**VA Salesforce  
Center of Excellence  
Standard Operating Procedures**

**Version 1.0**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Description |
| 1.0 | 12/08 | COE Team | Initial version |
|  |  |  |  |
|  |  |  |  |

Table of Contents

Introduction 5

Deployment Management Plan 6

Intended Audience 6

High Level Overview 6

Deployment Management Plan 6

Introduction 6

VA Salesforce Environment Management 7

Deployment Process 8

Deployment Technologies 13

Version control / Source Code Management 14

Current Source Code Management Process 14

Future-State Source Code Management Processes 14

Appendix A: Deployment Checklist Template 17

Pre-Deployment Manual Steps 17

List Change Sets 17

Post Deployment Manual Steps 17

Rollback / Backout Steps 17

Defect Management 18

Intended Audience 18

Defect Management Process 18

Types of Defects 18

Recording defects 18

Defect Management Life Cycle 19

Documenting and Reporting Defects 19

Resolving Defects 20

Defect Release Management 20

Support Models 21

Intended Audience 21

Overview 21

Support Models Best Practices 21

Pre Go Live Checklist for Production Support Team 21

Post Go Live Checklist for Production Support Team 21

Pre Go Live Checklist for Project Team 22

Post Go Live Checklist for Project Team 22

Quality Assurance 23

Intended Audience 23

Quality Assurance Management Process 23

Quality Assurance Overview 23

Types of Testing 24

Unit Testing 24

Integration Testing 25

Functional End-to-End Testing 25

Regression Testing 25

Quality Assurance Testing 25

Quality Assurance Environments 25

Sandbox Management 26

Testing Team Roles 26

Test Plan Guidelines 26

Release Schedule 27

Training 28

Intended Audience 28

Overview 28

VA Training Process 28

Center of Excellence Best Practices 28

Training Methodology Overview 28

Training Delivery Methodologies 29

Data Management 31

Intended Audience 31

High Level Overview 31

Data Management 31

VA Data Backups 31

Data Archiving 31

Audit Logs 31

Salesforce Center of Excellence Best Practices 32

Exporting Data 32

Data Export Applications 32

User Account Maintenance 33

Intended Audience and Overview 33

VA Specific User Management Policies 33

Account/License Types 33

Profiles 34

Roles 34

Public Groups 34

Permission Sets 34

Queues 34

User Management Best Practices 35

Creating & Maintaining Users 35

Authenticating Accounts 35

Deactivating Accounts 35

Salesforce COE Best Practices Overview for User Management 36

Issue and Requirements Tracker 37

Intended Audience 37

Developer Responsibility Tier 2 Defect Resolution 38

Introduction

The purpose of this document is for documenting the processes for different functions that the Salesforce Administrators at The Department of Veterans Affairs perform. The document is divided into the following sections:

* Deployment Management Plan
* Defect Management
* Support Models
* Quality Assurance
* Data Management
* Training
* User Account Maintenance
* Issues and Requirements Tracker

Deployment Management Plan

# Intended Audience

The primary audience for this document are members of the project teams responsible for developing applications that provide business capabilities on the VA Salesforce platform. Project managers should make this document available to all team members supporting application deployment to ensure they are familiar with the processes described in this document and understand the activities required to ensure a successful deployment.

The secondary audience for this document are the VA Salesforce System Administration team and the VA Salesforce Configuration Management team. These teams will have a working knowledge of the processes described in this plan.

# High Level Overview

The Department of Veterans Affairs (VA) Salesforce platform is a shared resource that enables projects across VA to quickly develop robust, flexible and scalable applications in a secure cloud environment. The VA Salesforce Center of Excellence (COE) provides structure and guidance necessary to assist project teams maximize VA’s investment in the platform, with the objective of helping project teams deliver quality solutions.

This document outlines the processes and technologies used to deploy VA Salesforce applications from development environments to the Systems Integration Testing (SIT), User Acceptance Testing (UAT), and Production (PROD) environments.

As use of the VA Salesforce platform grows, it will become increasingly important to manage changes to the configuration of the platform. This document describes the version control processes currently employed by the VA Salesforce platform and future version control processes that uses employ source control software to manage VA Salesforce metadata.

# Deployment Management Plan

## Introduction

The VA uses a centralized model for Salesforce deployment and release management activities. Responsibility for developing and maintaining deployment management processes lies within the VA Salesforce COE. Deployment of solutions to the systems integration testing, user acceptance testing, and production environments are executed by the VA Salesforce Administration team.

Project Teams perform a key role in the deployment processes. They ensure that deployment tasks are documented and tested; prerequisites tasks are completed successfully; and they work collaboratively with the VA Salesforce COE and the Salesforce Administration team to help ensure successful deployments.

