

Department of Veterans Affairs

Mental Health eScreening

System Administration Manual



December 2015
Software Version 1.0

Revision History

Date	Version	Description	Author
December 2015	1.4	<p>Added “64-bit” to JDK 8, 2.1, step 1.</p> <p>Added clarification of VistALink Listener Port in 2.3.1.</p> <p>Added note to 2.3.3 that Java options require a dash, and link to a Tomcat service reference, and image of the Java options box.</p> <p>Added section 2.3.5 Installing Maven.</p> <p>Added information to 2.3.6 for Instance IIS.</p> <p>Added section 2.3.7 Restarting Tomcat.</p> <p>Added mention of encrypted flag to 2.3.9 (in addition to original reference 4.3)</p> <p>Added error information & screenshots to 2.4.</p> <p>Added section 2.5 Updating eScreening.</p> <p>Added log locations to table 20 in 4.2.1.</p> <p>Added section 4.2.2 Dissection of error message.</p> <p><i>Appendix updates:</i></p> <p>Updated Hyperlinks permanent links.</p> <p>Provided .XML file from SVN.</p> <p>Added steps for VistALink .JAR files.</p>	<p>A. Boen</p> <p>E. Floto</p> <p>L. Deighan</p> <p>C. Hichak</p> <p>K. Rizvi</p> <p>R. Kumar</p>
November 2015	1.3	Corrected SQL reference in table, section 1.4	Information Innovators Inc.
November 2015	1.2	Added SSL information and procedure, and updated Java 7 to Java 8. Also updated deployment procedures: sections 2.1 and 2.4.	Information Innovators Inc.
August 2015	1.1	Minor updates to deployment procedure	Information Innovators Inc.
July 2015	1.0.9	Added VA proxy user account set-up	Information Innovators Inc.
July 2015	1.0.8	Simplified the setup process	Information Innovators Inc.
June 2015	1.0.7	Final document	L. Deighan
March 2015	1.0.6	Minor updates	L. Deighan
January 2015	1.0.5	San Diego server deployment data	L. Deighan
June 2014	1.0.4	Harmonized graphics	M. Roberts
June 2014	1.0.3	Sprint 16 updates	M. Roberts
March 2014	1.0.2	Sprint 11 update	M. Roberts
February 2014	1.0.1	Draft	M. Roberts

Contents

1. System Business and Operational Description	1
1.1. Operational Priority and Service Level	1
1.2. Logical System Description	2
1.3. Physical System Description	5
1.4. Software Description	7
1.4.1. Background Processes	8
1.4.2. Job Schedules	8
1.4.3. Dependent Systems.....	8
2. San Diego Server Deployment Information.....	10
2.1 Pre-requisite for deploying the eScreening application on a Windows server (initial setup only)	10
2.2. eScreening production deployment information	11
2.3. Creating a new eScreening instance.....	11
2.3.1. Configuring a Proxy User Account for MHE.....	11
2.3.2. Creating the Instance Tomcat Base Directory	12
2.3.3. Installing Instance IIS Service	13
2.3.4. Setting the Tomcat Parameters	13
2.3.5. Installing Maven.....	14
2.3.6. Adding Instance IIS Proxy Rule.....	15
2.3.7. Restarting Tomcat	18
2.3.8. Creating an Instance Database.....	19
2.3.9. Creating an Instance Deployment Staging Area	19
2.4. Deploying a new eScreening instance.....	20
2.5. Updating eScreening	21
2.6. SSL Certificate	22
3. Routine Operations.....	23
3.1. Administrative Procedures	23
3.1.1. System Start-up	23
3.1.2. System Shut-down.....	24
3.1.3. Back-up & Restore.....	24
3.1.4. Back-Up Procedures.....	24
Restore Procedures.....	26
Back-Up Testing	26
Storage and Rotation.....	27
3.2. Security and Identity Management.....	27
3.2.1. Identity Management	28
3.3. User Notifications	29

3.4. System Monitoring, Reporting, & Tools.....	29
3.4.1. Availability Monitoring	29
3.4.2. Performance and Capacity Monitoring	29
3.4.3. Critical Metrics	30
3.5. Routine Updates, Extracts, and Purges.....	30
3.6. Scheduled Maintenance	31
3.7. Capacity Planning.....	31
4. Exception Handling.....	32
4.1. Routine Errors.....	32
4.1.1. Security Errors.....	32
4.1.2. Time-outs.....	33
4.1.3. Concurrency.....	34
4.2. Significant Errors.....	35
4.2.1. Application Error Logs	35
4.2.2. Dissection of error message.....	37
4.2.3. Application Error Codes and Descriptions.....	37
4.2.4. Infrastructure Errors.....	38
Database	38
Web Server and Application Server.....	39
Network	39
Authentication & Authorization.....	39
4.3. Dependent System(s)	39
4.4. Troubleshooting.....	40
4.5. System Recovery	43
4.5.1. Restart after Non-Scheduled System Interruption.....	43
4.5.2. Restart after Database Restore	43
4.5.3. Back Out Procedures	44
5. Operations & Maintenance System Support.....	45
5.1. Support Structure	45
5.1.1. Support Hierarchy.....	45
5.1.2. Division of Responsibilities	45
5.2. Support Procedures	45
Appendix: Setting Up Your Development Environment.....	47

1. System Business and Operational Description

Mental Health eScreening (MHE) is a software application for automating the manual, paper-based process of screening Veterans for mental health issues. It consists of a web-based assessment runtime, a database for storing assessment data, a web-based user-friendly forms editor for designing assessments and notes templates, and a web administrative dashboard for operating the system. eScreening will replace manual screening processes in San Diego OEF/OIF/OND, Aspire Center, Mental Health, and Primary Care locations, as well as within the Department of Veterans Affairs (VA) specified care settings in four other VISN 22 potential pilot sites: Las Vegas, Loma Linda, Long Beach, and Greater Los Angeles.

The application exchanges data directly with VistA, primarily consisting of pulling open clinical reminders, pulling Veteran identification and demographic data, inserting Veteran assessment data in the form of notes, and closing clinical reminders based on completion of assessments, as well as creating new clinical reminders and inserting health factors based on the results of screening.

The system will be hosted on the VA network and will operate as follows:

- Assessment runtime: Designed to be run from a tablet browser, with the target configuration being Safari on iPad
- Administrative dashboard and forms editor: Designed to be run from a desktop browser, with the target configuration being Firefox 26 or later on Windows 7.

eScreening is a joint product of Center for Stress and Mental Health (CESAMH) and VHA Center for Innovation (VACI). The principal stakeholders are:

- Niloofar Afari, PhD: Division Director, Mental Health Integrative and Consultative Care Services, VA San Diego Healthcare System; Director of Clinical Affairs, VA Center of Excellence for Stress and Mental Health; Associate Professor of Psychiatry, UCSD Health System
- Clint Latimer: VA Innovation Coordinator/Project Manager/Contracting Officer Representative (COR), FAC-P/PM
- James Pittman (Co-Sponsor/SME, CESAMH & Department of Social Work)
- Elizabeth Floto (Project Manager, CESAMH)

The implementation work is performed by a contractor team of engineers and clinicians under VACI Innovation project 20388, contract VA118-11-D1002. A full list of stakeholders may be found within the roles and responsibilities section.

1.1. Operational Priority and Service Level

eScreening's overall importance to VA lies within its ability to improve the efficiency of existing or planned mental health screening procedures. The application improves care by scoring Veteran assessments in real-time, alerting VA personnel and auto-scheduling clinical reminders when mental, physical, or behavioral health symptoms are specified, and allowing VA to identify

and treat severe conditions within minutes that would normally take days or weeks with paper-based screening systems. The customers served are the Veterans who consume services provided by San Diego Mental Health, Primary Care, OEF/OIF/OND, and the Aspire Center.

eScreening is considered *important*, but not *critical* for patient care, in most cases. If the system is down or otherwise unavailable, patients' safety will not be immediately compromised and paper-based screening will be available as a contingency. However, the system's unique ability to identify and report Veteran mental or behavioral health issues in real time implies that service should be prioritized above other non-critical systems.

There is currently no Service Level Agreement (SLA) in place for eScreening. The need for an SLA will be determine and addressed by the VA COR/PM and VA San Diego IT.

1.2. Logical System Description

eScreening consists of a new application for designing, performing, and publishing mental health assessments, as well a data repository and an existing VistA instance. This figure displays the basic system components and their composition:

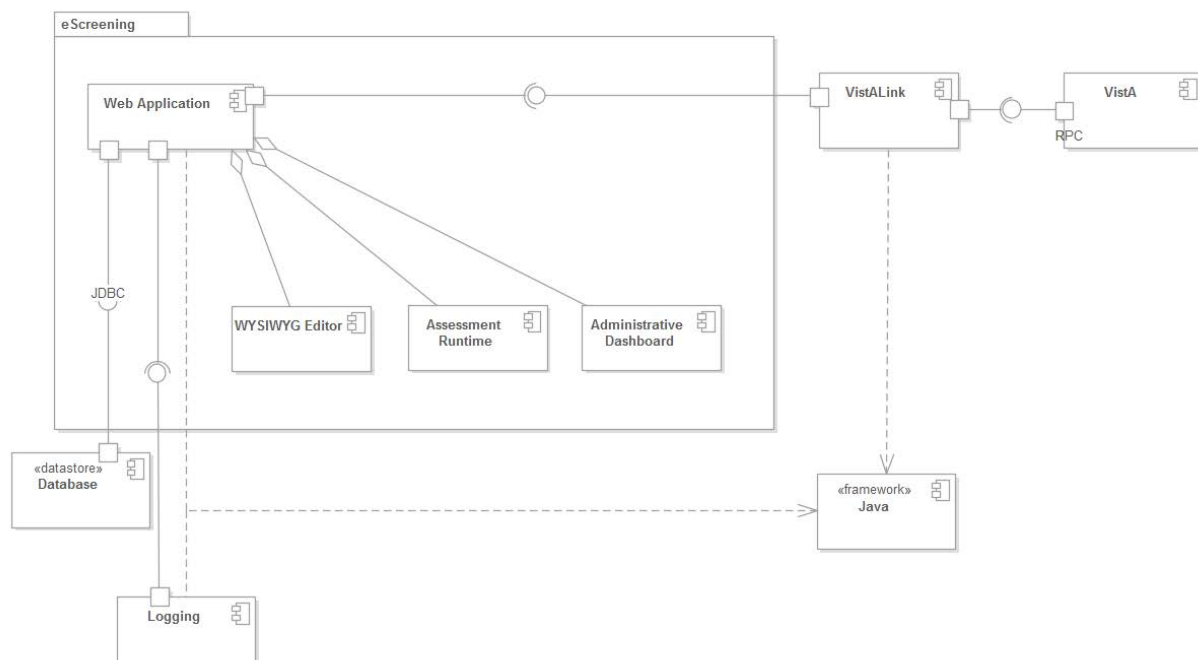


Figure 1: eScreening Logical Components

The application components are:

- Forms Editor: A What You See is What You Get tool for designing assessment forms and note templates. Staff use the designer to create or edit existing assessment and notes

templates; the assessment forms are then used by the assessments runtime and the notes templates are used by the dashboard.

- **Runtime:** The runtime executes assessment forms created by the designer. Veterans “take” assessments by inputting answers into questions within forms based on the templates. The assessment session and the answer to the forms are stored in the repository.
- **Dashboard:** The dashboard allows staff to create assessment sessions based on assessment forms, view the status of ongoing assessments, and upload the results of assessments to VistA based on note templates created by the designer. The dashboard uses the repository to track and store assessments and templates.
- **Database:** A repository of assessment forms, users, ongoing and historical assessments, and assessment metadata used by the designer, runtime, and dashboard.

This diagram visualizes the integrations between the application, the repository, and VistA:

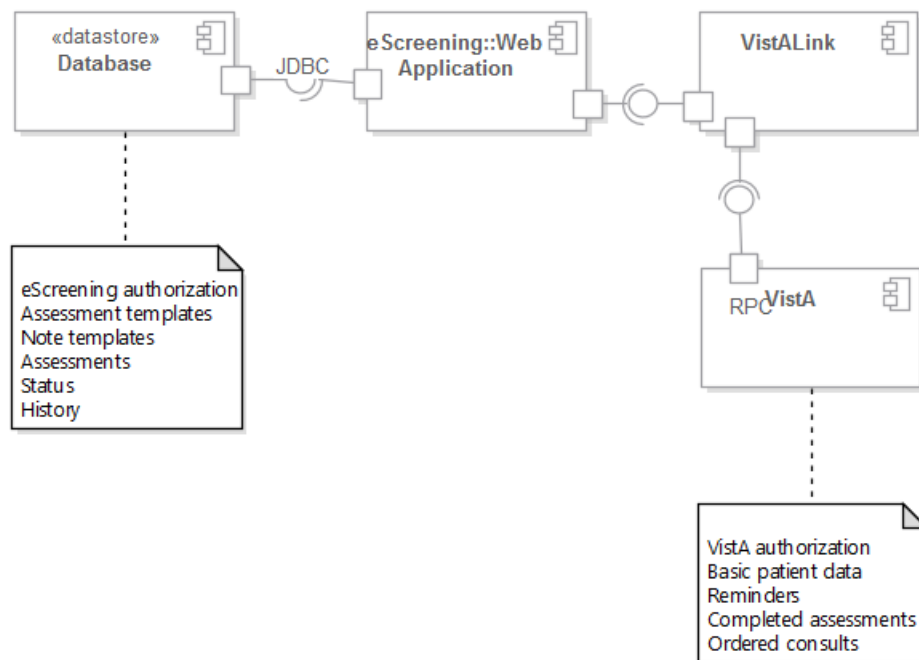


Figure 2: Logical Integration

The assessment process consists of staff creating an assessment session for a Veteran, the Veteran taking the assessment, staff reviewing the assessment, and eventually uploading the results of the assessment to VistA. The workflow is described below.

