptions**

Role	Description
Program Manager	Person that has overall responsibility for the successful planning and execution of a project.
Project Manager	Person responsible for project management.
Business Analyst	Person responsible for requirements analysis.
Scrum Master	Person responsible for planning agile activities.
Development Team	Persons that build or construct the product/product component.
Test Team/Testers	<p>Person responsible for ensuring full execution of the test process to include the following:</p> <ul style="list-style-type: none"><li>• Responsible for creating the MTP in collaboration with the project team members.</li><li>• Verification of functional requirements.</li><li>• Validation of business requirements.</li><li>• Coordination of activities related to all aspects of testing, based on the approved application-specific test plan and schedule.</li><li>• Creation and execution of test cases.</li><li>• Confirmation of test environment adequately supporting the planned test activities.</li><li>• Notation of defects found during the Component Integration Testing and System Testing.</li></ul>
Configuration Management Team	Persons that establish, maintain, and control test environments.

## 1.4. Processes and References

The VSE project follows agile software development and project management best practices. The processes that guide the implementation of the VSE MTP are:

- Test Preparation
- Product Build
- Independent Test and Evaluation

The references that support the implementation of this MTP are:

- [ProPath](#)
- [Section 508 Office Web Page](#)

The following documents are kept under VSE SharePoint

- [VistA Scheduling Enhancement Risk Management Plan](#)
- [VistA Scheduling Enhancement Requirements Specifications Document](#)
- [VistA Scheduling Enhancement Software Design Document](#)
- [VistA Scheduling Enhancement Requirements Traceability Matrix](#)

### Additional Websites:

- [Department of Veterans Affairs \(VA\) Software Document Library \(VDL\) User Documentation](#)
- VSE Business Justification Package – Enhancement 3 and 4
- Aggregate View of Clinic Profile Scheduling Grid\_BJP
- A Single Queue of Request List BJP
- [VA OI&T Master Glossary](#)
- User Acceptance Test (UAT) Cases will be placed in SharePoint

## 2. Items to be Tested

The following functional requirements documented within the VistA Scheduling Enhancement Performance Work Statement (PWS) - Section 5: Specific Tasks and Deliverables will be tested within the VSE MTP.

- Enhancement 1: An Aggregated View of Vista Clinic Profile Scheduling Grids for VistA Scheduling
- Enhancement 2: A Single Queue for Appointment Requests
- Enhancement 3: A Resource Management Dashboard
- Enhancement 4: High Priority/Urgent Enhancements of VistA

*Note: Please refer to the Scheduling Requirements Traceability Matrix (RTM), which provides traceability from the functional requirements to business requirements.*

### 2.1. Overview of Test Inclusions

The VSE Test Plan will test the following system features

**Table 2-1: Test Inclusions**

Enhancement #	Description
Enhancement 1(a)	Able to query clinic profiles using multiple criteria, such as: Appointment Type, Provider, and Desired Date.
Enhancement 1(b)	Display an aggregated view of clinic profile scheduling grids based on the query results.
Enhancement 1(c)	Sort specific criteria from the query results within the aggregated view by ascending and descending order.
Enhancement 1(d)	Filter the query results within the aggregated view using specific criteria, such as: Day of the Week, and Time of the Day.
Enhancement 1(e)	Navigate between the clinic profile views and the aggregated view of clinic profile scheduling grids.
Enhancement 1(f)	Open the specific clinic profile associated with the appointment opportunity displayed in the aggregated view.
Enhancement 1(g)	Display resource availability graphically in the aggregated view with a legend reflecting status, such as: Blocked, Available, or Scheduled.
Enhancement 2(a)	Generate a single queue of open request lists.
Enhancement 2(b)	Sort a single queue by ascending or descending order using specific criteria such as: Desired Date, Request Type, and/or Patient.
Enhancement 2(c)	Display and output single queue of patients that meet the sort criteria.
Enhancement 2(d)	Save Output from the results of a sort in a VA acceptable format such as: Microsoft (MS) Excel, PDF, or with option to print.
Enhancement 2(e)	Print/save screen prints and open lists in various points within utility.