This document is organized into the following sections to provide teams with a reference to understand the deployment process and the activities each organization is responsible for performing:

* **VA Salesforce Environment Management:** Provides an overview of the Salesforce environments in use at the VA and the environment management tasks the VA System Administration Team performs.
* **Deployment Process:** Describes the tasks performed by project teams and the VA Salesforce System Administrators to deploy applications on the VA Salesforce Platform.
* **Deployment Technologies:** Describes the technologies used by the VA Salesforce COE to perform deployments.
* **Version Control / Source Code Management**: Describes current version control management process used by the VA Salesforce Platform and provides an overview of a future state version control management system used to maintain Salesforce code and configuration metadata.
* **Deployment Checklist Template:** Template (found in the appendix) for the deployment checklist that will be created and maintained by VA Project Teams.

## VA Salesforce Environment Management

To understand the deployment process, teams must be aware of the various Salesforce environments at the VA and how they are managed. The table below provides an overview of each type environment and the supporting sandbox:

| Environment Type | Purpose | Sandbox Type | Deployments Performed By | Comments |
| --- | --- | --- | --- | --- |
| Production | Provides business capabilities to production users | N/A | VA Salesforce Administration Team | Currently there are three production Salesforce organizations operating in the VA environment. Administrator access to production environments is strictly controlled. (Note, efforts are underway to consolidate VA orgs) |
| User Acceptance Testing (UAT) | User Acceptance Test, and Staging | Full Sandbox | VA Salesforce Administration Team | The UAT sandbox is treated similarly to the Production environment. Accounts with elevated privileges are held only by the VA Salesforce Administration team. |
| System Integration Test (SIT) - Consolidated | Testing of code integrated from multiple project teams | Partial Data / Developer Pro | VA Salesforce Administration Team | Consolidated SIT sandboxes are available to integrate test code created by multiple project teams. |
| System Integration Test (SIT) - Team | System testing of code from a single project team | Developer | Project Teams | SIT sandboxes are used by project teams to conduct system test and to perform the mock deployments necessary to develop and confirm their deployment checklists. |
| Development | Development and Unit Test | Developer | VA Salesforce Administration Team\* | The VA Salesforce Administration team is responsible for development sandbox creation and refresh. Development sandboxes are not typically targets for migration. |

The VA Salesforce administration team is responsible for the management of all Sandboxes, including:

* **Sandbox Creation:** To request a new Sandbox, project teams open an issue with the VA Salesforce Help Desk. The VA Salesforce Administration team determines if the Sandbox supports an approved project, validates that an unused sandbox is available, and creates sandboxes for requests approved by the COE.
* **Sandbox Refresh:** To request a Sandbox Refresh, project teams open an issue with the VA Salesforce Help Desk. The VA Salesforce administration team determines if the Sandbox is eligible for refresh and confirms that the current use of the sandbox allows for a refresh to occur.
* **Sandbox Monitoring & Deletion:** The VA Salesforce Administration team monitors sandboxes for inactivity. If a Sandbox has not been used for three months, the administration team contacts the project and determines if the sandbox is a candidate for deletion.

## Deployment Process

This section describes how code and configuration are deployed in the VA Salesforce environment. It includes two diagrams. Figure 1 illustrates how code progresses from the development, and testing environments to production. Figure 2 illustrates the process that is followed to execute deployments.

Project Teams are responsible for designing and configuring their applications within their respective team development sandbox. Project teams migrate code to their Team SIT sandbox as needed. When testing is complete, the VA Salesforce Administration team then migrates code to a consolidated SIT environment, UAT, and then Staging, and the final migration to production. Figure 1 highlights this flow. Arrows representing the sources and targets of the change sets used in this migration.

**Change sets** are the mechanism by which configuration changes are deployed at the VA. This document will be updated over time as the deployment mechanism matures.