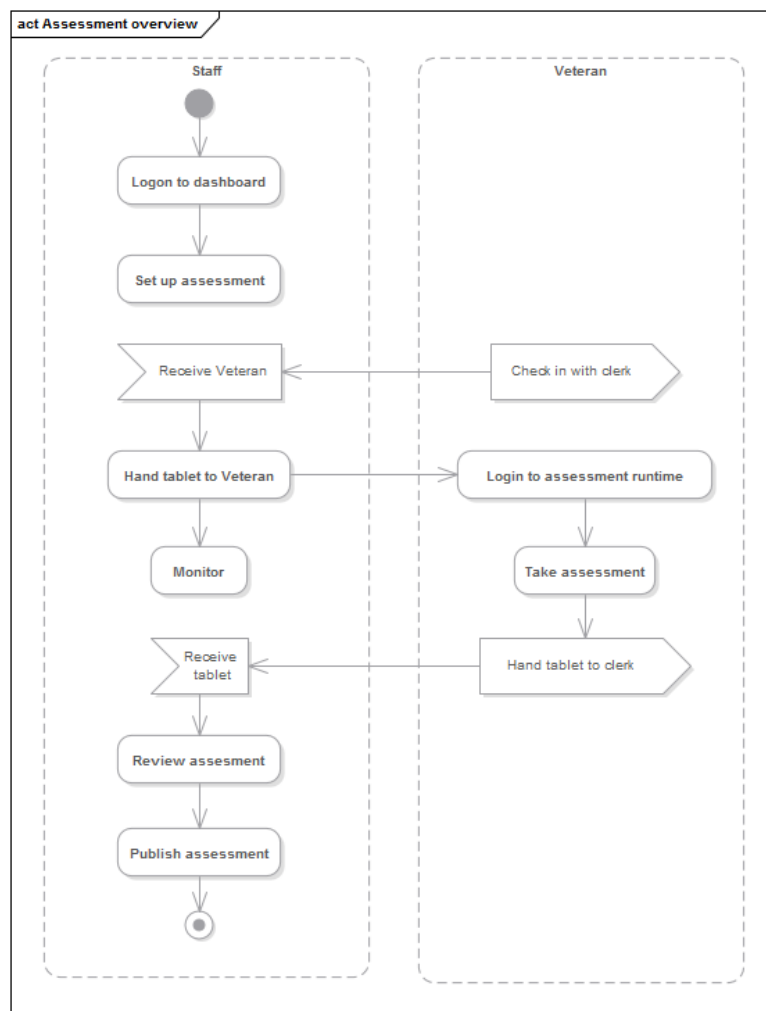


Figure 3: Perform Assessment Flow

Each of the application components utilizes the repository, but all three of the primary functions utilize VistA as well. The designer uses VistA for reference data in designing forms; the runtime uses VistA for authenticating Veterans and pulling basic Veteran data (such as demographics and open clinical reminders) for use in assessments, and the dashboard publishes assessment results to VistA in the form of notes, new reminders, etc. The following figure details the logical data integrations that occur between the application, VistA and the repository:

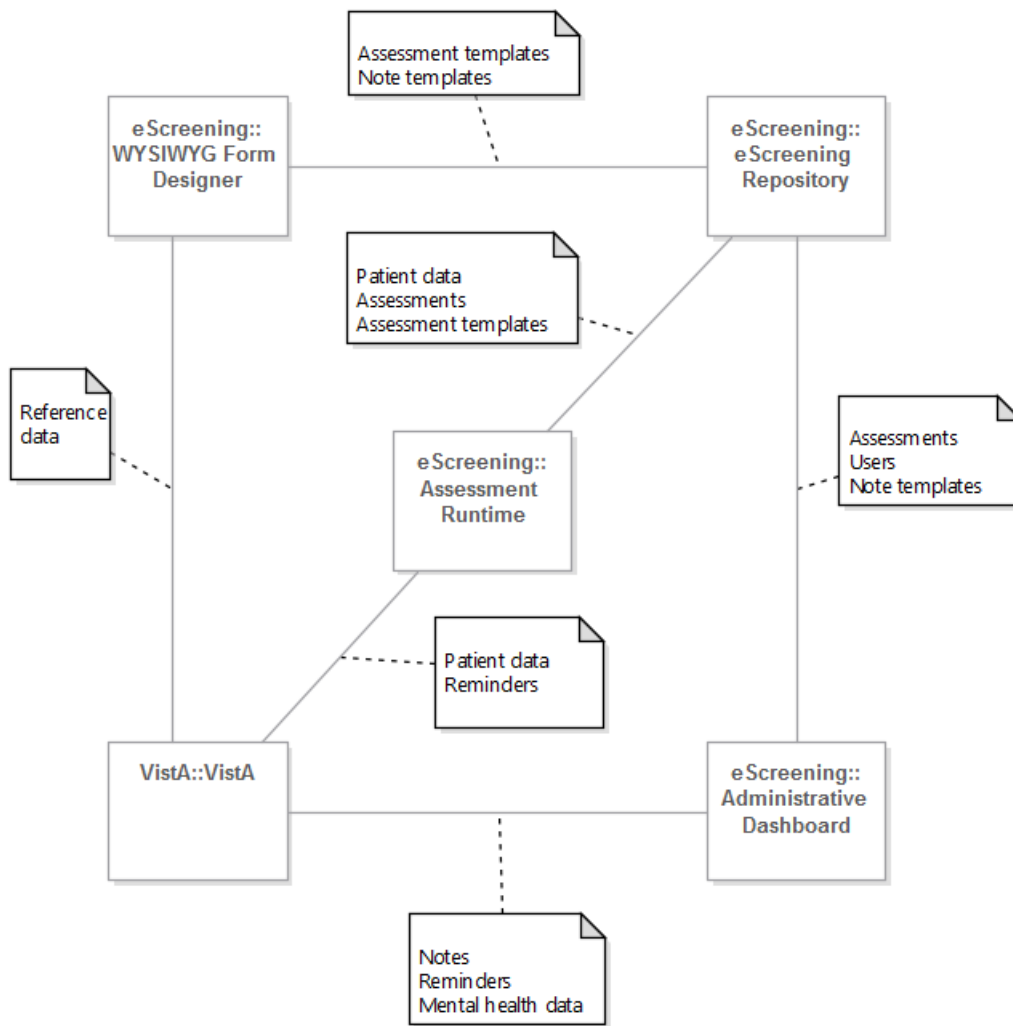


Figure 4: Logic Data Integration

Note: The application also caches some reference data based on lookups of VistA data. For example, the application currently pulls health factors periodically for use in the forms editor. This reference data does not change often and is safe to cache for short periods of time.

1.3. Physical System Description

The physical eScreening hardware consists of one physical server and 600 tablets. The eScreening application runs on the physical server in the San Diego VA Medical Center. Staff access the dashboard and designer components from VA workstations. Staff and Veterans access the runtime component from HTML5-capable browsers on tablet devices.

This table details the specs for the eScreening hardware:

Item	Make	Model	OS	Memory	Storage	Location
Server	Dell	R420	Windows Server 2012	64 GB	1.2 TB (after RAID 10)	VASD data center
Tablet	Samsung	Slate	Windows 7 Enterprise	4 GB	118 GB	SD VAMC
Tablet	Apple	iPad2	iOS 7.1	512 MB	16 GB	Each program location

Table 1: System Hardware

The application server hardware is a rack-mount server with the following rack and electrical footprint:

Element	Attribute
Form factor	2U
Power	Dual hot plug 550W power supplies, 2 x 15 amp 10 ft. wall plug

Table 2: Server Data Center Specifications

The server additionally contains 12 CPU cores (6 physical, 6 virtual), and can be upgraded to include another CPU for a total of 24 cores. The memory can be upgrade to a total of 384 GB 1600 MT/S over 12 DIMM slots. The internal storage can be upgraded to a maximum of 16 TB (8 TB usable via RAID 10).

The tablets connect to the server and the server connects to Vista. The tablets talk HTTP over TLS to the server via a SD VAMC 11g wireless network. The eScreening server communicates with Cache via RPC over port 8000. The diagram below shows all device communications, including type and bandwidth:

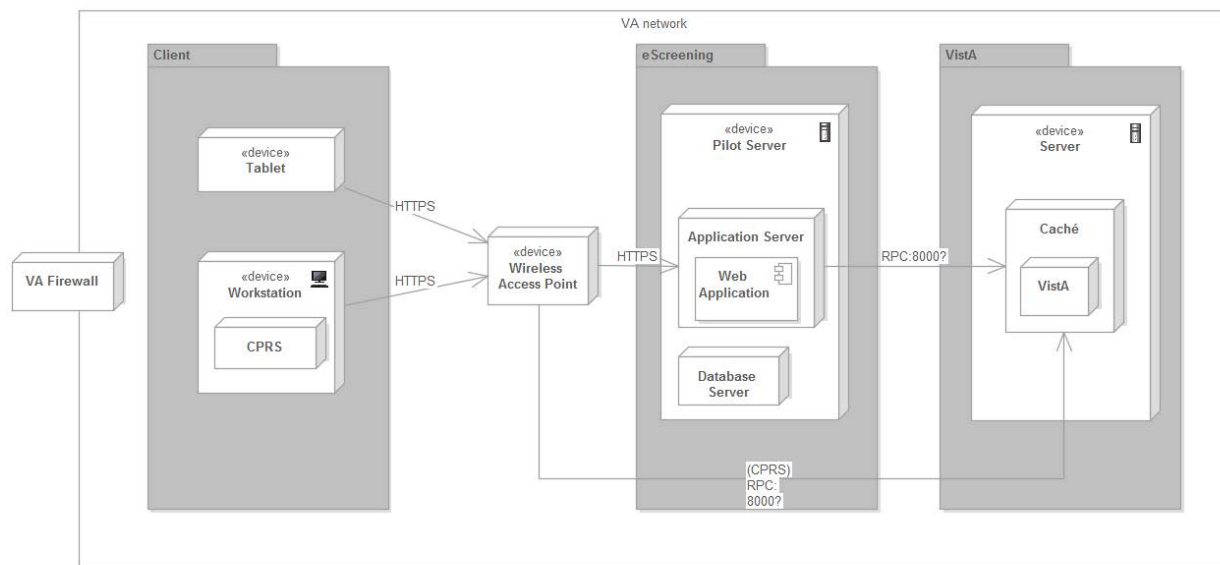


Figure 5: eScreening Hardware Connections

In OEF/OIF/OOO outreach scenarios, tablets connect to the VA network over VPN and MIFI. All communication between eScreening and VistA takes place behind the VA firewall.

1.4. Software Description

The system consists of the following components:

Web application: An application comprised of JavaScript, HTML5, and CSS3 on the presentation layer and Java on the service and data access layers. The application performs authentication/authorization against the eScreening database. It provides screening services to Veterans, and administration and reporting features to staff. The application integrates with VistA via VistALink.

- **Database:** A MySQL database that stores Veteran screening data and metadata, as well as VA staff credentials and permissions.
- **VistA:** The application integrates with VistA for security, basic Veteran information, clinical reminders, health factors, consults, and notes.

A full list of the software used in the system is described below:

Category	Product	License
Application	HTML5, CSS3, JavaScript, JQuery	Open source
Framework	Java 8 64 bit Oracle VM, Spring 3.2.6-RELEASE	Open source
Web server	Apache Tomcat 7 servlet container	Open source

Category	Product	License
Database	MySQL Ver 14.14 Distrib 5.6.19, for Win 64 (x86_64)	Open source
Integration	VA VistALink 1.6	VA
Operating system	Windows Server 2012 with 1.2 TB disk RAID 10	Commercial (provided)

Table 3: Software Used in eScreening

All application software is open source or provided by the VA. The operating system is San Diego's preferred operating system (Windows), but there are no Windows-specific components to the system.

1.4.1. Background Processes

The application background processes are:

- java.exe: The container technology hosting the web application servlet
- mysqld.exe: The server daemon for the MySQL database

Note: MySQL is configured to run as a Windows service so that it starts automatically with Windows.

1.4.2. Job Schedules

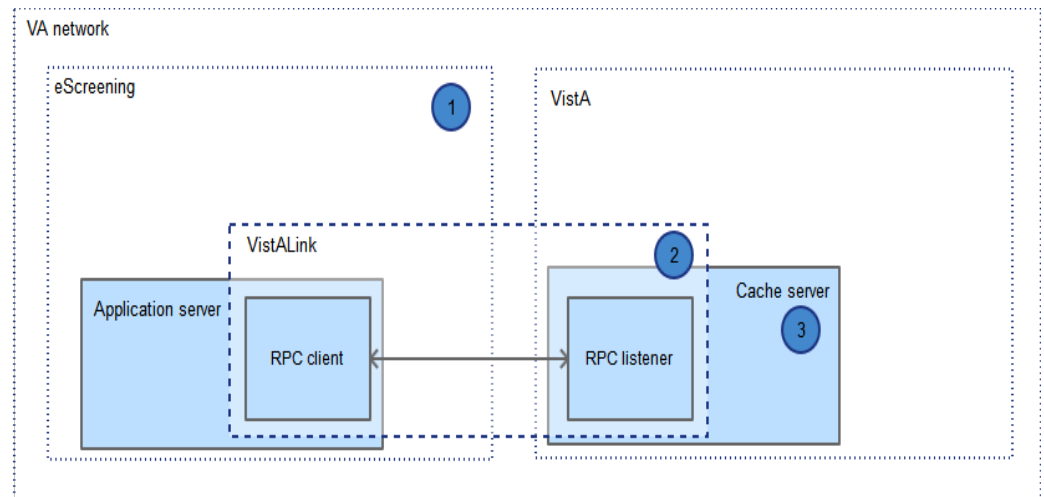
No batch jobs are currently scheduled to run on any interval.