<b>Enhancement #</b>	<b>Description</b>
Enhancement 2(f)	Generate an audit report on all single queue activity that can be sorted and filtered by parameters such as scheduler name, total appointments scheduled, total appointments pending action, and total new requests added.
Enhancement 3(a)	TOTAL SUPPLY (HOURS) – Provide a report that total hours (capacity) in the ACCESS BLOCK file for specified resource(s) in the specified time frame.
Enhancement 3(b)	TOTAL DEMAND (HOURS) – Provide a report that totals of all appointment lengths in the APPOINTMENT file for specified resource(s) in the specified time frame.
Enhancement 3(c)	RESOURCE SCHEDULE UTILITIZATION (HOURS) – Calculate and report TOTAL DEMAND (HOURS) divided by TOTAL SUPPLY (HOURS).
Enhancement 3(d)	CALCULATED ACTIVITY (HOURS) – Provide a report with Calculated Activity (hours) that reflects the number of appointment hours checked out (used) in face to face patient encounters for a specified time frame.
Enhancement 3(e)	CALCULATED UTILIZATION (HOURS) – Provide a report that calculates the percentage of appointment hours checked out (used) for a specified resource for a specified time frame. Numerator: CALCULATED ACTIVITY (HOURS). DEMONINATOR: TOTAL SUPPLY (HOURS).
Enhancement 4(a)	Appointment Management – When scheduling an appointment, VistA Scheduling currently prompts the clerk to pick “next available date.” The to-be process shall prompt the clerk to enter the patient’s desire date and only display “next available” option if the clerk selects that option.
Enhancement 4(b)	Clinic Set Up Redesign <ol style="list-style-type: none"> <li>1. Add clinic location and default provider to the Scheduling Letters.</li> <li>2. Expand the Clinic Profile Name from 30 characters to 45 characters</li> <li>3. Increase Clinic Name Field to 45 characters on the Un-Reviewed Clinic Report</li> </ol>
Enhancement 4(c)	Clinic Inactivation Date Prompt – When setting a Clinic to Inactive, create a reminder prompt to remove the inactive clinic from the active database.
Enhancement 4(d)	Auto-Generate Appointment Letter – In the Appointment Manager Package, create an option to allow the user to auto-generate and print a letter when making an appointment for the patient within the Appointment Manager Package.

Enhancement #	Description
Enhancement 4(e)	Ensure correct entry of desire date when creating appointments in a series – When creating a series of appointments, currently the desire date entered is assigned to all subsequent appointments. In the to-be solution, the desire date should only populate the first appointment in the series and the all subsequent appointments should be adjusted accordingly.
Enhancement 4(f)	Telephone Number Extension Field – Create a new 26 character alpha-numeric field in the Clinic Profile entitled Clinic Extension.
Enhancement 4(g)	Add Provider Name column to the Scheduling Letter.

## 2.2. Overview of Test Exclusions

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

- Non-functional requirements.
- Testing of functionality within the VistA Scheduling application that does not have Enhancement 1, 2, 3, and 4 impacts.

### 3. Test Approach

- HPES testers will create a MTP – One plan will be created for Enhancements 1, 2, 3, and 4. Unique sections will be provided for each enhancement as needed.
- HPES testers will create UAT Test Cases to assist testers with testing in controlled Bay Pines test environments.
- Software Quality Assurance (SQA) Review Checklist – The SQA Review Checklist will be used as a reference for all sprints prior to the final. The final SQA Review Checklist will be executed with the last sprint for each enhancement and will be submitted to VA after each increment is complete.
- HPES testers will create detailed Component Integration/System Test (CI/ST) Test Cases/Scripts using MS Excel to validate and verify the new functionality.
- The Requirements Traceability Matrix will be updated to map the UAT and CI/ST test scripts to the requirements/user stories and Software Design Document
- The HPES project team will provide support for UAT and Initial Operating Capability (IOC) testing. HPES will not manage the testing at the test sites, but will provide testing support and retest any defects from IOC. That work will likely be outside of any formal Increments.
- HPES Software Testers will conduct a Test Readiness Review (TRR) prior to UAT.