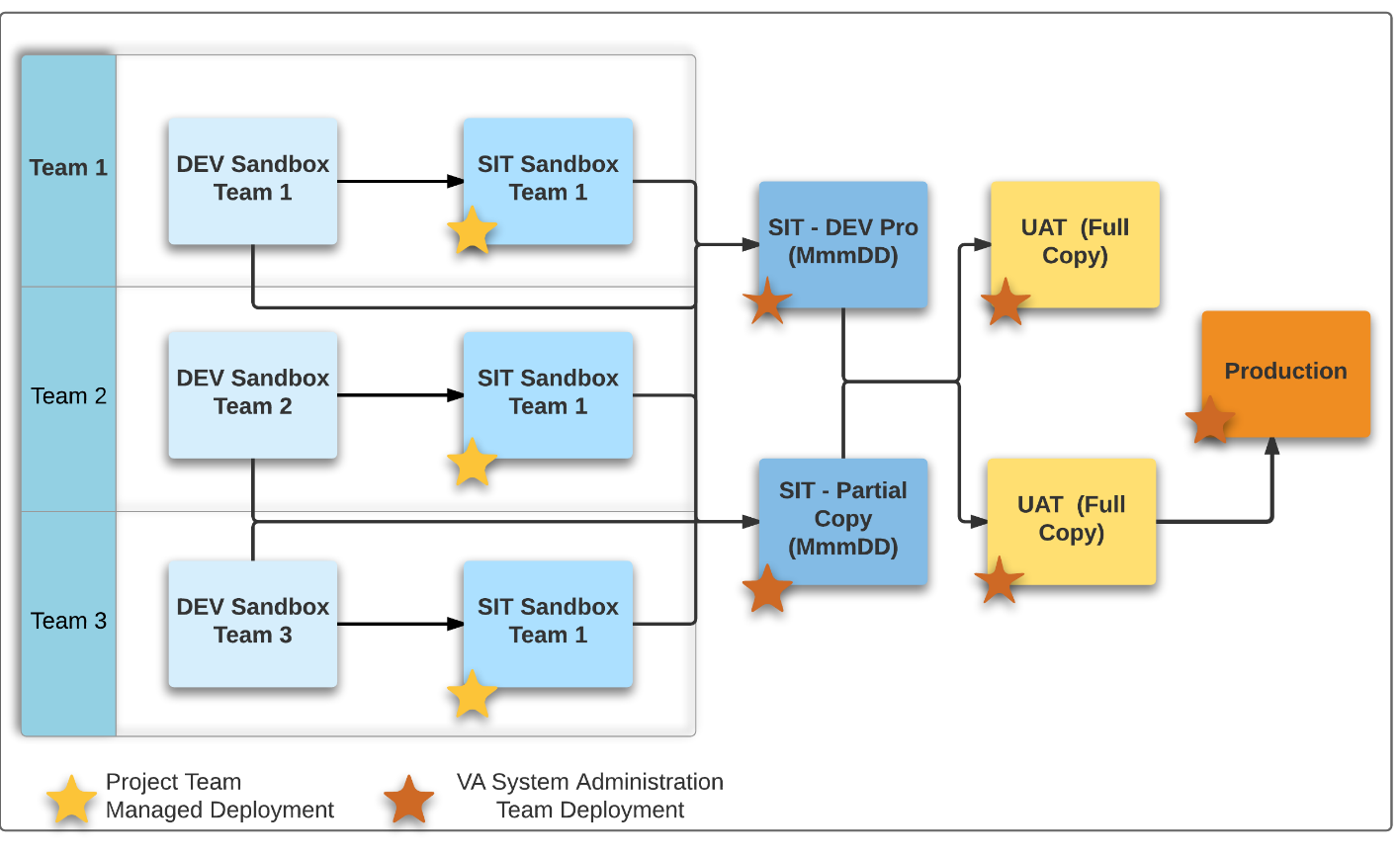


Figure 1: Migration Flow

Figure 2 graphically depicts how deployments are conducted on the VA Salesforce Platform.  Deployment checklists created by the project team are executed by the VA Salesforce Administration team to migrate code to the consolidated SIT, UAT, and production organizations.