1.4.3. Dependent Systems

The application requires VistA in order to function. The application uses VistA for security, general patient data, clinical reminders, health factors, and clinical notes. The following attributes describe the integration with the dependent system:

- Name: VistA
- Location: San Diego VAMC
- Function: Repository for existing patient data and events, as well data and events generated by eScreening
- Interface method: RPC over VistALink

This diagram describes how eScreening integrates with VistA:



Legend

- ❶ The eScreening application uses the VistALink Java client library
- ❷ VistALink provides bi-directional communication between the client and the server
- ❸ The Cache (M) server runs the VistALink listener

All communication between eScreening and VistA takes place behind VA firewalls via VA VistALink, a Java RPC framework that is part of the OneVA architecture. eScreening uses VistALink for bidirectional communications between the client and VistA (M) server. eScreening utilizes existing CPRS RPCs to perform the same actions as CPRS without requiring any VistA code changes.

2. San Diego Server Deployment Information

2.1 Pre-requisite for deploying the eScreening application on a Windows server (initial setup only)

1. Install **JDK 8, 64-bit**, from Oracle.
2. Install and set up **Create the MySQL Database**.
Install **MySQL 5.6.17 community edition** (or the latest edition):
<http://dev.mysql.com/downloads/mysql/>
3. Open the MySQL Workbench program that installed with MySQL and connect to the server, then execute the scripts:
/eScreeningDashboard/src/main/sql/initialization/dev_env_run_once.sql
4. Install Tomcat:
<http://tomcat.apache.org/download-70.cgi>
Choose the **64-bit Windows zip** (pgp, md5, sha1), download, and unzip to:
D:/apps/
5. Install VA VistALink JARS to local machine:
 - a. Download the artifact from VA:
https://downloads.va.gov/files/FOIA/Software/Patches_By_Application/XOBV-VISTA%20LINK/XOBV_1_6/
If the above link has expired, contact the VA and obtain a new link.
 - b. Unzip the file.
 - c. Open a command prompt and CD to the folder. For example:
C:\Users\somebody\Desktop\vlj-1.6.0.028\samples-J2SE
 - d. Manually install the three JARS by entering these commands:

```
mvn install:install-file -DgroupId=gov.va.med.vistalink -DartifactId=vljConnector -Dpackaging=jar -Dversion=1.6.0.028 -Dfile=vljConnector-1.6.0.028.jar
```



```
mvn install:install-file -DgroupId=gov.va.med.vistalink -DartifactId=vljFoundationsLib -Dpackaging=jar -Dversion=1.6.0.028 -Dfile=vljFoundationsLib-1.6.0.028.jar
```



```
mvn install:install-file -DgroupId=gov.va.med.vistalink -DartifactId=vljSecurity -Dpackaging=jar -Dversion=1.6.0.028 -Dfile=vljSecurity-1.6.0.028.jar
```
6. Install Git using the Windows download link:
<https://git-scm.com/downloads>
7. Install Maven apache-maven-3.3.9-bin.zip:
<https://maven.apache.org/download.cgi>

2.2. eScreening production deployment information

Currently, MHE is deployed in 2 locations (4 instances). Here is the information for both locations.

Tomcat location:

D:\apps\apache-tomcat

Tomcat service names:

- tomcat-sdc-prod (San Diego production instance)
- tomcat-sdc-test (San Diego test instance)
- tomcat-lon-prod (Long Beach production instance)
- tomcat-lon-test (Long Beach test instance)

Production database names:

- sdc-prod
- lon-prod

The eScreening deployment directories are located at:

D:\escreening

Configurations for each of the Tomcat instances are located under:

D:\apps\tomcatInstances

2.3. Creating a new eScreening instance

Each of the production instances of Tomcat are run individually. Follow the steps below to create a new instance (for example, tomcat-lon-prod).

For each instance you create, create an accompanying TEST instance. Repeat the steps, however, change the variables to <3_letter_abbreviation>-test. See examples above in section 2.2.

2.3.1. Configuring a Proxy User Account for MHE

A Proxy User account should be set up by a VistA Account Specialist or a VistA Support Specialist, using the Foundations Manager Menu in VistA. Here is an excerpt from the *VistALink 1.6 System Management Guide (December 2010)*.

8.6.2. Creating the "Connector Proxy User"

To create a connector proxy user:

1. You must hold the Kernel XUMGR key.
2. Add a new connector proxy user by using the Foundations menu on your M system and choosing the Enter/Edit Connector Proxy User option.
3. The account requires no additional information from what is prompted for by the option.
4. Leave the connector proxy user's Primary Menu empty.

5. Securely communicate the access code and verify code for the connector proxy user to the J2EE system manager setting up access from J2EE to your system. Also communicate the IP and port of your VistALink listener.

****naming convention for the account should contain either:**

CONNECTOR
or
APPLICATION
or
PROXY

...not USER, as it may be confused as a regular user account.

- **Do not enter divisions for a connector proxy user.**
- **Do not enter a primary menu.**
- **Do not also use the connector proxy user as a test "end-user"**
- **Utilize the user only as a connector proxy user.**

The account option for the secondary menu should be enabled for “OR CPRS GUI”. Do not assign a primary menu.

The Clinical Applications Coordinator or VistA Support Specialist should provide the following to the staff setting up the client side of the MHE application (setup is in section 2.3.6):

- TBI Consult Order IEN (quick.order.iem)
- Proxy user access code
- Proxy user verify code
- Proxy user DUZ
- VistA Link IP address or Host name (ex. test.vista.san-diego.med.va.gov)
- VistA Link Listener Port number
- VistA station (or primary station) number
- VistA Sample Patient IEN (can be any Veteran IEN from the given instance)

See *VistALink 1.6 System Management Guide* for further troubleshooting assistance.

The VistALink Listener Port is not the same as the RPC Broker port used to connect to CPRS and other VistA GUI applications. The VistALink Listener Port can be found on a spreadsheet of Port Numbers, IP Addressing, and Site Specific Ports maintained by the Region OI&T.

2.3.2. Creating the Instance Tomcat Base Directory

The new base directory will contain all Tomcat work folders for the new Tomcat instance.

1. Using Windows explorer, navigate to D:\apps\tomcatInstances
2. Copy and paste instance-template directory to this same directory
3. Rename the new directory using the convention:
<3_letter_abbreviation>-prod
4. Update the instanceIDs.txt document with a new entry for this new instance with a new unique ID.
5. Edit the file in: <new instance directory>\conf\server.xml
 1. Update the following using Notepad:
 - a. Server port to 81** where ** is the ID of this server

- b. Http Connector port to 82** where ** is the ID of this server
 - c. AJP Connector port to 83** where ** is the ID of this server
2. **Save**, then close the editor.

2.3.3. Installing Instance IIS Service

The following steps cover how to add a new service responsible for starting and stopping the new Tomcat instance. For help understanding the commands, visit this link:

<https://tomcat.apache.org/tomcat-7.0-doc/windows-service-howto.html>

Note: JAVA options require a dash, otherwise the service will fail to start.

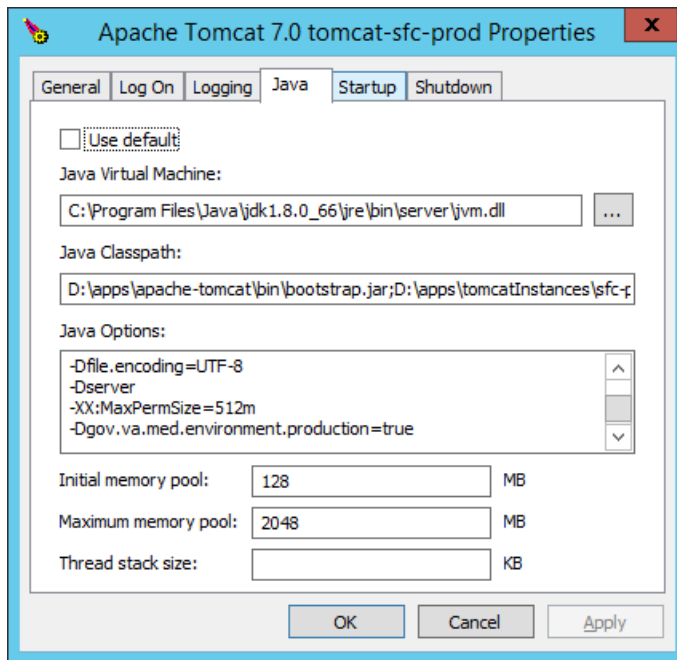
1. Open up a terminal/shell with admin privileges (for example, right-click cmd.exe and select "Run as administrator")
2. If using power shell execute: cmd
3. Run: cd D:\apps\apache-tomcat\bin
4. Run: set CATALINA_HOME=D:\apps\apache-tomcat
5. Run: set CATALINA_BASE=D:\apps\tomcatInstances\<new instance directory>
Here <new instance directory> is the name of the new base directory created in the previous section.
Below "<new_instance_name>" is "tomcat-<new_instance_directory_name>"
6. Run: .\service install <new_instance_name>
7. Run: .\tomcat7 //US//<new_instance_name> --Startup=auto --JvmMx=2048
The JvmMx=2048 sets the maximum memory to 2 GB. This setting should be calculated carefully so that the total amount of memory used by the system is not greater than the total physical memory. Also, the value should be large enough to adequately handle the expected load of the server.
8. To have the new service show up, open the Server Manager and press the F5 key.

2.3.4. Setting the Tomcat Parameters

To edit the Tomcat parameters (e.g. JVM options) for the new instance:

1. Run: cd D:\apps\apache-tomcat\bin
2. Run: set CATALINA_HOME=D:\apps\apache-tomcat
3. Run: set CATALINA_BASE=D:\apps\tomcatInstances\<new instance directory>
4. Run: .\tomcat7w //ES//<new_instance_name>
5. Set the required JVM settings:
 1. Click the **Java** tab
 2. Add these settings in the Java Options text box (include the dash):

-Dfile.encoding=UTF-8
-Dserver
-XX:MaxPermSize=512m

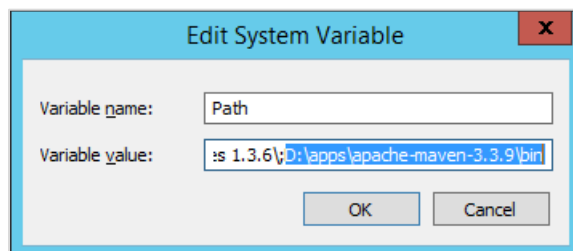


3. If this is a production instance, add:
`Dgov.va.med.environment.production=true`

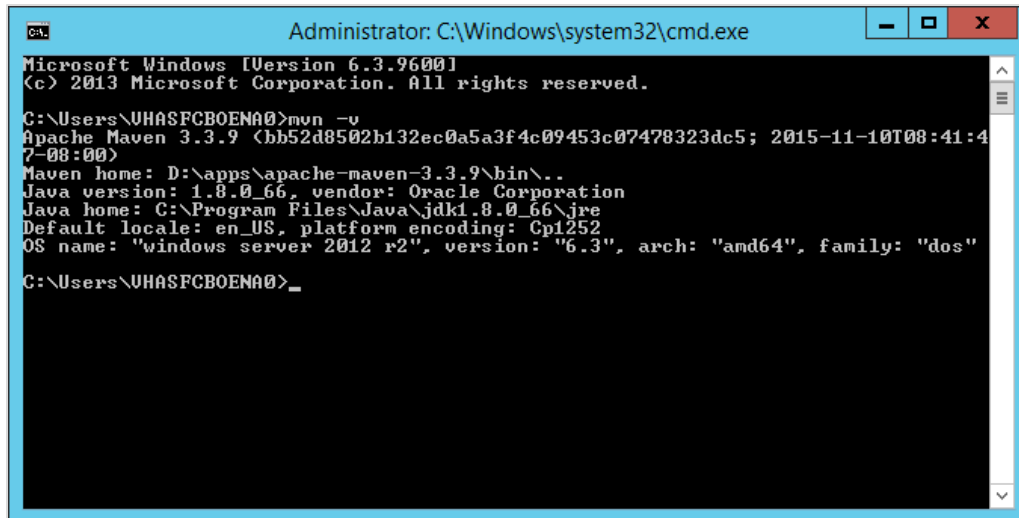
2.3.5. Installing Maven

Maven installation can be followed from the Apache Maven Project site
<https://maven.apache.org/install.html>

1. Extract apache-maven-3.3.9-bin to D:\apps
2. Add D:\apps\apache-maven-3.3.9\bin in the PATH environment variable.



3. Run **mvn -v** in command prompt to verify install.



```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\UHASFBOENAO>mvn -v
Apache Maven 3.3.9 (bb52d8502b132ec0a5a3f4c09453c07478323dc5; 2015-11-10T08:41:47-08:00)
Maven home: D:\apps\apache-maven-3.3.9\bin\..
Java version: 1.8.0_66, vendor: Oracle Corporation
Java home: C:\Program Files\Java\jdk1.8.0_66\jre
Default locale: en_US, platform encoding: Cp1252
OS name: "windows server 2012 r2", version: "6.3", arch: "amd64", family: "dos"

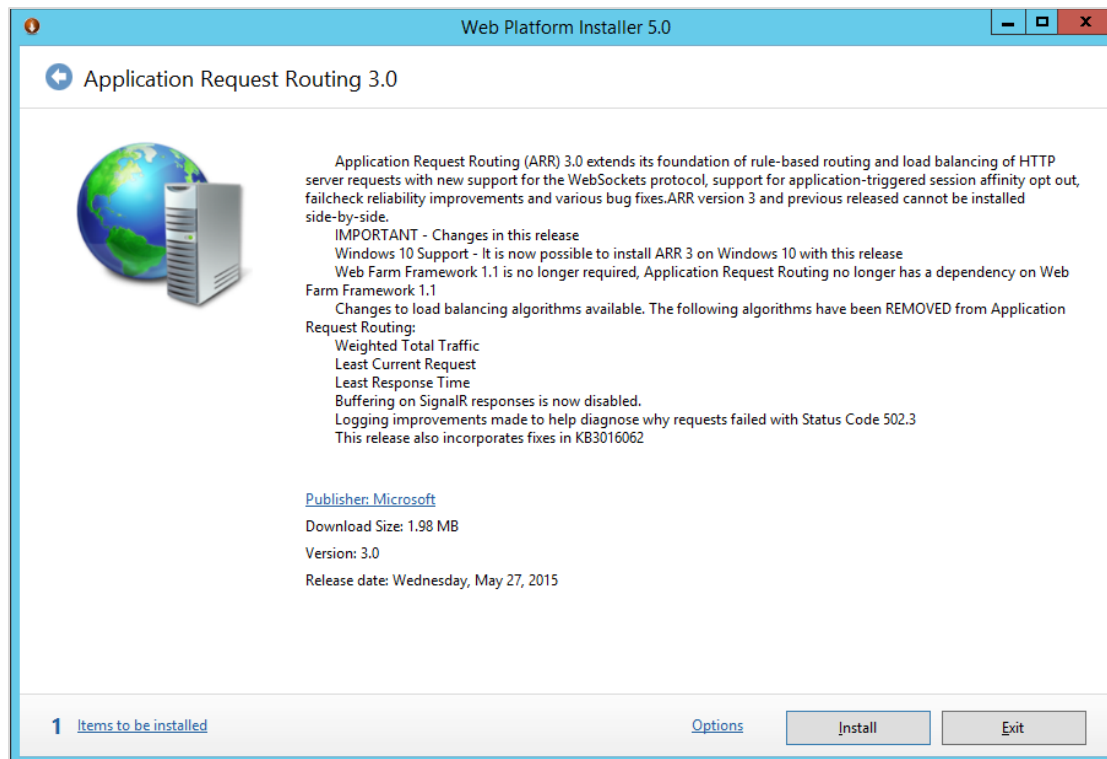
C:\Users\UHASFBOENAO>
```