**Table 3-1: Test Environments**

Enhancement #	Environment Name	Type of Testing	Description
Enhancement 1,2,3, & 4		CI/ST	CI/ST environment will be used for starting the SQA Checklist, conducting a basic Smoke Test and execution of test cases/scripts that encompass main application functionality.
Enhancement 1,2,3,& 4		UAT	UAT Testing will be performed on this environment, will be used for execution of UAT test cases.
Enhancement 4		Primary Development	This environment is used for Primary development and Product Component Testing will be conducted by the HPES Project Team developers
Enhancement 4		Secondary Development	This environment is used for Secondary Developer peer review.

### **3.1. Product Component Test (Unit Test)**

Product Component Testing will be conducted by the HPES Project Team developers within the Albany environment, TBD. The Developer performs Product Component Testing (Unit Testing), which includes the internal technical and functional testing of a module/component of code, and verifies that the requirements defined in the System Design Document (SDD) have been successfully applied to the module/component under test. Steps include:

- Analyze requirements to understand the application functionality and dependencies
- Identify all the routines affected by the module or object
- Specify all the routines that are called from various locations
- Execute tests on prioritized options
- Execute tests with different combinations of options and data. For example, 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

### **3.2. Conduct Peer Review of Product Component (Secondary Developer Review)**

The HPES Developer conducts the Product Component Peer Review in accordance with the ProPath Reviews Guide (appropriate sections pertaining to Peer Reviews) performing the following general steps:

- Distribute the Peer Review Materials
- Review the Peer Review Materials
- Distribute the Consolidated Peer Review Findings
- Record the Finding Resolutions
- Implement the Finding Resolutions

The goal of the peer review of the Product Component is to resolve any questions the project team may have and to ensure the quality of the deliverable.

### **3.3. Component Integration**

The HPES Configuration Manager installs the Product Component and Tester performs the Component Integration (CI) Test within the Albany environment, CHEYL134. CI/ST will be conducted for each sprint. UAT will be conducted when all sprints have completed CI/ST.

Component Integration Testing – Product CI is performed to expose defects in the interfaces and interaction between integrated components. CI will verify the patch installation and the testing environment. The installation will be installed into the testing environment ensuring the installation process does complete and does not produce any errors. This will be followed by a smoke test which will verify stability of the environment.

The Tester will log defects using MS Excel in application-specific databases as defects are encountered.

At start of CI, the Tester will initialize the SQA Review Checklist and complete the following sections:

- Pre Installation (All Steps)
- Installation (All Steps)
- Post Installation (Steps 1-13)

### 3.4. System Test

The HPES Tester performs System Test, employing a variety of test types (i.e., compliance, regression, access control, interoperability, etc.). System Tests exercise all parts of an integrated system including interfaces to external systems within the Bay Pines test environment CHEYL134.

1. The Tester installs the build into the Bay Pines Test Environment.
2. The Tester will repeat the following sections from the SQA Checklist:
  - Installation (Steps 1-22)
  - Post Installation (Step 1)
3. Any defects that are encountered against the specific application database that is being tested are logged using MS Excel

When System Test is complete for the final sprint:

1. A new version of the SQA checklist is created and executed for every new build that comes out of CI/ST.
2. Defects will be logged and managed using MS Excel.
3. Defects are written and reported when detected.
4. System testing completed.
5. End of testing reports are generated and distributed along with the completed SQA checklist and all deliverables defined within the Performance Work Statements (PWS): Test Execution Log, Test Defect Log, and Test Evaluation Summary at the end of the increment.
6. Remediated software will be delivered via FORUM.

### 3.4.1. HPES Testing Reports

The Tester will generate Testing Results reports using MS Excel and distribute the reports to the HPES Project Manager (PM) at the end of each testing cycle. The frequency of the reports can be revised by HPES PMs at any time.

- **Test Execution Log** – This report will include the number of test scripts that have been completed based on the Test Execution Plan for the week and should include the following information:
  - The number of planned test scripts
  - The number of planned passed test scripts
  - The number of planned failed Test scripts
  - The percentage of overall test scripts executed
  - The number of defects written for the week
- **Test Defect Log** – This report will be an ongoing report that contains a list of defects created during testing as required by the PM. This report will be generated using MS Excel.
- **Test Evaluation Summary** – The purpose of the Test Evaluation Summary is to present a summary analysis of the key test results and key measures of test for review and assessment by designated stakeholders.