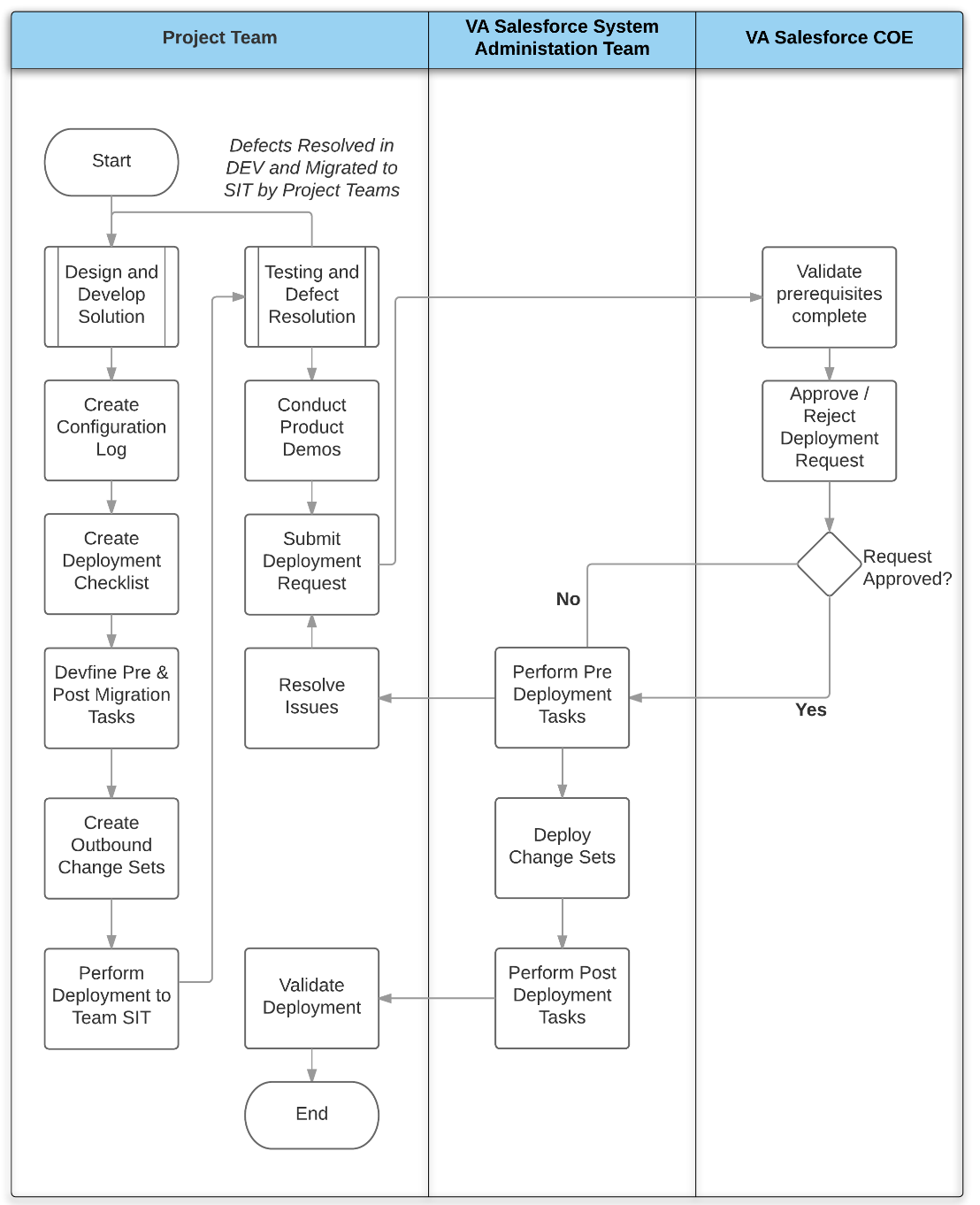


Figure 2: Deployment Process

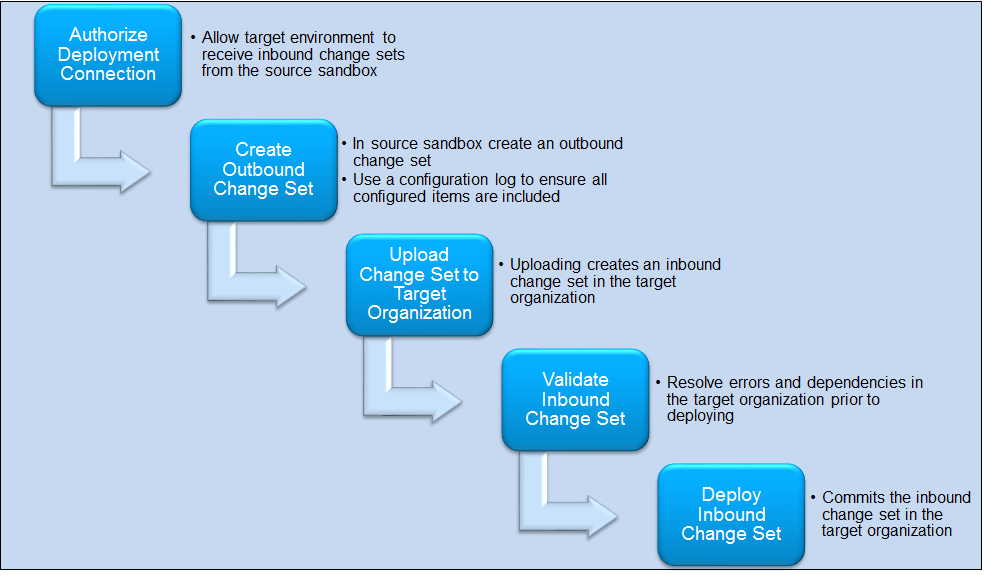
The tasks involved in the deployment process are described in detail below. Please note, that not all tasks in this table are performed for all project deployments. Additionally, rows with a gray background are not deployment tasks, but are listed to provide the context of the deployment task within the overall development process.

| Task | Performed By | Comments |
| --- | --- | --- |
| Create Development Sandbox | VA Salesforce Administration Team | When a project is approved, the VA Salesforce Administration team creates a sandbox and provides access to project team |
| Design and develop solution | Project Team | The project team configures Salesforce to meet requirements as detailed in the approved user stories |
| Create and maintain a log of configured items | Project Team | Project teams maintain a configuration log listing all Salesforce components created or updated during development |
| Create and maintain Deployment Checklist | Project Team | Lists the manual steps performed as part of deployment and the change sets involved in the deployment |
| Deploy to Team SIT Sandbox | Project Team | Project teams follow the deployment checklist and use change sets to deploy their configuration to their SIT sandbox. During this deployment, the deployment checklist is validated and refined. |
| Perform System Integration Tests | Project Team | Testing performed in team SIT Sandbox |
| Remediate defects and deploy to SIT | Project Team | The project team remediates defects in DEV sandbox and migrates to SIT for each successive test cycle. Best practice is for project teams to continuously deploy to their own SIT environment and use the SIT environment for product demonstrations. |
| Deploy to consolidated SIT environment | VA Salesforce Administration Team | The VA Salesforce Administration team uses the deployment checklist to deploy to the consolidated SIT environment |
| Conduct Integration Testing | Project Team | Testing performed in consolidated SIT sandbox |
| Remediate defects and deploy | Project Team / VA Salesforce Administration Team | Defects are remediated in the DEV sandbox and migrated to the consolidated SIT environment by the VA Salesforce Administration Team |
| Validate UAT Requirements met | VA Salesforce COE | Verifies that code reviews have been complete, SIT complete, and code coverage standards have been met |
| Conduct UAT Migration | VA Salesforce Team | The VA Salesforce Administration team uses the deployment checklists to migrate the configuration to the UAT environment |
| Conduct User Acceptance Testing | Project Team | UAT performed full copy sandbox |
| Remediate defects and deploy to UAT | Project Team / VA Salesforce Administration Team | Defects are remediated in the DEV sandbox and migrated to the UAT environment by the VA Salesforce Administration Team |
| Verify Production Prerequisites | VA Salesforce Administration Team | Verifies completion of UAT and the Product Owner acceptance of the application prior to production deployment |
| Conduct Production Migration | VA Salesforce Administration Team | The VA Salesforce Administration team uses the deployment checklists to conduct the Production deployment |