4. Install the VistALink JAR files using the following commands:
mvn install:install-file -DgroupId=gov.va.med.vistalink -
DartifactId=vljConnector -Dpackaging=jar -Dversion=1.6.0.028 -
Dfile=vljConnector-1.6.0.028.jar
mvn install:install-file -DgroupId=gov.va.med.vistalink -
DartifactId=vljFoundationsLib -Dpackaging=jar -Dversion=1.6.0.028 -
Dfile=vljFoundationsLib-1.6.0.028.jar
mvn install:install-file -DgroupId=gov.va.med.vistalink -DartifactId=vljSecurity -
Dpackaging=jar -Dversion=1.6.0.028 -Dfile=vljSecurity-1.6.0.028.jar

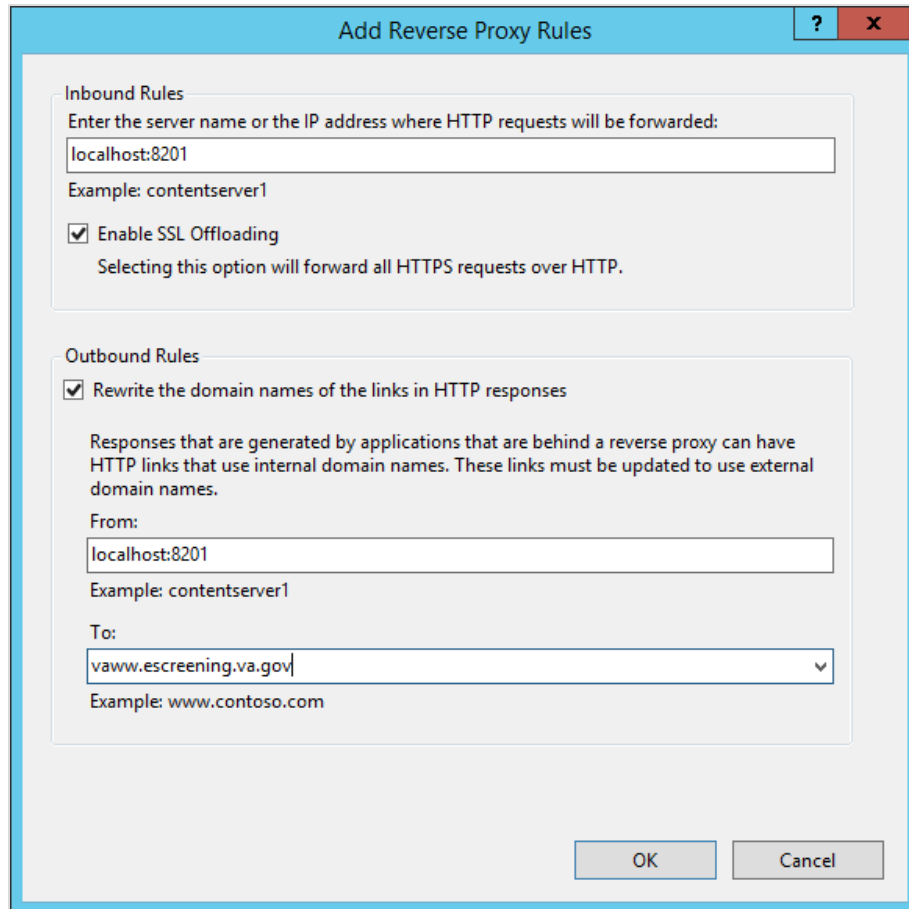
2.3.6. Adding Instance IIS Proxy Rule

To add a proxy rule which uses the URL of incoming client requests and routes the request to the correct Tomcat instance:

1. Install Web Platform Installer.
This will enable you to download and install two additional components:
 - a. URL Rewrite (needed because it's not part of IIS):
<http://www.iis.net/downloads/microsoft/url-rewrite>
 - b. Application Request Routing (needed to configure reverse proxy):
<http://www.iis.net/downloads/microsoft/application-request-routing>



2. Start the new Tomcat service.
3. Open the **IIS Manager**.
4. Unfold: VHASDCAPP22 > Sites > 'Default Web Site'
5. Click **Default Web Site**.
6. Open **URL Rewrite**.
7. Click **Add Rules**.
8. Select **reverse proxy**.
The port to use below is the 82** where ** is the server ID (for example, 8203 for the lon instance).
9. Set the following:
 - a. Inbound field: **localhost:82<server ID>**
 - b. Check off: **Rewrite the domain names of the links in HTTP responses**
 - c. From field: **localhost:82<server ID>**
 - d. To field: **vawww.escreening.va.gov**



The image shows a Windows-style dialog box titled "Add Reverse Proxy Rules". It has a blue title bar with a question mark icon and a close button (X). The dialog is divided into two main sections: "Inbound Rules" and "Outbound Rules".

Inbound Rules

Enter the server name or the IP address where HTTP requests will be forwarded:

localhost:8201

Example: contentserver1

☒ Enable SSL Offloading

Selecting this option will forward all HTTPS requests over HTTP.

Outbound Rules

☒ Rewrite the domain names of the links in HTTP responses

Responses that are generated by applications that are behind a reverse proxy can have HTTP links that use internal domain names. These links must be updated to use external domain names.

From:

localhost:8201

Example: contentserver1

To:

vawww.escreening.va.gov

Example: www.contoso.com

At the bottom right, there are two buttons: "OK" and "Cancel".

10. Click **OK** in the Add Reverse Proxy Rules dialog.
11. Double-Click the new rule from the inbound requests.

Edit Inbound Rule

Name:
ReverseProxyInboundRule1

Match URL

Requested URL:
Matches the Pattern

Using:
Regular Expressions

Pattern:
^sfc\$sfc/.*

Test pattern...

☒ Ignore case

Conditions

Server Variables

Action

Action type:
Rewrite

Action Properties

Rewrite URL:
http://localhost:8201/{R:0}

☒ Append query string

☐ Log rewritten URL

☒ Stop processing of subsequent rules

12. In the Pattern field, enter:
`^<instance_3_letter_code>$|<instance_3_letter_code>\.*`
 For example: `^sdc$sdc\.*`

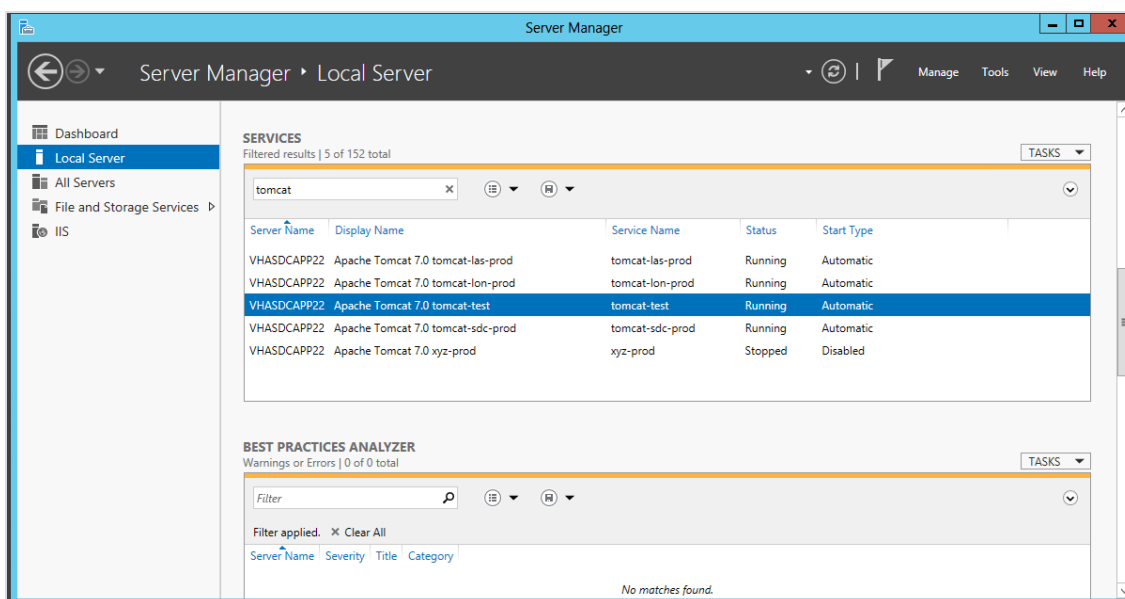
13. In the Rewrite URL field, enter:
`http://localhost:82<server ID>/{R:0}`

14. Click **Apply** and go back to rules.

15. Close the IIS manager.

2.3.7. Restarting Tomcat

1. Open **Server Manager**.
2. Select the system you are trying to restart (in the example below, tomcat-test is selected).
3. Right-Click, then select **Stop Service**.
4. After service has stopped, right-click and select **Start Service**.



2.3.8. Creating an Instance Database

Each Tomcat site will have its own eScreening database schema. There are two MySQL instances running which manage various schemes:

- Test on port 3307
- Production on port 3306

These steps show how to initialize a new database schema:

1. Open the MySQL workbench.
2. Log into the instance, depending on the type of Tomcat instance being deployed (in other words, test or production).
3. Run the following to create the new database (replace *database_name* with the name of each database):

```
CREATE DATABASE IF NOT EXISTS database_name;
```

4. Give the escrapp user permissions to build the new database by (replace *database_name* with the name of each database):

```
GRANT ALL ON database_name.* TO 'escrapp'@'localhost';
```

2.3.9. Creating an Instance Deployment Staging Area

To facilitate simple, error-free eScreening maintenance, each instance has a separate staging area where an instance manages version and database updates.

In the following steps, when <profile_name> is shown, replace this with the name of the Maven profile which has been created for this instance (for example, sdc-prod).

1. Start the **GIT** Bash program.

2. Run:
cd d:/escreening
3. Review the code by running:
git clone https://github.com/VHAINNOVATIONS/Mental-Health-eScreening.git
<profile_name>-release
This operation will create a new directory with eScreening code. For example, the new directory might be called “sdc-prod-release”.
4. Copy and paste the file deploy-template.sh
5. Rename the copy to:
deploy-<profile_name>.sh
For example: deploy-sdc-prod.sh
6. Edit the new file.
7. Set all of the instance-specific parameters using the values previously provided by the CAC (see the bulleted list in section 2.3.1).

When changing the configuration, set the encrypted flag to “false.” This will cause the system to encrypt the connection information and then set the flag back to “true.” (See section 4.3.)
8. Save the file.

2.4. Deploying a new eScreening instance

When <profile_name> is shown below, replace this with the name of the Maven profile which you are deploying (for example, sdc-prod).

1. Start the GIT Bash program
2. Run: cd d:/escreening
3. Run: ./deploy-<profile_name>.sh
 - a. Follow the instructions for the deployment. Use the parameters from Section 2.3.1 to modify the script before running.
 - b. If this is a new instance which is being deployed for first time, choose **Create the database**. If this is a test instance, enter **yes** to add test data, otherwise do not enter ‘yes’ (just press Enter).

Note: If you are *updating* eScreening, make sure to enter “**NO**” to the question, “Should the current database be deleted and rebuilt?”.

At the end of the script any new or changed database scripts will be listed.

- i. If you receive the error “line 37: deploy.sh:command not found” see below.


```

$ ./deploy-sfc-prod.sh
bash: ./deploy-sfc-prod.sh: command not found

VHASFBCBOENAO@VHASFCAAPPMHE MINGW64 /d/escreeing
$ ./deploy-sfc-prod.sh
./deploy-sfc-prod.sh: line 37: deploy.sh: command not found

VHASFBCBOENAO@VHASFCAAPPMHE MINGW64 /d/escreeing
$ ./deploy-sfc-prod.sh

Current tag is: MHE_Prototype_061215
Getting latest code from branch: master...

remote: Counting objects: 48, done.
remote: Compressing objects: 100% (36/36), done.
remote: Total 48 (delta 17), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (48/48), done.
From https://github.com/VHAINNOVATIONS/Mental-Health-eScreening
   7fd2850..476cbb4  master    -> origin/master
   4451a5f..da8c942  resolved  -> origin/resolved
   = [new branch]    t1007     -> origin/t1007
Already on 'master'
Your branch is behind 'origin/master' by 1 commit, and can be fast-forwarded.
(use "git pull" to update your local branch)
Updating 7fd2850..476cbb4
Fast-forward
  escreeing/pom.xml | 11 ++++++++
  1 file changed, 11 insertions(+)

If there are no conflicts or errors, hit enter to build the new production packa
ge

Created tag sfc-prod_20151125_1708

Should the current database be deleted and rebuilt
*****
** WARNING ALL DATA WILL BE LOST **
*****
Enter 'Yes' to rebuild database and lose all data.

Yes

Should test data be inserted into the new database? (Enter 'y' to include)
n

Building WAR...