### 3.5. User Acceptance Test

The Project Manager ensures the UAT is conducted. UAT is a formal test conducted by the end-users to determine whether a system satisfies its acceptance criteria and enables the customer to determine whether to accept the system. The purpose of the UAT is to (1) exercise the functionality of the application using test data in a controlled test environment in order to validate functionality and (2) evaluate the usability of a component or system. Additionally, during User Functionality Testing, Enterprise Shared Service functionality, such as Identity and Access Management, are tested. UAT will be conducted when all sprints have completed CI/ST.

### 3.6. Enterprise System Engineering Testing

ESE testing need is known .If it is to be done it will be done by ESE organization. The HPES Project Team will support ESE testing.

### 3.7. Initial Operating Capability Evaluation

The IOC Implementation Manager coordinates the performance of the IOC evaluation. IOC evaluation (formerly known as field testing) occurs when a product/system that has been modified/enhanced is placed into a limited number of production (live) environments, in order to evaluate the new features and functionality of the product/system and to ascertain if the features and functionality perform as expected and do not adversely affect the existing functionality of the product/system.

Activities include:

- Distribute the product and product documentation to the Evaluation Sites (see More Info)
- Facilitate the timely installations at the Evaluation Sites
- Conduct formal or bi-weekly Evaluation Site calls
- Track defects identified during Initial Operating Capability Evaluation
- Address issues and questions identified during evaluation
- Obtain Site Concurrence Statements

**Table 3-2: IOC Test Sites**

State	Station ID	VISN #	VHA Facility Name	Integrated Sites	Facility Size	FY11 Complexity Level	SMALL Facilities (0 - 80,000); MEDIUM Facilities (80,001 – 150,000); LARGE Facilities (Greater than 150,001)	Date of Update
DC	688	5	Washington DC VA Medical Center		Large	1b	376201	6/6/2014
UT	660	19	VA Salt Lake City Health Care System		Large	1b	254443	6/7/2014
NC	637	6	Asheville VA Medical Center		Large	1c	166016	6/28/2014



State	Station ID	VISN #	VHA Facility Name	Integrated Sites	Facility Size	FY11 Complexity Level	SMALL Facilities (0 - 80,000); MEDIUM Facilities (80,001 – 150,000); LARGE Facilities (Greater than 150,001)	Date of Update
NY	620	3	VA Hudson Valley Health Care System, NY (Montrose, Castle Point)		Large	3	159746	6/29/2014
NY	630	3	VA New York Harbor Health Care System (New York, Brooklyn, Manhattan)	Integrated	Large	1a	604266	5/17/2014
OH	541	10	Louis Stokes VA Medical Center (Cleveland)		Large	1a	589149	6/22/2014
TN	621	9	James H. Quillen VA Medical Center (Mountain Home)		Large	1c	168,978	05/17/2014

## **4. Testing Techniques**

### **4.1. Risk Based Testing**

The VSE project will test identified risks throughout each testing cycle.

### **4.2. Enterprise Testing**

#### **4.2.1. Security Testing**

The VSE project adheres to existing standard VistA security specifications; therefore, the need to perform Security Testing is not applicable.

Enhancement 1 and 2 will have appropriate testing for proper user authentication to ensure appropriate measure is applied related to access limitations. Any other security testing needs that are not part of VistA will be evaluated for Enhancements 1, 2, and 3 and will be incorporated as necessary at a future date.

#### **4.2.2. Privacy Testing**

The VSE project adheres to existing standard VistA privacy specifications; therefore, the need to perform Privacy Testing is not applicable.

The portions of Enhancements 1, 2, and 3 that are not part of VistA will be evaluated for any unique Privacy testing needs and will be incorporated as necessary at a future date.

#### **4.2.3. Section 508 Compliance Testing**

Enhancement 1 and 2 will be evaluated for 508 Compliance and any other testing plans for this have not been defined at this point.