## Deployment Technologies

Deployments on the VA Salesforce Platform are currently conducted using the following:

**Change Sets**: Change sets are a native Salesforce technology used for migrating Salesforce components (metadata) between related sandboxes. This is the process currently in use at VA. The process to deploy Salesforce components using change sets is illustrated in the figure below:



Change sets allow fine-grain control over the items that Salesforce will deploy. Project teams must ensure they have included all configuration items developed or updated in their release when defining the change set. A configuration log is used for managing the creation of a change set.

Change sets must be recreated manually in each source environment. This is a manual and tedious process. The configuration log is a critical input that project teams must provide to the VA Salesforce Administration to allow them to duplicate change sets in higher environments (SIT - Full, UAT).

**Configuration Log:** The configuration log is a complete list of all the Salesforce components created or updated in a release. Configuration logs can be maintained in spreadsheets. Project teams must ensure their configuration log is up to date, throughout the development lifecycle.

**Deployment Checklist:** The deployment checklist defines all the activities needed to perform a deployment. It includes the following:

**Pre-deployment tasks**: An ordered list of activities that must be manually performed prior to deployment

* **Change Sets:** An ordered list of the change sets to be applied during this deployment
* **Post-deployment tasks:** An ordered list of activities that must be manually performed after deployment
* **Verification Steps**: A list of tasks to be executed that confirm the deployment was successful.
* **Roll-back Procedures:** If the deployment of a change set fails, Salesforce will automatically roll-back the metadata changes. However, pre-deployment tasks (documented above) must be manually backed out. The roll-back procedure describe how to back out these tasks.  Note, once a change set successfully deploys, the change set cannot be rolled back.

# Version control / Source Code Management

## Current Source Code Management Process

The VA Salesforce Platform currently uses developer sandboxes to support configuration management. Weekly and prior to the deployment of significant changes to production, a configuration-only sandbox is refreshed with the latest production metadata. This process allows a 7-week rolling history of metadata to be maintained.

## Future-State Source Code Management Processes

Reliance on manual deployments via change sets and use of sandboxes to maintain software versions is a good initial solution for the VA Salesforce platform. However, as the use of the platform grows and projects increase in complexity, the limitations of the tools and processes will become more apparent. Some of the limitations of the existing process include:

* Requires a labor-intensive (manual) process to develop and maintain change sets.
* Deployments that require objects to be deleted or renamed require manual steps. (change sets cannot delete or rename components).
* Conflict identification and resolution is more difficult without a source code management solution
* Traditional development techniques including branching and merging code lines is more difficult without a source code management system.

A phased approach, as illustrated below, should be used to incorporate new technologies and processes that address the above limitations.