```

ii. Enter a `./` on line 37 of `deploy-sfc-prod.sh` to resolve this error.

```

37:  ./deploy.sh $tomcatInstance $profile $branch $jdbcUsername $jdbcPassword $vistaIP $vistaPort $vistaPrimaryStation $visaAccessCode
    $samplePatientIen "$refTbiServiceName"

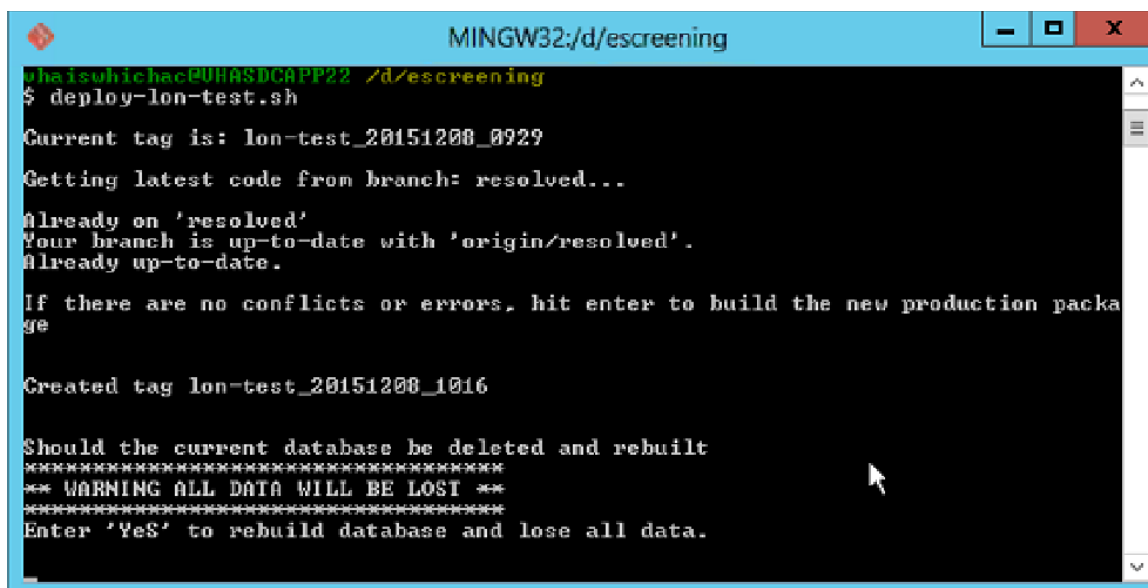
```

The after the edit the line 37 should read “`./deploy.sh $tomcatInstance $profile $branch $jdbcUsername $jdbcPassword $vistaIP $vistaPort $vistaPrimaryStation $visaAccessCode $vistaVerifyCode $vistaDUZ $vistaEncrypted $quickOrderIen $samplePatientIen "$refTbiServiceName"`”

4. After deploying the new eScreening version, if any database scripts were listed, take the following steps:
 1. Open the MySQL workbench.
 2. Connect to the correct MySQL instance (production or test).
 3. Choose the correct schema.
 4. Run each database script listed by the deployment script ordered by sprint directory.
5. Restart the Tomcat service for the updated instance.

2.5. Updating eScreening

To update the parameters provided in section 2.3.1 in the deploy shell script, follow instructions in section 2.4, step 3b, however, make sure that you enter “**NO**” to the question “Should the current database be deleted and rebuilt?”.



```
whaiswhicac@UHASDCAPP22 /d/escreeing
$ deploy-lon-test.sh

Current tag is: lon-test_20151208_0929

Getting latest code from branch: resolved...

Already on 'resolved'
Your branch is up-to-date with 'origin/resolved'.
Already up-to-date.

If there are no conflicts or errors, hit enter to build the new production package

Created tag lon-test_20151208_1016

Should the current database be deleted and rebuilt
*****
** WARNING ALL DATA WILL BE LOST **
*****
Enter 'Yes' to rebuild database and lose all data.
```

2.6. SSL Certificate

The eScreening domain must be joined to the VA domain and an SSL certificate issued.

Basic instructions:

<https://support.godaddy.com/help/article/4801/installing-an-ssl-certificate-in-microsoft-iis-7>

The SSL certificate expires one year from the issue date. Before the certificate expires, request and install a new certificate:

1. Create a certificate request from IIS manager, using the same parameters as the existing certificate.
2. Submit the certificate request to the portal:
<https://vaww.portal.va.gov/sites/PKI/Lists/SSLTLS%20Requests/AllItems.aspx>
The turnaround for the certificate is within a few days.
3. After receiving the new certificate, install it by clicking the **Complete Request** button.
4. Switch the default site's SSL certificate to the new certificate:
Click **Sites** → **Default Web Site** → **Binding**, and select the new certificate.

3. Routine Operations

There are two routine operations that must be performed on the system: user management and backing up the database. User management is performed within the administrative dashboard and consists of adding, editing, and removing users, and is covered in Section 3.2, Security/Identity Management. Backing up the database is performed at the operating system level and is covered in Section 3.1.3 Back-up & Restore.

Note: The system administrator role is an IT role, not an eScreening program role.

Operation	Role	Section
User management	Healthcare System Technical Administrator (HSTA)	3.2
Database backup	System Administrator	3.1.3

Table 4: Routine Operations

3.1. Administrative Procedures

3.1.1. System Start-up

The system does not require any regular manual start-up procedures. The database and application servers are both implemented as Windows services that automatically start with Windows, and it is unlikely that either service will fail under normal conditions. The services are listed below:

Service	Name	Display Name
MySQL	MySQL56	MySQL56
Tomcat	Tomcat7	Apache Tomcat 7.0 Tomcat7

Table 5: System Services

In the event that either does not start, or shuts down prematurely, they can be manually started via the Windows services snap-in. However, before starting either service, consult the Windows event viewer and individual service logs for information about the error. See Section 4, Exception Handling, for more details about error handling and logs.

3.1.2. System Shut-down

The system can be shut down by shutting down the two system processes from the Services snap-in. The services are listed in Section 2.1.1, System Start-up.

3.1.3. Back-up & Restore

Database backup and restore in eScreening are scripted in order to simplify maintenance. The backups are automated and run on a VA-configured frequency, whereas restores are done manually when necessary, as depicted in the image below:

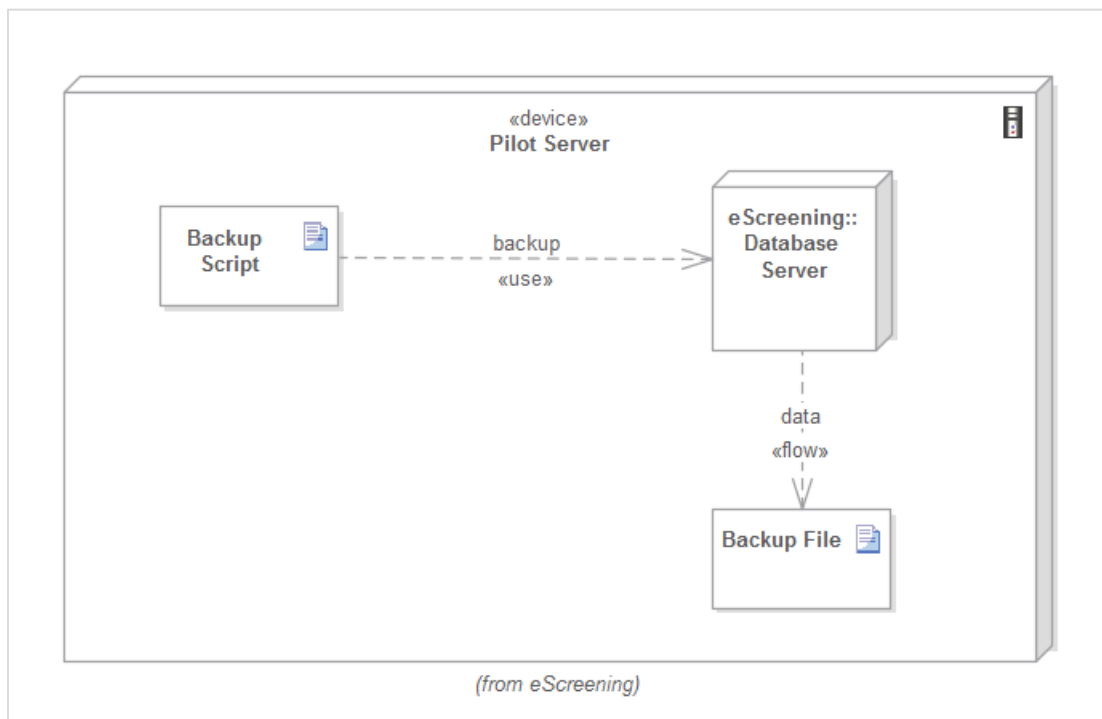


Figure 6: Database Backup Process

The only system component that requires backup is the eScreening database/repository. Backups are done to local storage, but it is expected that VA IT will move backups to external storage for safekeeping.

3.1.4. Back-Up Procedures

Database backup is automated via a Windows system task, a backup script, and a separate VA IT process to copy the backup files to remote storage. The Windows task runs nightly, but can be configured to run on any time interval. The backup process can be run during hours of operation, but it is recommended that it be run outside of operating hours in order to maintain the best user experience.

The following table reflects the backup schedule for the eScreening database:

Component	Backup schedule	Type	Copy to remote disk
eScreening database	Nightly	Full	By VA per VA standards

Table 6: Backup Schedule

The Windows task runs a script called *backup.bat*, which performs the database backup and copies the backup file to the folder specified by the BACKUP_FOLDER variable (default value: d:\data\backup). The BACKUP_FOLDER variable can be overridden on the command line by passing a value for the /BACKUP_FOLDER parameter. The backup script also contains a NUM_BACKUPS_KEPT variable for the number of backups to keep (default: 7). NUM_BACKUPS_KEPT can be overridden on the command line by passing in a different value for the /NUM_BACKUPS_KEPT parameter.

Each time the backup script runs, it creates a backup of the database in the folder specified by BACKUP_FOLDER with a filename of *escreening.yyyyMMddhhmmssfff.bak*. This pattern can be changed in the script file. After the file is copied, the backup script automatically removes the oldest backup files in excess of the number given in the NUM_BACKUPS_KEPT variable. The backup script output is logged in the d:\logs\backup folder in files named *backup.yyyyMMddhhmmssfff.log*. This log file also contains a hash of the newest key for each record in each table, which can be used later for validating backups, regardless of whether they are full or differential.

A full description of the backup command variables and parameters are as follows:

Variable	Description	Default value	Override parameter
BACKUP_FOLDER	Path to backup folder	D:\data\backup	/BACKUP_FOLDER
NUM_BACKUPS_KEPT	Number of backups kept	7	/NUM_BACKUPS_KEPT

Table 7: Backup Script Variables

By default, backup script performs a full back up each time. This is due to the relatively small amount of data. Should the data eventually grow past a VA-specified amount, VA can follow the instructions in the script file for creating differential backups instead.

The following diagram illustrates the process:

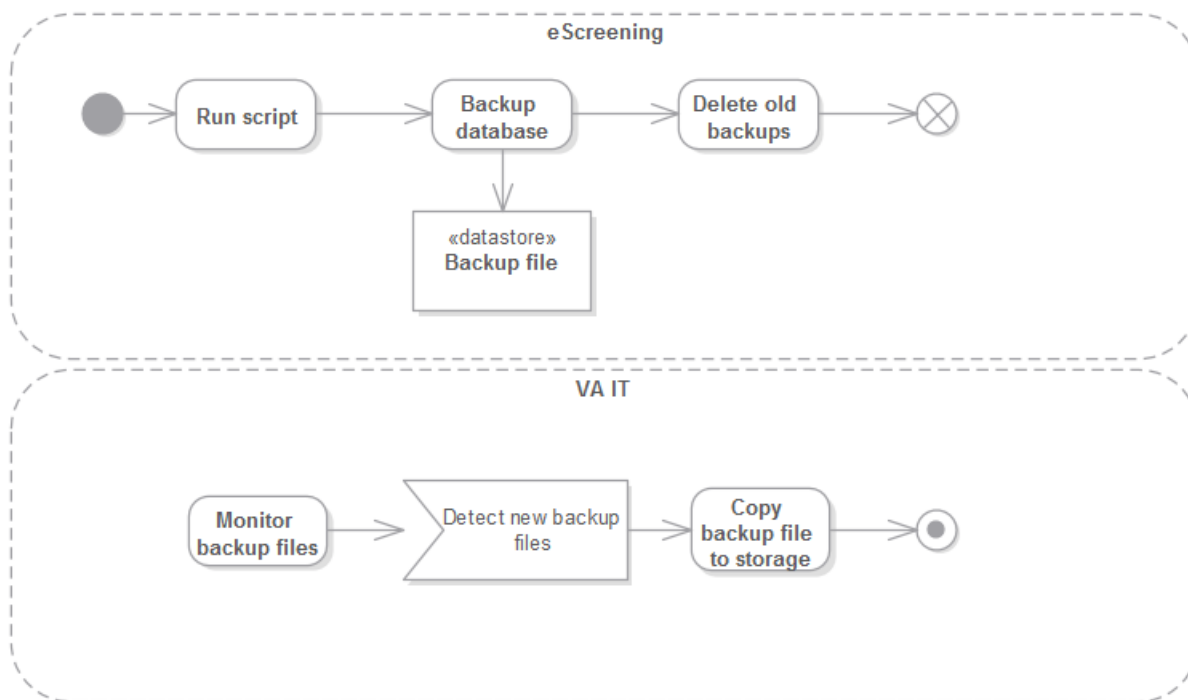


Figure 7: Backup Process

Note: It is recommended that the system administrator move backups off the eScreening server to a remote drive for safe keeping and to reduce disk space usage. A description of processes created by IT in order to safeguard backup files is beyond the scope of this manual.