#### **4.2.4. Multi-Divisional Testing**

The VSE project will not replace or modify the existing VistA or HealthVet Application Architecture; therefore, the need to perform multi-divisional tests is not applicable.

### 4.3. Test Types

The table below specifies the test types to be performed and the party responsible for performing the test.

*Note: Please refer to [Appendix A](#) for test definitions.*

**Table 4-1: Test Types**

Types of Test	Party Responsible for Plan	Party Responsible for Execution
Perform Product Component Test	HPES Project Team within the HPES VSE Dev Environment TBD	HPES Project Team
Conduct Peer Review of Product Component	HPES Project Team within the HPES VSE Dev Environment TBD	HPES Project Team
Perform CI/ST	HPES Project Team within the HPES VSE Test Environment CHEYL134	HPES Project Team
Perform UAT	VA Testing Manager, HPES Project Team (Support), VA SQA Team within the HPES VSE environment CHEYL135	VA Testing Manager, HPES Project Team (Support), VA SQA Team, Subject Matter Experts (SMEs)
Conduct Enterprise System Engineering Testing	ESE Project Team	ESE project team, HPES Project (Support) Team
Conduct Initial Operating Capability Testing Activities	VA Project Manager	Field Testers, HPES Project (Support) Team
Conduct Final Review and Acceptance	VA Project Manager	VA Project Manager

### 4.4. Productivity and Support Tools

The table below describes the tools that will be used by the HPES Project team to support the VSE Test Plan.

**Table 4-2: Productivity and Support Tools**

Tool Category or Type	Tool Brand Name	Version
Test Management	SharePoint®	Refer to VSE CM Plan
Defect Tracking	MS Excel	Refer to VSE CM Plan
Project Management	MS Project	Refer to VSE CM Plan
Configuration Management	Rational Team concert	Refer to VSE CM Plan
Requirements Management	SharePoint®	Refer to VSE CM Plan

## 5. Test Criteria

### 5.1. Process Reviews

The following review will be conducted on the VistA Scheduling Enhancement Test Plan:

- Internal Peer Review – Conducted by the HPES Project Team upon completion.
- Test Preparation Formal Review – The VSE Formal Review with participants from the Veterans Health Administration (VHA) Business organizations and Office of Information (OI) technical staff.

### 5.2. Pass/Fail Criteria

**Pass** – A test script will be considered as passed when all of the conditions, steps, and requirements that are indicated as being covered within the test script were executed without a variance in the described functionality and expected outcome.

**Fail** – A test script will be considered as failed when all of the conditions, steps, and requirements that are indicated as being covered within the test script were executed and any of the following conditions exist.

- Any defect that can compromise patient safety or system security. Examples of system security defects include breach of confidentiality requirements of the Privacy Act, Health Insurance Portability and Accountability Act (HIPAA). Any requirements that might potentially impact patient safety will be marked as such.
- Any defect that causes corruption of data from a result of the system (as opposed to user error) that cannot be corrected, within reason through the application.
- Any defect in which inappropriate transmissions are consistently generated or appropriate transmissions of Health Level Seven (HL7) messages fail to be generated.
- Any defects in the functionality which result in erroneous information being communicated to the user or patient.
- Any defects in the functionality resulting in a failure of all or part of the application where the expected results can temporarily be achieved by alternative means and the customer indicates the workaround is acceptable for the short term.
- A crash or hang that prevents further testing of the complete application or a section of the application.
- A loss of system functionality with no suitable workaround (i.e., there is no way to achieve the expected results using the application).
- A minor functionality is not working as intended and a workaround exists that the customer determines does not prevent the user from continuing to test the affected part of the system but may not be desirable in the long term.
- The inability of a valid user to access the system consistently with granted privileges.
- Any cosmetic issues that do not result in data entry and or data quality problems.
- A minor loss of or defect in the functionality.
- Any low-level cosmetic issues.

### 5.3. Suspension and Resumption Criteria

**Suspension** – Every effort will be made to continue testing when failures are found. However, some failures can affect product functionality to the point where further execution is not advisable. At this point, testing will be suspended or blocked until a fix or workaround is provided. An example if scheduling an appointment could not be completed using VistA Scheduling, this would block any further testing since this is core functionality and is required as a precondition for testing.