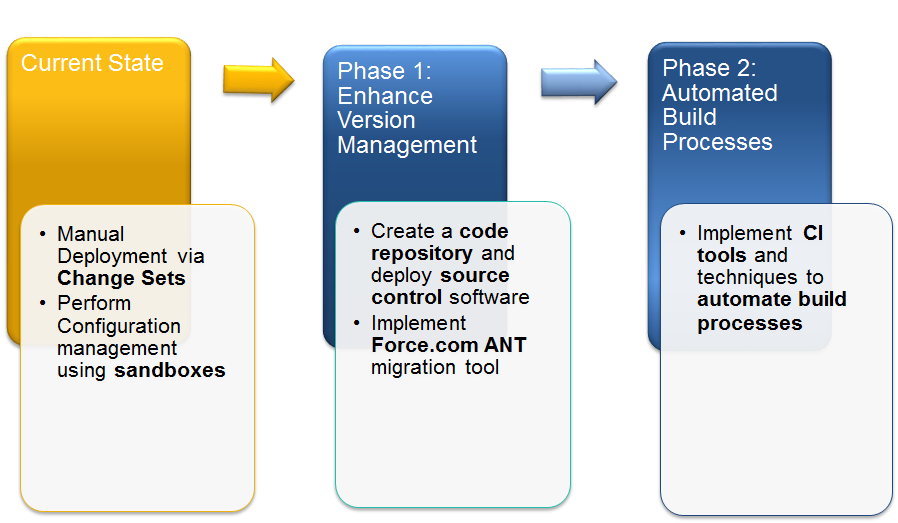


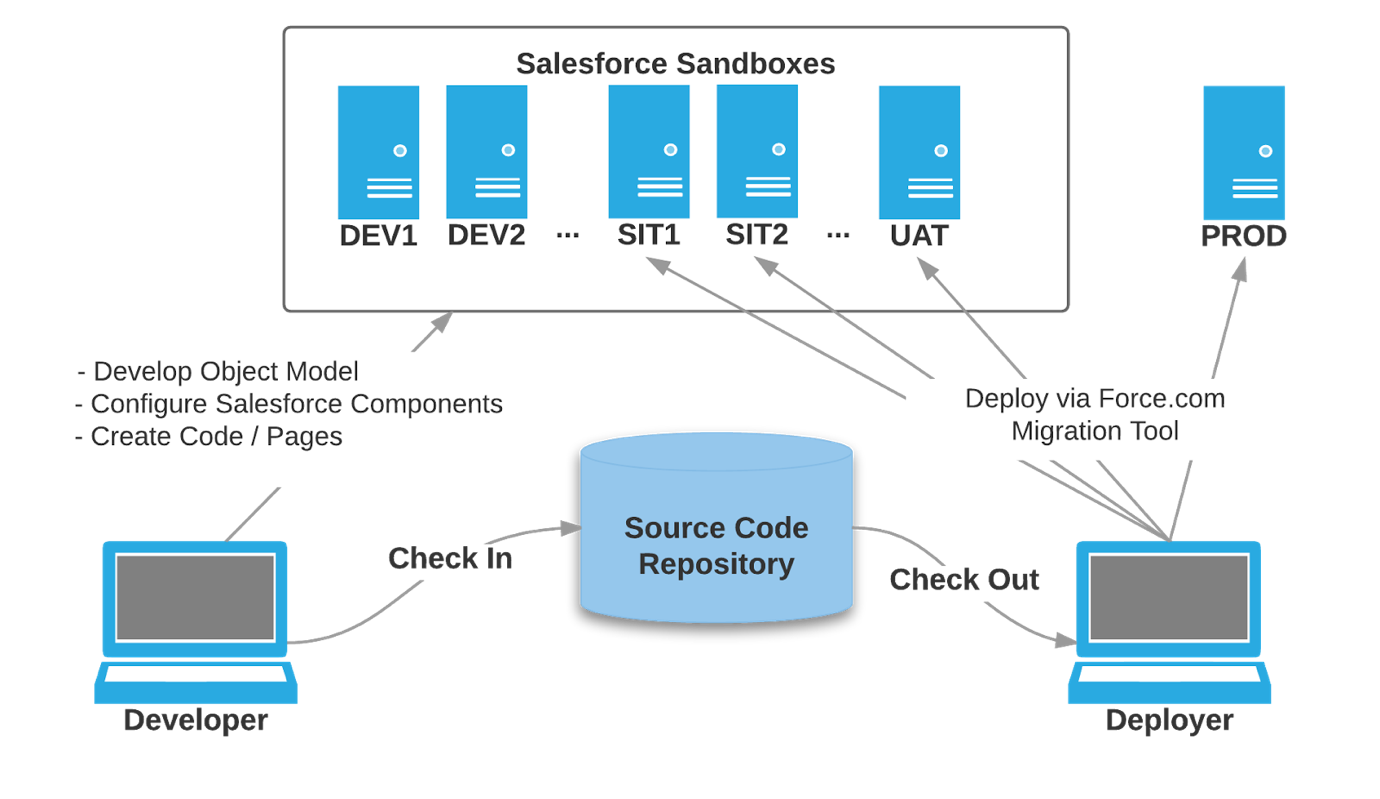
Figure 3: Phased Approach

The first phase in this process is to transition away from using sandboxes for configuration management through the implementation of a code repository and source code management software. This has the added benefit of allowing deployments to be scripted using the Force.com ANT migration tool, reducing the manual effort associated with creating change sets.

It is recommended that a source code management tool should be used because of the following benefits:

* Providing the ability to track changes and easily reverse changes in code as needed
* Allows multiple developers to work on their own copy of a resource
* Allows conflicts in resources to be identified and managed by developers
* Allows multiple branches from the main code line to be worked simultaneously and merged at various points during the development lifecycle

Below is a notional diagram of how a source code management system can be integrated into the VA Salesforce environment. In this approach, Salesforce project team developers configure their solution in development sandboxes, however they also use the Salesforce Force.com IDE to check their changes into the VA source code repository. Deployment is performed scripted using the Force.com Migration tool.



The Force.com Migration tool is a command line utility that moves metadata between a local file system and a Salesforce organization. Instead of change sets, components for deployment are identified using an XML manifest (package.xml). The scripted retrieval and deployment of components allows for repetitive deployment management processes to be executed more efficiently.

As the VA Salesforce deployment management process matures, the VA build upon the platform established in the first phase by replacing the Force.com Migration Tool with a build automation tool like Jenkins. Jenkins can automatically monitor a Sandbox for metadata changes, and check code into the source code repository. Deployments to other sandboxes and Production can be similarly automated.