Restore Procedures

Database backups can be restored by running the *restore.bat* script. The restore script takes one parameter: the fully-qualified path of the backup file to be restored. For example, the following command restores the database using the file *escreeening.2014.04.03.59.01.bak*:

```
restore.bat d:\data\backup\escreeening.2014.04.03.59.01.bak
```

Warning: Database restoration restores the database to a previous state. Changes made to the database after the most-recent backup and before the restoration runs will be lost.

The restore script is reentrant and will roll back the restoration if it encounters any errors. The output of the script is captured in the *d:\logs\restore* folder in files named *restore.yyyyMMddhhmmssfff.log*. The log contains a hashed key for the newest record in each table; this metadata can be used later to validate backups.

Back-Up Testing

Backup testing should be performed by VA IT on an external system on a regular basis as dictated by VA standards.

Frequency	Location	Process
VA specified	VA-specified external server	Comparison of backup and restore metadata

Table 8: Backup Testing Schedule

The backup script, as indicated in Section 3.1.3, Backup Procedures, emits the hash of the key of the newest record for each database table in the backup-specific log file. The restore script logs a hash of the newest in each table that it restores. VA can validate the backup file by restoring it to an external database server and then comparing the backup metadata to the restore metadata.

Step	Description
1	Note backup log metadata
2	Restore backup to external server
3	Compare restore metadata to backup metadata.

Table 9: Backup Validation Process

In the unlikely event that the restore and backup metadata do not tie, the IT system administrator should perform a detailed investigation in the current state of the database and the backup files as per VA guidelines.

Storage and Rotation

The VA system administrator should copy/move database backups to remote storage for safekeeping according to IT guidelines. The eScreening server provides RAIDed storage and a rolling backup system for safeguarding backups locally, but relocating backups to SAN or other storage per VA IT guidelines provides additional safety and redundancy. External/redundant storage by IT is beyond the scope of this document.

3.2. Security and Identity Management

The eScreening security architecture consists of components that perform authentication and authorization of VA staff and Veterans operating on the VA network via a client system over WIFI or VPN. These components include:

- Client device (tablet). Provides strong password and locks down access to only eScreening
- Network: Encrypted and secured communication between the tablet and the server over TLS
- Web application: Authenticated and authorized access to features
- VistA: Authenticated and authorized access to pull some basic data and upload data

The tablet has a strong password and can only be unlocked by program staff. The tablet runs in “kiosk” mode, limiting the user to only the eScreening application within the tablet web browser; no other web site can be accessed in the browser, and no application other than the browser can be accessed when in kiosk mode.

The following attributes describe the eScreening architecture as related to security:

- The eScreening system resides in the San Diego VA Medical Center and consists of a web application, web services, and a database
- Clinicians access authorized portions of the web application from VA facilities over VA WIFI using the clinicians’ credentials
- Veterans access authorized portions of the web application from VA facilities over VA WIFI from within a locked-down supervised mode session. Veterans input answers to assessment questions, and their answers are securely transmitted to the eScreening server in the VA data center.
- eScreening reads limited patient identification and demographics data from VistA, and writes assessment results to VistA
- eScreening integrates with VistA via VistALink entirely on the VA network
- Staff use CPRS to view/sign assessment notes, maintain patient record

For more information on eScreening security, see the System Security Plan.

3.2.1. Identity Management

Users are added, modified, or deactivated through the administrative dashboard user interface. In the user interface, the Healthcare System Technical Administrator user has the ability to create users, assign their access (add them to program locations), and deactivate them. Adding a user consists of using the *create user* form to fill in the new user’s name, phone number, email address, and other attributes. Modifying a user consists of using the *edit user* form to modify values. Deactivating consists of changing the user’s status to *inactive*.

Note: Users are not deleted in eScreening; they are simply inactivated and they can be reactivated in the future as needed.

Activity	Location	Interface
Add user	Administrative dashboard	Add User form
Edit user	Administrative dashboard	Edit User form
Deactivate user	Administrative dashboard	User Status field

Table 10: Identity Management Functions

For more information on user management, see the Administrator Training Manual, which details the Healthcare System Technical Administrator’s tasks.

3.3. User Notifications

The user community will be notified of any scheduled changes via email distribution lists. It is recommended that separate mailing lists should be established for users, program administrators, and support staff.

3.4. System Monitoring, Reporting, & Tools

System monitoring should be performed using VA's enterprising monitoring suite. Probes should be established for operating system CPU, memory, disk space, and the Tomcat and MySQL processes.

3.4.1. Availability Monitoring

Probe the status controller regularly (for example, every 10 minutes) for the application's availability. This keeps the Java VM warm and allows the monitoring tool to test the status of system components such as the database and VistA connectivity.

Concern	Test
Web application	Application: OK
Database connectivity	Database: OK
VistA connectivity	VistA: OK

Table 11: Application Status Checks

Loading the status screen checks the application status in general, as well as the database and VistA connectivity. In addition to automated monitoring, this screen can be checked manually to determine the status of the system after a deployment or patch, or during troubleshooting.

3.4.2. Performance and Capacity Monitoring

eScreening performance and capacity management consists of two concepts: verifying system performance through page loads and log analysis, and verifying capacity through disk and network analysis.

There are three stated KPIs for eScreening: initial page load ≤ 15 s, subsequent page load ≤ 3 s, and individual assessment upload ≤ 5 s. These KPI are summarized below:

Action	Threshold	Verification
Initial page load	15 seconds	Manual
Subsequent page load	3 seconds	Manual
Assessment upload	5 seconds	Log analysis

Table 12: Performance Thresholds

Page performance verification is currently a manual process performed by the system administrator as per VA guidelines. Ongoing page performance analysis can be performed by scraping the application server logs for page response times. If desired, the system administrator can compute averages and percentiles. The logs can be exported to VA's enterprise log analysis system as needed.

eScreening disk and network capacity can be assessed by the system administrator or NEDIIS per VA guidelines. Free space can be queried via VA's enterprise monitoring tool (e.g. SolarWinds, etc.). Network link capacity can be accessed via ongoing link analysis via the network OSS team or NEDIIS.

Element	Procedure	Actor
Disk space	Disk free probe	System administrator
Network links	NetScout ongoing analysis	NEDIIS
30 concurrent users/site	Log analysis	System administrator

Table 13: Procedures for Monitoring Capacity

For more details on disk or link analysis, see VA guidelines.

3.4.3. Critical Metrics

The critical metric for eScreening is whether 30 concurrent users can simultaneously use the system at a single VA site. The upstream implication of a failure to support that level of concurrency is a possible delay in performing screening for some Veterans. The downstream implication is a possible delay in identifying or seeking treatment for some Veterans. The critical metric is summarized below:

Metric	Threshold	Upstream implications	Downstream implications
Concurrent users/site	30	Delay or errors performing assessments	Delayed identification of health issues

Table 14: Critical Metrics for eScreening

The system's current or historical support for concurrent users/site can be assessed by exporting the log files to VA's enterprise log analysis service. The logs have an industry standard structure that will be recognized without custom parsing by most commercial or open source log parsing tools. Adherence can be determined by comparing page requests times and error counts against concurrent logins.

3.5. Routine Updates, Extracts, and Purges

Updates, extracts, and purges are performed for eScreening per VA guidelines and as requested by program administrators. These activities are summarized below:

Activity	Periodicity	Responsible party
Updates	As needed	DBA
Extracts	As needed (monthly?)	Scripted by DBA, run by specified individuals
Purges	As needed	DBA

Table 15: Routine Data Activities

Updates consist of inserting or updating data in the database can be performed as needed. Updates should be scripted with sufficient error handling and rollback logic to handle expected and unexpected errors during execution while protecting data integrity. Examples of scriptable updates include changes to health factors, program data, or Veteran data. Data changes require expertise in SQL and the eScreening schema (see project schema document). Updates should be performed by qualified DBAs as requested by eScreening program coordinators.

Extracts consist of exporting data for analysis. It is expected that OIA will periodically (perhaps monthly) extract Veteran assessment data or metadata for use in external systems. Extracts can be performed via the eScreening user interface or via SQL scripts against the database. Scripting extracts requires knowledge of the eScreening schema and should be performed by qualified DBAs. After the script is created by the DBA, it can be run by authorized individuals with shell-level access to the system as specified by program administrators.

Purging consists of deleting or tombstoning data in the database via SQL scripts. Purging requires knowledge of the eScreening schema and should be performed by qualified DBAs. Purging requires authorization by program administrators.

Warning: Purging removes data from the system and should only be performed after taking a database backup and via express authorization of program administrators.

3.6. Scheduled Maintenance

Scheduled maintenance will be performed as authorized by program administrators. Currently, there is no schedule for maintenance.

3.7. Capacity Planning

Capacity planning should be performed by VASD IT in cooperation with eScreening program administrators. Currently, there is no schedule or requirements for capacity planning.

4. Exception Handling

Runtime errors in eScreening are typically related to configuration, connectivity, or data issues. Errors related to connecting to the eScreening database, configuration, and bad or unmatched Veteran data can be resolved locally by the system administrator. Other kinds of errors, such as problems connecting to VistA can be resolved through cooperating with external teams. The types of errors are summarized below:

Type	Examples
Locally resolvable	Unmatched records, bad data, DB connectivity
Externally resolvable	Network or VistA issues
Unresolvable	Errors due to bugs

Table 16: Types of Errors

Note: Some errors, such as those due to unidentified bugs, require application source code changes and cannot be changed by the system administrator.

4.1. Routine Errors

Like most systems, eScreening may generate a small set of errors that may be considered routine in the sense that they have minimal impact on the user and do not compromise the operational state of the system. Most of the errors are transient in nature and only require the user to retry an operation. The following subsections describe these errors, their causes, and what, if any, response an operator needs to take.

While the occasional occurrence of these errors may be routine, getting a large number of an individual error over a short period of time is an indication of a more serious problem. In that case the error needs to be treated as an exceptional condition.

4.1.1. Security Errors

Security errors in eScreening consist of authentication or authorization issues related to accounts. For example, if a staff user attempts to logon to eScreening with invalid credentials three contiguous times, the system will display an error message directing the user to see the clerk.

Likewise, if a staff user is already authenticated but attempts an unauthorized action in VistA, the system will display an error message. This can occur if the user attempts an operating that requires querying VistA, such as setting up a Veteran for a screening or uploading Veteran assessment data after a screening. If a user enters an invalid access/verify code combination and eScreening cannot authenticate and authorize the user, eScreening will display a message indicating that the user seek assistance from the clerk.

The following table displays the known security error types, descriptions, and resolutions:

Type	Description	Resolution
Staff authentication failure	When a staff user does not have a valid account or does not use a valid password, the user will be unable to login and an error message will display instructing the user to see the clerk.	Ensure user is active and correct in eScreening.
Staff authorization failure	When a staff user attempts to access an application feature to which he/she does not have access (through URL manipulation), the site directs the user to a help page instructing the user to see the clerk.	Ensure user has correct entitlements for the job he/she is performing and trained on system use.
Veteran data lookup/upload failure	When a staff user does not enter the correct access/verify codes, the system will be unable to authorize the user's lookup/upload action in VistA, and an error will display.	Ensure user is using correct VistA access/verify codes.

Table 17: Security Errors in eScreening

Note: If a Veteran attempts to take an assessment without known credentials (last name and last four SSN digits) the system will treat the issue as being a data condition instead of an error condition. eScreening is designed to handle new Veterans who do not have VistA records. Because the system cannot know whether the Veteran user is new to VistA or simply mistyping his credentials, the system will treat the Veteran as a new VHA patient. eScreening staff will then be responsible for addressing the Veteran's status. If the Veteran already has a VistA record, staff will be able to *map* the eScreening record to the VistA record. On the other hand, if the Veteran is truly new, VA will need to create a new VistA record for the Veteran, and then map his or her eScreening record to the new VistA record.

4.1.2. Time-outs

In eScreening, timeouts can occur between the client and the server, and between the server and VistA. Timeouts can be due to capacity issues with regard to the eScreening server, the VistA server, or the network fabric in between. It is expected that most timeouts will be due to capacity or contention issues caused by the tablet communicating with the server over VA WIFI, not on the server itself or between the server and VistA. When any part of the system times out, the application displays a user-friendly error message indicating that the user should talk to the clerk. The following table summarizes the types of possible timeouts:

Type	Incidence	Response
Timeout connecting to VistA	Unknown	Repeat attempt or file support ticket with VistA or NOSS group.
Timeout uploading data to eScreening	Unknown	Try again, troubleshoot server, or file support ticket with NOSS

Table 18: Possible eScreening Timeouts

Most timeouts will be transient in nature, and resolve after the network or server contention abates. However, timeouts can also be investigated and submitted to the appropriate support groups. Some timeouts between the server and client can be logged, timeouts on the server itself,

and timeouts between the server and VistA are logged on the server. This allows the system administrator to investigate individual timeout issues as well as use system tools or external tools in order to investigate patterns of timeouts.

For example, the system administrator can see the most-recent errors by tailing the log in PowerShell:

```
gc d:\data\logs\logFileName.log -tail 100
```

Likewise, the admin can search all log files for timeout errors:

```
dir d:\data\logs\*.log | select-string "connection refused"
```

Timeouts tend to be sporadic, based on transient network or server conditions. However, the system administrator can analyze the logs in VA's enterprise log analysis utility for greater insight into trends.

4.1.3. Concurrency

Concurrent updates can lead to unpredictable errors in any system, including eScreening. However, due to the nature of eScreening, concurrency issues are very unlikely to occur. If they did occur, they would be related to very rare events like multiple staff attempting to update a Veteran's record or upload a Veteran's assessment. In the case of concurrent updates to Veteran or assessment data in eScreening, the system will note if one user is updating an old version of the data and prompt the user to view the updated record and possibly try again.