**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 associated test procedures or test cases.

### 5.4. Acceptance Criteria

The following are the acceptance criteria that must be satisfied for the patch/build to be accepted by the customer or other authorized entity.

- All business and functional requirements have been validated
- All test cases/scripts are complete
- All tests have been executed
- The UAT test environment(s) are validated/verified
- Test data requirements are determined and met
- All Critical and Major defects are corrected
- Customer approves ALL remaining open defects within the Scheduling application and agrees to a corrective action plan within the remaining testing phases

## 6. Test Deliverables

The table below lists the test deliverables for the VSE project.

**Table 6-1: Test Deliverables**

<b>Test Deliverables</b>	<b>Responsible Party</b>
Master Test Plan	Test Team/Testers
CI/ST Test Scripts	Test Team/Testers
CI/ST Defect Log	Test Team/Testers
CI/ST Evaluation Summary	Test Team/Testers
CI/ST Execution Log	Test Team/Testers
SQA Review Checklist	Test Team/Testers
UAT Test Cases/Scripts	Test Team/Testers
UAT Defect Log	Test Team/Testers
UAT Evaluation Summary	Test Team/Testers
UAT Execution Log	Test Team/Testers
Updated Source Code	Test Team/Testers
Requirements Traceability Report or Matrix	Test Team/Testers

## 7. Test Schedule

Please refer to the VSE Project Schedule.

## 8. Test Environments

The HPES project team has acquired and designated the following test environments within Albany and Bay Pines:

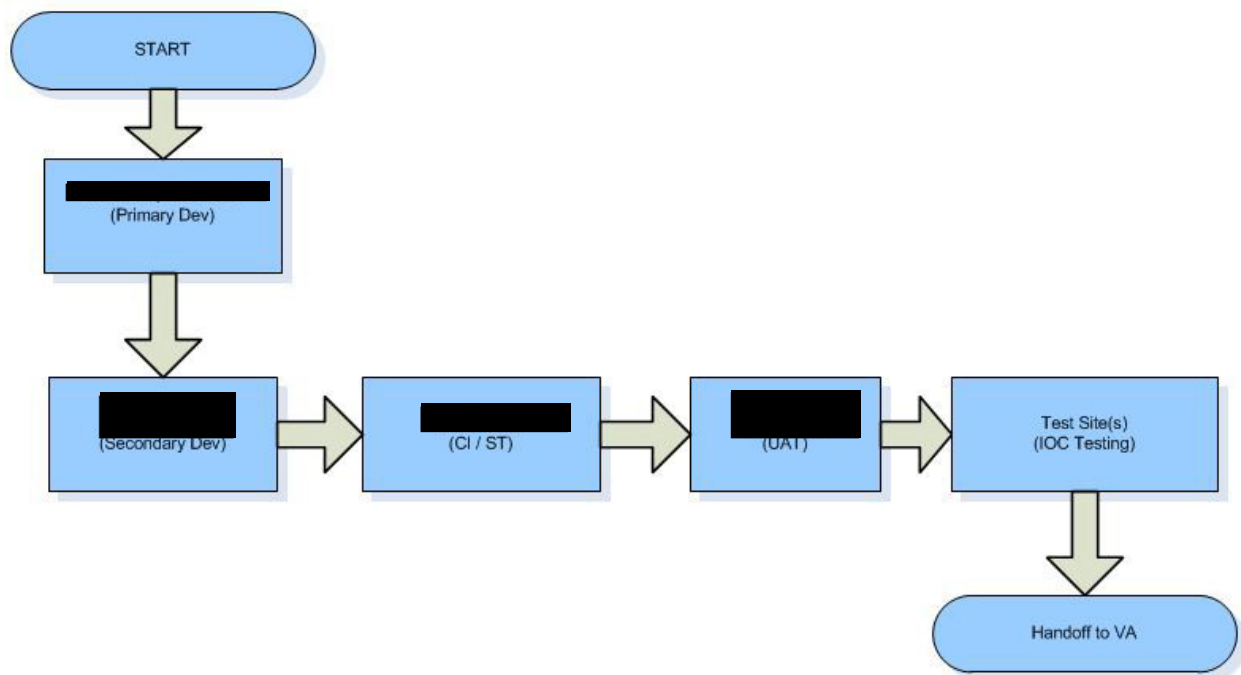
- **CHEY134 (Bay Pines) – CI/ST Test Environment**