# Appendix A: Deployment Checklist Template

## Pre-Deployment Manual Steps

List the Pre-Deployment steps for this deployment below in order in which they are executed, this list may include Freeze Users, delete Apex Jobs, session settings, Timeout values, Create/ Delete groups, permissions sets etc.

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Task Name | Description | Note |
|  |  |  |  |
|  |  |  |  |

## List Change Sets

List the Change Sets used for this deployment below in the order in which they are needed, including specific reference to the source sandbox.

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Task Name | Description | Note |
|  |  |  |  |
|  |  |  |  |

## Post Deployment Manual Steps

List the Post Deployment steps for this deployment below in order in which they are needed, this list includes Freeze Users, delete Apex Jobs, session settings, Timeout values, Create/ Delete groups, permissions sets, user modification/creation, etc. Project Teams should consider the setup for testing deployed sandboxes and production sandboxes during verification.

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Task Name | Description | Note |
|  |  |  |  |
|  |  |  |  |

## Rollback / Backout Steps

List all the steps necessary to rollback this deployment if necessary.

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Task Name | Description | Note |
|  |  |  |  |
|  |  |  |  |

Defect Management

# Intended Audience

The primary audience for this section are the personnel and teams at the VA involved with identifying, testing, and fixing the defects identified in Salesforce applications. The secondary audience for this section are the VA Salesforce Administrators.

# Defect Management Process

The Defect Management Processes is an important piece of the Quality Assurance and Testing effort. The process helps identify issues and focuses on fixing them based on their priority and severity. General best practices for defect management include making changes in one sandbox, and following a repeatable process for moving the change to production.

This document outlines the defect management process that the VA should follow when defects are identified in Salesforce applications.

Defect Management is standardized by a Center of Excellence (COE) for the following reasons:

* Effective defect management improves the quality of applications deployed on the platform, drives higher returns on investment, and promotes better overall adoption of the platform
* A consistent defect management process that spans applications on the platform allowing for a better analysis of defects, and their root causes
* Insights gained from analysis can lead to defect prevention by enabling resolution of common defects through configuration standards and coding standards which therefore reduce cost corrections

## Types of Defects

Defects reside in multiple places. Most *defects* at VA will be in one of these environments:

* **Development/Testing Sandbox**: These are defects identified during Unit Testing, End to End Testing, or User Acceptance Testing
* **Production:** These are defects identified in the production org. The best practice is to follow the issue tracking process set up by the team supporting the Salesforce platform identified for the organization

## Recording defects

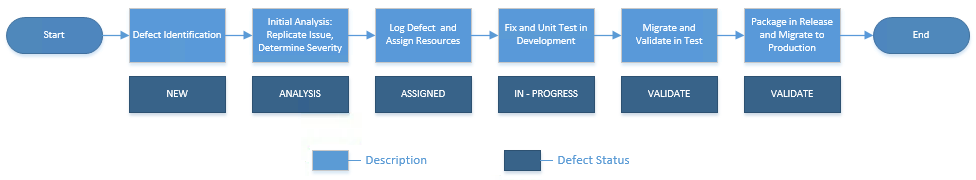
Periodically, end users report defects that require changes to the application. This section describes handling defects, from reporting through resolution.

### Defect Management Life Cycle

It is important to fix a defect based on its priority and severity. Usually the development team fixes defects in the development sandbox. Then changes are pushed to the testing sandbox. After testing verifies that a defect is resolved, the update is moved to production during the next scheduled deployment. The steps involved are:

* Defect Identification
* Initial Analysis
* Log Defect and Assign Resources
* Fix and Unit Test
* Migrate and Validate in Test Sandbox
* Migrate to Production

Here is a process flow diagram of the Defect Management Life Cycle.



### Documenting and Reporting Defects

The system of record should log certain defect information so that the person resolving the issue can easily understand, recreate, and prioritize the defect. At a minimum, the following information should be included for each defect:

* **Defect Description:** A brief description of the defect/issue for searching and reporting. This also helps determine whether it is an actual system defect or a User Error.
* **Steps to Recreate:** The steps needed to recreate the issue so the person assigned to it can determine if it is a defect or a user error. This helps the support people fixing the issue understand what is not working in the system.
* **Screenshots:** Includeif there is an error code displayed or if a screenshot is relevant to the defect.
* **Severity of the Defect:** The user reporting a defect may determine its severity. The group that resolves a defect or issue can modify a defect’s severity level as appropriate. In general, if there is a work around for a defect, its severity is low; similarly, if there is no work around for a defect - and the defect prevents or impedes system use - then typically the severity level is high.
* **Priority of the Defect:** priority may be assigned by the group that works defects. This helps determine the queue that a defect is assigned to and the individuals that will resolve it. This approach also helps determine the time and resources needed to fix a defect.

### Resolving Defects

When defects are identified, the original developer is contacted to find a solution if they are available. When the defect is resolved, it is tested through all testing stages from development to UAT. If a defect cannot be resolved, additional resources, including System Administrators, will help find a solution.