The table below summarizes the type of possible concurrency issues that could occur:

Activity	Incidence	Response
Simultaneous updates to modules in the forms editor	Very rare	System will catch and log error, then prompt user to (optionally) try again.
Simultaneous updates of patient data	Very rare	System will catch and log error, then prompt user to (optionally) try again.
Simultaneous uploads of assessment data to eScreening	Very rare	System will catch and log error, then prompt user (optionally) to try again.
Simultaneous uploads of assessment data to VistA	Very rare	System will catch and log error on the later attempt.

Table 19: Possible Concurrency Issues

In the case of multiple concurrent uploads of assessment data to VistA, eScreening will throw an exception internally when it sees the redundant request, which the UI will catch and display an error message to the user that says the record has already been uploaded.

4.2. Significant Errors

Significant errors can be defined as errors or conditions that affect the system's stability, availability, performance, or otherwise make the system unavailable to its user base. The following subsections contain information to aid administrators, operators, and other support personnel in the resolution of errors, conditions, or other issues.

4.2.1. Application Error Logs

eScreening logs are, by default, all kept within d:\data\logs*. A subfolder should exist for each component that does logging; collocating the logs this way simplifies finding and querying the logs.

Logging for each component is configurable at the component level. The log configuration files are stored in these folders:

- Application: d:\apps\tomcat7\webapps\escreening\WEB-INF\classes\log4j.xml
- Tomcat: d:\apps\tomcat7\conf\logging.properties
- MySQL: d:\apps\mysql\MySQL Server 5.6\my-default.ini
- eScreening application log is located in:
d:\apps\tomcatinstances

The log files for the eScreening application are found under:

d:\apps\tomcatinstances\<instance-name>\logs\es_web_app_<instance_name>.log

The instance name, i.e., <instance_name> represents sdc-prod, lon-prod, sfo-prod, etc. Replace the <instance_name> with the appropriate instance names.

The log files are configured to create a new log file with a file extension after every log file reaches the size of 10 MB. For more details about the setup of the log files, go to:
<https://logging.apache.org/log4j/1.2/apidocs/org/apache/log4j/RollingFileAppender.html>

Logging is configured by the system administrator. Sensible defaults are supplied along with the application, allowing adequate log coverage for troubleshooting without affecting performance or taking up excessive disk space. The following table outlines the key logging attributes:

Type	Location	Max size	Growth rate	Rotation suggestion	Retention suggestion
Application logs	d:\data\logs\application	100 MB (suggested)	Varies	Daily	10 days
MySQL logs	d:\data\logs\mysql	As configured	Varies	Daily	10 days

Type	Location	Max size	Growth rate	Rotation suggestion	Retention suggestion
eScreening application logs	<p>\apps\tomcatInstances\<instance-name>\logs\esc_web_app_<instance_name>.log</p> <p>For example, if the instance_name of escreening application is sfo-prod. At any given point the loggable information could be found under</p> <p>d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.1 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.2 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.3 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.4 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.5 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.6 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.7 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.8 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.9 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.10 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.11 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.12 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.13 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.14 or d:\apps\tomcatInstances\sfo-prod\logs\esc_web_app_sfo-prod.log.15</p>	10 MB	Varies with traffic	After every file reaches 10 MB	Do not touch. The system automatically starts over on the first log file after 150 MB.

Table 20: eScreening Logging

Note: These values should be adjusted by the system administrator based on VA guidelines instead of kept at their default levels.

Querying and analyzing the log files is simple because they are text files and use the industry-standard log4j log conventions (INFO, WARN, ERROR, etc.) or built-in Java logging (WARN, SEVERE, etc.) conventions. PowerShell or other Windows shell utilities can be used to query the files. Some examples are listed in the following table:

Activity	Example
Find all errors	select-string "ERROR SEVERE" *.log
Find warnings in last 100 lines of a file	gc tomcat7-stdout.2014-01-10.log -tail 100 select-string WARN

Table 21: Example Log Queries

For more detailed log file analysis, the system administrator can import the log files with VA's enterprise log analysis tool.

4.2.2. Dissection of error message

Each Log message follows the following format:

`[%p] %d{dd/MM/yyyy HH:mm:ss} [%t] [(%C{1}:%L)] %m%n`

The above format is described as follows:

`[%p]` ==> [Priority (DEBUG, INFO, WARN, ERROR)]

`%d{dd/MM/yyyy HH:mm:ss}` ==> DATE/MONTH/YEAR HOUR:MIN:SS

PS: Please note that the time stamp is DATE followed by MONTH and not MONTH followed by DATE as we are accustomed in United States of America

`[%t]` ==> [NAME OF THREAD]

`[(%C{1}:%L)]` ==> [(fully qualified class name of the caller issuing the logging request:line number from where the logging request)]

`%m%n` = application supplied message associated with the logging event and line separator character

4.2.3. Application Error Codes and Descriptions

eScreening does not currently use error codes; rather, it defines custom exception classes that can be used for structured exception handling. These classes can be reused across a family of issues. The table below describes the existing custom application types and descriptions:

Type	Description
AssessmentEngineDataValidationException	Error validating assessment data
AssessmentHadUnexpectedNumMappedTemplatesException	Internal error
AuthenticationException	Error authenticating user.
BadPasswordException	User entered bad password
BadUseridException	User entered bad user id

Type	Description
CellDoesNotMatchColumnException	Internal error
CouldNotResolveVariableException	Internal error
CouldNotResolveVariableValueException	Internal error
EmptyDataExportException	User attempted to export
EscreeningDataValidationException	Error validating user-entered data
InvalidAssessmentContextException	Error authorizing Veteran
ReferencedFormulaMissingException	Internal error
ReferencedVariableMissingException	Internal error
TemplateProcessorException	Internal error
UnregisteredDataTableColumnException	Internal error

Table 22: Existing Custom Application Types and Descriptions

4.2.4. Infrastructure Errors

eScreening relies on various infrastructure components and must handle temporary failures in those components when they occur.

Database

eScreening can experience errors connecting to the database or performing data operations. Because the database currently resides on the same server as the application, the most likely cause of database connectivity failures is unhandled exceptions around database connections. These types of errors are not very likely because the system uses an ORM to handle connections, but if they do occur, they will most likely be transient. Database connection errors can be found in the logs by querying for “connection” and orphaned connections can be queried and forced close via MySQL commands. For more information on querying and force-closing orphaned connections, see the MySQL online manual: <https://dev.mysql.com/doc/refman/5.6/en/index.html>

The application can experience errors performing data operations as well. This includes errors querying, inserting, updating, or deleting data. When these types of database errors occur, the application will catch the exception and log it. If the error is something that the user can fix by trying again, the application will display a message to the user; otherwise, the application will handle the error itself and may direct the user to a user-friendly error page based on the severity of the error.

Web Server and Application Server

Tomcat automatically logs all errors to the *stderr* and *stdout* files, although the system administrator can configure the logging per VA guidelines. Errors are denoted in the logs by severity (e.g., “SEVERE”).

By default, Tomcat uses Java logging; however, for the system administrator can easily configure Tomcat to use log4j instead as per VA conventions. For more information on Tomcat logging, see the online manual <http://tomcat.apache.org/tomcat-6.0-doc/logging.html>

The eScreening web application is currently configured to do its logging through Tomcat. The system administrator can configure application-specific behavior in the log4j.xml file.

Network

eScreening can suffer from errors due to network conditions between the client and the server, or between the server and VistA. If there are network problems during the initial loading of a page, the client may display built in error messages (e.g., HTTP 404). On the other hand, if there are errors transmitting data in the background, the client JavaScript will attempt to retry the operation before failing with a user-friendly error message.

For network issues between the server and VistA, if the application can catch and retry the operation, it will. For network errors beyond the application’s grasp, the server will fail and log the operation and redirect the user to a user-friendly error page. The error page typically instructs the user to see the clerk.

Authentication & Authorization

All authentication and authorization errors are caught by the application and logged.

For eScreening authentication errors, the system will prompt the user a total of three times and then redirect the user to an error page instructing the user to see the clerk. eScreening authorization errors should be rare, but if they occur, the user will be directed to the page instructing the user to see the clerk. The clerk can adjust the user’s settings as needed.

For VistA authentication issues, the system will prompt the user several times before redirecting the error page. For VistA authorization issues, the application will direct the user to the error page after the first failure. The clerk can then coordinate with external VA resources to resolve the user’s VistA access issues.

4.3. Dependent System(s)

eScreening is dependent upon VistA for authorizing Veterans and uploading Veteran assessment data. For persistent failures connecting to VistA or performing VistA operations, the system administrator should verify with VistA support resources the VistA connection information in:

```
.\WEB-INF\classes\gov.va.med.vistalink.connectorConfig.xml
```

Changes to the VistA configuration file will be picked up on subsequent requests. When changing the configuration, set the *encrypted* flag to “false.” This will cause the system to encrypt the connection information and then set the flag back to “true.”

```
<connector jndiName="vljtestconnector" ip="54.235.74.13"
```

```

port="8000" primaryStation="500" access-
code="SGll8EpcOHZe9oXgwuPsFg=="
verify-code="JQ9S3/VDJAQJO39bZVcqP8q3W8JSIxt9" encrypted="true"
enabled="true" timeout="15" always-use-default-as-min="false" />

```

Other than configuration errors, there is nothing that can be done to resolve VistA access or connectivity issues within eScreening; all other errors must be resolved in cooperation with VistA support.

4.4. Troubleshooting

Troubleshooting eScreening issues consists of checking the logs and tweaking configuration settings. Most application behavior cannot be adjusted without modifying code. The following table summarizes the types of errors and resolution procedures likely to occur in eScreening:

Type	Procedure
Errors	Check logs and report issue.
Database connectivity issues	Check status page, application and database logs and connection string. Troubleshoot using MySQL CLI.
Other database issues	Check logs and report issue.
VistA connectivity issues	Check logs, report issue to appropriate help desk.

Table 23: Troubleshooting eScreening

The first step in most cases is to check the system status page. The process of authenticating and viewing the status page will give you some information about the system stability, because this process exercises the application, database, and VistA. The inability to authenticate or errors reported on the status page allows the system administrator to narrow his/her focus.

The next step is to check the logs. The logging level can be temporarily dialed up in each logging configuration file (see 4.2.1) to support DEBUG-level messaging as needed. The logs will display detailed information about the type of problem that is occurring, and can be tailed and searched. If the application is operational in general, failing actions can be tested in the application and checked in the logs.

If the application cannot connect to the database, the system administrator can check whether the *mysqld* process is running, check the MySQL logs, and test connecting to the database using various parameters using the MySQL CLI (command line interface). The CLI can also be used to query and modify data or state as needed in order to resolve the issue.

If the problem is with VistA, the connection information can be changed or confirmed with VistA support technicians. Most VistA issues will require cooperation with VistA support.

Finally, if the issue lies within the application itself, such as a bug or the inability to deal with an unforeseen issue in the production environment, the application source code can be modified as needed to resolve the issue.

The following tables provide a detailed listing of error conditions and resolution actions.

Category	Description	Actions
Verify VistA Account in the MyAccount Page	User is presented with "Account is locked. Failed to verify Access/Verify codes"	User's account is locked in VistA. User must close their browser, start a new session, and then retry. If the account is still locked, then the user needs to contact the VistA System Administrator to resolve the locked account issue.
Verify VistA Account in the MyAccount Page	User is presented with: "Another account has already verified this Access/Verify codes"	Another user has already verified their VistA account and is using the DUZ associated with the verify code. Sys admin should use CPRS to look up the user's DUZ, search the MySQL USER.VISTA_DUZ field to determine who is using it, and then determine which user owns that DUZ. If the existing user made a mistake, then the system administrator should NULL out the USER.DUZ field and set USER.CPRS_VERIFIED to "0"
Verify VistA Account in the MyAccount Page	User is presented with: "Invalid Access/Verify codes. Please see your admin. Too many attempts to verify a VistA account will lock your account."	User has entered the wrong Access/Verify code more than 2 times. User should seek assistance with a VistA admin to ensure their Access/Verify code is correct or have it be reset to something else.
Verify VistA Account in the MyAccount Page	User is presented with "Failed to connect to VistA" and the application log file will contain "VistaSocketException" entry.	See "VistA Connection Issues" below
Staff Login	Nobody is able to log in to the system despite providing valid credentials. This can happen when the database is down and the server cannot access the database to verify a user's credential. The application log file entry will have a "CannotGetJdbcConnectionException" exception.	See "Database Connection Issues" below
Select Veteran Page in the Create Battery Tab	After clicking on Search, the application spins for a minute and then redirects to the System Exception page. The application log file entry will have "VistaSocketException" entry.	See "VistA Connection Issues" below
Import Data Page of the System Configuration Tab	After clicking on one of the Import buttons, user is presented with "An unexpected error occurred while trying to import Clinical Reminder list from VistA." The application log file entry will have "VistaSocketException" entry.	See "VistA Connection Issues" below
Veteran Login	If veteran doesn't include Last name a form error is shown: "Last name is required"	Veteran should be directed to give last name
Veteran Login	If veteran doesn't include 4 numbers for SSN	Veteran should be directed to give valid last

Category	Description	Actions
	field a form error is shown: "The last 4 SSN is required"	four of social security number
Veteran Login	Veteran is presented with "Unable to connect" page after trying to log in.	See "Unable to Connect" below
Veteran Login	If Veteran is show form error: "Last name / Last 4 SSN were not found, please try again." Veteran has entered an incorrect combination of last name and last four of SSN.	Log into eScreening as a VA staff member. Search for the veteran to make sure there is an account. If not, then an assessment should be created with the veteran's credentials. If the veteran does have an account, verify the credentials the veteran is using. If the credentials are incorrect in eScreening, go to 'create battery' find the veteran's battery, click on select to select the battery, and then from the 'veteran detail' page, click on 'map to vista record' or 'refresh from vista'. If the credentials are still incorrect they CPRS must be used to update the VistA record for the veteran
Veteran Login	Veteran is presented with a page titled "Please See A Clerk for Assistance" with message "For assistance, please contact the Help Desk". This happens when the veteran does not have a battery in either the Clean or the Incomplete state. This can happen if no battery was assigned to the veteran or if the veteran has completed the battery.	Log into eScreening as a VA staff member. Navigate to "create battery" and create the correct battery for the veteran's appointment.
Veteran Login	Could not communicate with the database. Please try again and if the problem persists, notify the clerk.	See "Database Connection Issues" below
Veteran Assessment	During the taking of a battery, veteran is shown a dialog with title "Server Error!" and message "Unable to connect. Please see support staff for assistance." This occurs when the veteran's device is unable to connect to the eScreening server.	See "Unable to Connect" below
Veteran Assessment	After answering a question a banner error with the text "Error, Please contact support" shows up. This is shown because the veteran answered a question which may have follow-up questions and the veteran's device is unable to contact the eScreening server.	See "Unable to Connect" below.
Veteran Assessment	During the taking of a battery, veteran is shown a dialog with title "Server Error!" and message "Unable to process submitted data." or "Submitted data could not be processed". This indicates that the client application running in the veteran's browser is submitting invalid data to the eScreening server.	Baring malicious intent, this is most likely a bug and should be reported to the developer team. A code is sent to the veteran's device and can be shown if the Details link is clicked in the dialog. This will provide an error code that can be used by the system administrator to find the correct place in the eScreening system log. The eScreening system log should be inspected to look for the veteran's submission by the system administrator. Then the log segment tracking this submission should be sent to the developer team.