The CHEYL134 account will be used for VSE Component Integration/System testing by the VSE SQA Team to conduct VSE testing. The Testing team will control the configuration of the environment by requesting when patches are to be installed within this environment. This applies to nationally released patches, test patches from other teams as well as patches delivered to SQA from the VSE Development Team. This account will be under strict configuration control following the processes outlined below.

- **CHEY135 (Bay Pines) – UAT Test Environment**

The CHEYL135 account will be used for VSE Component Integration/System testing by the VSE SQA Team to conduct VSE testing. The Testing team will control the configuration of the environment by requesting when patches are to be installed within this environment. This applies to nationally released patches, test patches from other teams as well as patches delivered to SQA from the VSE Development Team. This account will be under strict configuration control following the processes outlined below.

**Figure 1: Test Environment Flow Diagram**



## **8.1. Test Environment Configurations**

The HPES VSE Configuration Manager will manage the test environments.

## **8.2. Base System Hardware**

The VSE project is updating existing software; therefore, no changes are required to existing base system hardware unless otherwise specified.

## **8.3. Base Software Elements in the Test Environments**

- VistA Surgery (SR\*3\*182 and other required patches)
- VistA TIU (TIU\*1\*270)
- CPRS v29 (client install)
- Attachmate Reflections (client install)
- VistA Registration (DG\*5.3)
- VA FileMan (V22.0)
- CLINICAL REMINDERS (PXRM\*2.0)
- Scheduling Appointment Menu(SD\*5.3)
- VistA - Patient Care Encounters (PCE\*1.0)
- Ambulatory Care Reporting

## **9. Staffing and Training Needs**

Please refer to the VSE Project Management Plan.

## **10. Risks and Constraints**

At this point and time the HPES project team has not identified any risks or constraints for CIT/ST test execution. The risks identified within this VistA Scheduling will be recorded and tracked in MS Excel.



## 11. Test Metrics

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

The VSE project can be measured against the following Test metrics:

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

# Attachment A - Approval Signatures

The MTP documents the project's overall approach to testing and includes:

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

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

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

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

REVIEW DATE: *<date>*

*< Program/Project Manager >*

---

Signed:

Date:

*< Business Sponsor Representative >*

---

Signed:

Date:

*< Integrated Project Team (IPT) chair >*

## A. Test Type Definitions

Test analysts use “test types” to validate the system or application under test. Simply put, test types are test techniques used to exercise the system or application. This table presents a listing of possible test types that may be utilized during the Product Build, Independent Testing, Operational Readiness Review (ORR) and IOC Testing. The test analyst in consultation with the Development Manager selects the test types best suited to the system or application being tested. A minimum set of test types is suggested here. More tests may be added at the discretion of the Development Team.

Description	Product Build Testing	Independent Testing	IOC Testing
Types of Test			
Access Control Testing	X		
Benchmark Testing			
Build Verification Testing		X	
Business Cycle Testing			X
Compliance Testing	X		
Component Integration Testing			
Configuration Testing			
Contention Testing			
Data and Database Integrity Testing			
Documentation Testing	X		X
Error Analysis Testing			X
Exploratory Testing	X	X	X
Failover Testing			
Installation Testing	X		
Integration Testing	X	X	
Load Testing		X	
Migration Testing			
Multi-Divisional Testing			
Parallel Testing			
Performance Monitoring Testing			
Performance Testing		X	
Privacy Testing	X		

Description	Product Build Testing	Independent Testing	IOC Testing
Product Component Testing	X		
Recovery Testing			
Regression Test	X		
Risk Based Testing	X		X
Section 508 Compliance Testing	X		
Security Testing			
Smoke Testing	X	X	
Stress Testing		X	
System Testing	X	X	
Usability Testing			X
User Functionality Testing	X		
User Interface Testing			