Table 24: Errors and descriptions

The following table shows actions for common errors:

Category	Actions
Unable to Connect	The server cannot be connected to from the device the Veteran is using. The network used by the Veteran should be checked to make sure it has a network connection (e.g. navigate to a page within the VA's network). If the device can navigate to another VA site then the server should be checked to make sure it is still running. If it is then IT should be contacted to ensure that the network hasn't become fragmented.
Database Connection Issues	As a system administrator, log into the web server and ensure the web server can communicate with the database, ensure the database is running by checking the Windows service list, and ensure the database account the web site is using is able to be used to log into MySQL.
VistA Connection Issue	As a system administrator, log into the web server and ensure the web server can reach the VistA server. Contact the VistA administrator to verify the Proxy Connection account is still valid. This can be found in the following file \$(rootwebsite)/WEB-INF/classes/gov.va.med.vistalink.connectorConfig.xml, If the access code or the verify code needs to be updated, update the fields, set the connector attribute of encrypted="false", and then restart Tomcat. This will trigger VistA Link to encrypt the fields and save it back into the XML file.

Table 25: Actions for common errors

4.5. System Recovery

The following subsections define the process and procedures necessary to restore the system to a fully operational state after a service interruption. Each of the subsections starts at a specific system state and ends up with a fully operational system.

4.5.1. Restart after Non-Scheduled System Interruption

If the system crashes or is brought down, it can be simply restarted by restarting the database and Java server processes and then viewing the application status page. The two processes, which are covered earlier in this document, are run as Windows services and can be started from the Services snap-in. Once the services start, the system administrator can log on to the application and view the status page in order to verify connectivity. The full steps are:

1. Ensure the MySQL and Tomcat services are running.
2. Load the application home page.
3. View the status page to ensure the application can connect to VistA.

If the Windows services do not start properly, the Event Viewer and the log files for each service can be checked for errors. Failures in the services are unlikely to happen, however, if the operating system itself is healthy.

4.5.2. Restart after Database Restore

The system can be restarted after restoring from a database backup by simply accessing the application. If the application server was taken offline in order to prevent access to the database

during the restore, the application server should be restarted as well before utilizing the application.

4.5.3. Back Out Procedures

The upgrade back out procedure consists of notifying the service desk, taking the application offline such that users see a “down for maintenance” page, performing the back out steps (restoring the database, redeploying the old version of the application, etc.), checking the application locally, restoring service, and then notifying the help desk that the maintenance is over.

Notifying the service desk should be done in accordance with the policies established at the time. (Currently, there is no service desk for supporting eScreening.) This may take the form of an email, call or service desk ticket.

Taking the application offline consists of changing configuration so that all requests except those passing in a special maintenance parameter (established by the system administrator) are routed to a simple HTML page that contains the designated “down for maintenance” message instead of hitting the web application.

After all existing requests to the web application have ended the old version of the database or application can be restored as per the back-out plan. This configuration can be verified by using another maintenance parameter to access the application and view the status page.

Once the back-out has completed, the routing expression on the server is restored to point to the desired version of the web application. The full steps are given below:

1. Notify the Service Desk about back-out plan initiation via email or service ticket
2. Disable user access to the system by engaging the maintenance mode routing expression
3. Restore backup taken before the change implementation by following the database restore procedures in this document
4. Conduct system health checks by utilizing the maintenance mode health check parameter and viewing the status page
5. Enable user access by disengaging the maintenance mode routing expression
6. Notify the Service Desk of successful back out

For more information on how to configure the maintenance routing expressions, see the Tomcat manual: <http://tomcat.apache.org/tomcat-7.0-doc/index.html> .

5. Operations & Maintenance System Support

An understanding of how eScreening is supported by various organizations within the VA is important to operators and administrators of the system. If you are unable to resolve an issue, then it is necessary to understand how to obtain support through OI&T's system support organizations. The following sections describe the support structure and provide procedures on how to obtain support.

5.1. Support Structure

This section describes the systems support structure as seen from the perspective of operations personnel. The first section defines the support hierarchy through which a support request may navigate. The second section defines the responsibilities for each level of support.

5.1.1. Support Hierarchy

There will be two levels of production support for eScreening until the application achieves nationwide deployment. The first level will consist of triage, account management, and basic troubleshooting performed by a Healthcare System Technical Administrator (HSTA). The second level will consist of application code and database change management as described within the eScreening Change Management Guide.

Following nationwide deployment, it is expected that the application will migrate to standardized VA support and change management practices, with tier 1 support performed by the National Service Desk, tier 2 support performed by VA regional IT support staff, and tier 3 performed by application developers as designated by eScreening program management.

5.1.2. Division of Responsibilities

This section defines the scope and responsibilities of each support tier.

5.2. Support Procedures

The eScreening support procedures will consist of triage, troubleshooting, and change management.

1. Defect and change requests triaged by Program Administrator
2. Troubleshooting by Healthcare System Technical Administrator
3. Change management performed by application developers as authorized by Change Control Board

1. Triage: The Program Administrator will collect and triage application defect and change requests from users. These requests will be entered in the eScreening change management backlog in the form of trouble tickets.

2. Troubleshooting: The program administrator will assign trouble tickets to the Healthcare System Technical Administrator, who will analyze, troubleshoot, and document the reported issues. If the HSTA can resolve the issue through at the configuration or database level, or through coordination with the National Service Desk (in the event of a CPRS or VistA issue), the HSTA will document the resolution within the ticket and mark it resolved.

3. Change management: If an HSTA is unable to resolve an issue without modification of the application source code, the HSTA will change the ticket state to needing Change Control Board (CCB) review. The CCB, which will consist of the VA PM, Program Administrators, Healthcare System Technical Administrators, and designated VA IT/support staff, will prioritize and assign all application change requests to designated application developers. The application developers will estimate the amount of time needed to complete the work associated with the ticket, and the PM will allocate the ticket to a specific development sprint. After the application development team completes and tests the work, they will mark the ticket resolved and perform the application release as authorized by the CCB.

For a detailed explanation of change and defect management, see the eScreening Change Management Guide.

Appendix: Setting Up Your Development Environment

1. Install Maven:

Download Maven 3.1.1

<http://maven.apache.org/download.cgi>

- a. Unzip the file and copy it to where ever you like, for example:
C:\Users\somebody\apps\apache-maven-3.1.1
- b. Create an environment variable
MAVEN_HOME=C:\Users\somebody\apps\apache-maven-3.1.1
- c. Add the bin folder to the PATH if you want.

%MAVEN_HOME%\bin

2. Install VA VistALink JARS to the local machine.

- a. Download the artifact from VA:
https://downloads.va.gov/files/FOIA/Software/Patches_By_Application/XOBV-VISTA%20LINK/XOBV_1_6/

If this link is not working, download the zip file (VistALinkJars) from the eScreening SharePoint Site in San Diego (<http://vaww.sandiego.portal.va.gov/eScreening>) which is located [here](#).

- b. Unzip the VistALink zip file.
- c. Open up a command prompt and CD to the folder:
(For example: C:\Users\somebody\Desktop\vlj-1.6.0.028\samples-J2SE)
- d. Manually install the three JARS by entering these commands:

```
mvn install:install-file -DgroupId=gov.va.med.vistalink -  
DartifactId=vljConnector -Dpackaging=jar -Dversion=1.6.0.028 -  
Dfile=vljConnector-1.6.0.028.jar
```

```
mvn install:install-file -DgroupId=gov.va.med.vistalink -  
DartifactId=vljFoundationsLib -Dpackaging=jar -Dversion=1.6.0.028 -  
Dfile=vljFoundationsLib-1.6.0.028.jar
```

```
mvn install:install-file -DgroupId=gov.va.med.vistalink -  
DartifactId=vljSecurity -Dpackaging=jar -Dversion=1.6.0.028 -  
Dfile=vljSecurity-1.6.0.028.jar
```

3. Install [Git client](#).

4. Install JDK from Oracle:

Latest JDK8:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

5. Install Spring Tool Suite (STS)

<http://spring.io/tools/sts/all>

STS 3.7.0.RELEASE

If you installed the x64 JDK, then make sure you download the 64 bit version. You can download the zip or the executable.

6. Create a new project with code from GitHub:

- a. Go to: **New -> Project...**
- b. Select: **Maven -> Check out Maven Projects from SCM.**
- c. Click **Next**.
- d. Set SCM drop-down to Git (If you do not have Git in the list, click the link **m2e Marketplace** in the lower right corner, then find and install **m2e-egit**).
- e. Set SCM URL to:

<https://github.com/VHAINNOVATIONS/Mental-Health-eScreening.git>

You can also use this if you have setup SSH keys:

`it@github.com:VHAINNOVATIONS/Mental-Health-eScreening.git`

- f. Click **Finish**.
- g. Wait (this takes a bit). You will see a dialog: Discover and map Eclipse Plugins to Maven plugin goal. If you see problems with the sql-maven-plugin you can ignore them and resolve them later (see step i).
- h. Click **Finish**, then click **OK** in the confirmation dialog.
- i. In the Eclipse Markers tab, you may see several "Maven Problems" related to the sql-maven-plugin. Select them all, then right-click, then click **Quick Fix**, then ignore them. The problems should not show up anymore.
- j. After the process is complete, right-click the project root directory, then click **Maven -> Update Project...**

7. Create the MySQL Database:

- a. Install MySQL 5.6.17 community edition (or the latest edition)
<http://dev.mysql.com/downloads/mysql/>
- b. After installation, open the MySQL Workbench program that installed with MySQL. Connect to the server, then execute the scripts:
`/eScreeningDashboard/src/main/sql/initialization/dev_env_run_once.sql`
- c. After running this step, open a command prompt and change directory to where the pom.xml file is.
- d. To create the tables and insert the test data, execute the following Maven command:

```
mvn integration-test -DskipTests=true -Drecreate_db=true -P dev
```

8. Configure the default web browser for Eclipse.

Using the eclipse menu, select **Windows/Web Browser/Firefox** (or **Chrome**).

9. Install Tomcat (this is optional since STS comes with its own tomcat)

- a. Download **Tomcat 7** from <http://tomcat.apache.org/download-70.cgi>
- b. Expand the compressed (downloaded) file and put its contents into a directory.
- c. Create a new Tomcat server in Eclipse that points to the directory where you put the contents.
- d. Double-Click the new server, then click **Open launch configuration**, then go to **Arguments**, then add this VM argument to the arguments already present:

`XX:MaxPermSize=256m`

10. Run in eclipse:
 - e. Right-Click **eScreeningDashboard**
 - f. Select **Run As/Run On Server**
 - g. To run the web app on the Eclipse Tomcat server, click **Next**, then **Finish**.
The browser launches and opens the web site.
11. Import coding convention file for eclipse (STS)
 - a. Start eclipse/sts
 - b. Select **Windows/Preferences**
 - c. Navigate on the tree node to Java/Code Style/Formatter
12. Click **Import...** select **spring-eclipse-code-conventions.xml** from here:
<https://github.com/spring-projects/spring-batch/blob/master/spring-eclipse-code-conventions.xml>

Servers

Tomcat 7.0.42 Tomcat will be used as the server for the Dashboard application. It is installed on the sandbox and integration application server. We recommend that developers have a local instance to check war file deployments against.

Apache-tomcat-7.0.42 (64 bit)

<http://tomcat.apache.org/download-70.cgi>

Frameworks

Software	Version	URL
Java	8u45	http://www.oracle.com/technetwork/java/javase/downloads/index.html
Spring Tool Suite	3.7.0.RELEASE	http://spring.io/tools/sts
MySQL	5.6.17	http://dev.mysql.com/downloads/mysql/
Apache Tomcat	7.0.53	http://tomcat.apache.org/download-70.cgi
Spring Framework	4.1.6.RELEASE	http://projects.spring.io/spring-framework/
Spring Security	3.2.7.RELEASE	http://projects.spring.io/spring-security/
Hibernate	4.3.0.Final	http://hibernate.org/orm/downloads/

VistALink	1.6.0.028	https://downloads.va.gov/files/FOIA/Software/Patches_By_Application/XOBV-VISTA%20LINK/XOBV_1_6/
jquery	1.10.2	http://jquery.com/download/
jquery ui	1.10.3	<a >version='1.9.2"' href="http://jqueryui.com/download/#">http://jqueryui.com/download/#"/>version=1.9.2
Bootstrap	v. 3.1.1	http://getbootstrap.com/
AngularJS	v. 1.2.15	https://angularjs.org/
Maven	3.1.1	http://maven.apache.org/download.cgi

Tasking and source control

We used a feature branch workflow with GIT described in the [GIT Workflow.pdf](#) .

VA sandbox wiki (requires VA sandbox account)

Put all official project documents here:

<http://sandbox.vacloud.us/groups/20388/>