## A.1. Test Type Definitions

Test Type	Definition
Access Control Testing	A type of testing that attests that the target-of-test data (or systems) are accessible only to those actors for which they are intended, as defined by use cases. Access Control Testing verifies that access to the system is controlled and that unwanted or unauthorized access is prohibited. This test is implemented and executed on various targets-of-test.
Benchmark Testing:	A type of performance testing that compares the performance of new or unknown functionality to a known reference standard (e.g., existing software or measurements). For example, benchmark testing may compare the performance of current systems with the performance of the Linux/Oracle system.
Build Verification Testing (Prerequisite: Smoke Test)	A type of testing performed for each new build, comparing the baseline with the actual object properties in the current build. The output from this test indicates what object properties have changed or don't meet the requirements. Together with the Smoke test, the Build Verification test may be utilized by projects to determine if additional functional testing is appropriate for a given build or if a build is ready for production.

Test Type	Definition
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 programs compatibility with as many possible configurations of hardware and system software. In most production environments, the particular hardware specifications for the client workstations, network connections, and database servers vary. Client workstations may have different software loaded, for example, applications, drivers, and so on hand, at any one time; many different combinations may be active using different resources. The goal of the configuration test is finding a hardware combination that should be, but is not, compatible with the program.
Contention Testing	A type of performance testing that executes tests that cause the application to fail with regard to actual or simulated concurrency. Contention testing identifies failures associated with locking, deadlock, livelock, starvation, race conditions, priority inversion, data loss, loss of memory, and lack of thread safety in shared software components or data.
Data and Database Integrity Testing	A type of testing that verifies that data is being stored by the system in a manner where the data is not compromised by the initial storage, updating, restoration, or retrieval processing. This type of testing is intended to uncover design flaws that may result in data corruption, unauthorized data access, lack of data integrity across multiple tables, and lack of adequate transaction performance. The databases, data files, and the database or data file processes should be tested as a subsystem within the application.

Test Type	Definition
Documentation Testing	<p>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:</p> <p>Release Notes, Installation Guide, User Manuals, Technical Manual, and Security Guide.</p>
Error Analysis Testing	<p>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.</p>
Exploratory Testing	<p>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.</p>
Failover Testing	<p>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.</p>
Installation Testing	<p>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.</p>

Test Type	Definition
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	<p>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:</p> <ul style="list-style-type: none"> <li>• Data conversion has been completed</li> <li>• Data tables are successfully created</li> <li>• Parallel test for confirmation of data integrity</li> <li>• Review output report, before and after migration, to confirm data integrity</li> <li>• Run equivalent process, before and after migration</li> </ul>
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.

Test Type	Definition
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 (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.



Test Type	Definition
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.

## B. Acronyms

Term	Meaning
ACAP	Access and Clinic Administrative Program
AITC	Austin Information Technology Center
CI	Component Integration
CIT/ST	Component Integration/System Test
CVS	Conformance Validation Statement
GUI	Graphical User Interface
HIPAA	Health Insurance Portability and Accountability Act
HL7	Health Level Seven
HPES	Hewlett-Packard Enterprise Services
IOC	Initial Operating Capability
ISO	Information Security Officer
IT	Information Technology
MTP	Master Test Plan
N/A	Not Applicable
NPCDB	National Patient Care Database
OI&T	Office of Information and Technology
ORR	Operational Readiness Review
PM	Project Manager
PMAS	Program Management Accountability System
PWS	Performance Work Statement
RTM	Requirement Traceability Matrix
SDD	System Design Document
SEDR	Systems Engineering and Design Review
SME	Subject Matter Expert
SOP	Standard Operating Procedures
T4	Transformation Twenty-One Total Technology
TBD	To Be Determined
TRM	Technical Reference Model
TRR	Test Readiness Review
UAT	User Acceptance Testing

<b>Term</b>	<b>Meaning</b>
VA	Department of Veterans Affairs
VDL	VA Document Library
VHA	Veterans Health Administration
VistA	Veterans Health Information System and Technology Architecture
VSE	VistA Scheduling Enhancements