## Defect Release Management

Coordinating defect resolution with the Release Management Team is an integral part of the COE deployment process. The developer of the solution is responsible for testing the solution through the appropriate sandboxes. After testing, the developer creates a change set in their test org and validates that it works through all the other orgs. If it fails anywhere, the developer is responsible for resolving the issue and returning to the beginning of the change set verification process.

Per the COE guidelines, Go-Live and Security Package Approval (if required) will be completed prior to any deployment to Staging or Production. In addition, updates to a Technical Deployment Plan will be made and reviewed with the Release Management team.

Support Models

# Intended Audience

The primary audience for this section are the personnel, project teams, training teams, and change management teams involved with Salesforce Support. The secondary audience are the VA Salesforce Administrators that support the users and applications.

# Overview

This document outlines multi-tiered support models for project teams at VA to follow while supporting new VA Salesforce Applications. This document also identifies support models for changes made to existing Salesforce Applications in the VA Salesforce instance.

# Support Models Best Practices

Periodically, end users report issues that require changes to an application. Issues that typically require less than one business day’s level of effort and need to be resolved promptly are considered to be reasonable and necessary, and fall under this support definition. These support issues are not be considered configuration or integration enhancements.

Two teams are involved with deployments: the Production Support Team and the Project Team. Each of these teams has a checklist for Pre and Post deployments. These checklists are identified in the following sections.

## Pre Go Live Checklist for Production Support Team

Depending on the scope of the support issues, certain tasks occur. The following is a list of activities the Support Team could be responsible for prior to going live:

1. Conduct UAT sessions and Train-the-Trainer (TTT) sessions as needed
2. Complete User eLearning if applicable
3. Define the Production Support Teams’ Admin rights levels by profiles, permission sets, and roles
4. Establish guidelines to identify and fix bugs vs. longer term project enhancements
5. Post any training material to a group site or store this material in a shared drive
6. Set Service Level Agreements (SLAs)
7. Preplan a way to categorize issues and prioritize fixes (for example: critical, high, medium)
8. Create a release schedule/governance plan for future enhancements

## Post Go Live Checklist for Production Support Team

The following is a list of activities that the Support Team could be responsible for after deployment:

1. Leverage User Tip Sheets for training questions
2. Track repeated issues to escalate for root cause fix
3. Keep an updated release schedule
4. Follow a governance plan for future enhancements

## Pre Go Live Checklist for Project Team

The following is a list of activities that that the Project Team could be responsible for before deployment:

1. Host UAT, TTT, and Code review for Production Support Team
2. Communicate to Production Support how to escalate issues
3. Communicate to Production Support what is expected of Users
4. Clarify roles and provide contact information (as applicable) for Managers, Champions, Project Team Members, SIs, and Salesforce Premier Support
5. Communicate how end users can get help (with SLA and instructions for each) with issues
6. Communicate processes to submit the following:

* Training/Process Questions
* Minor Technical Issues
* Major Technical Issues

## Post Go Live Checklist for Project Team

The following is a list of activities that the Project Team could be responsible for after deployment:

1. Communicate FAQs on new processes
2. Communicate technical errors with the root cause, user impact, and repair plan/schedule
3. Communicate fixes as they are pushed to production and any consequent action needed by users
4. Notify user(s) when their application improvement ideas are implemented
5. Host meetings with Production Support

Quality Assurance

# Intended Audience

This section is intended for personnel and teams at The Department of Veterans Affairs (VA) who are involved with testing and Quality Assurance processes in Salesforce applications. The section will also help the VA Salesforce Administrators have a clear understanding of the Quality Assurance Process.

# Quality Assurance Management Process

Quality Assurance Processes help monitor the quality of work and to identify and prevent non-recurrence of defective services. They ensure conformance to one or more guidelines provided by Salesforce in order to implement a solution. To fulfill this objective, the QA process vets work products through a multi-tier review process. These multi-tier reviews allow all relevant parties adequate time to review project deliverables prior to formal acceptance. This process provides VA with details about how/when to inspect work, groups and people responsible for inspecting work, the process for addressing defects, and critical metrics to track effectiveness of other tasks.

To ensure that quality standards are upheld, VA will use a Quality Assurance (QA) plan to control the quality of work and to help identify and prevent non-recurrence of defective services. When new Salesforce applications are deployed to a VA Salesforce Organization, specific steps are followed. To fulfill this objective, the QA plan vets work products through a multi-tier review process. These multi-tier reviews allow all relevant parties adequate time to review project deliverables prior to formal acceptance. The QA plan provides details about how and when to inspect work, the person responsible for inspecting work, the process for addressing defects, and critical metrics to track data cleanup effectiveness.

# Quality Assurance Overview

Because testing begins with requirements and continues throughout the lifecycle of an application, each interim work product created by the development team will be tested by the development team and the testing teams to ensure that it meets the client’s requirements and specifications. This reduces risk, improves the quality of the resulting solution, and delivers solutions that meet end-user needs. The following testing should be included in the process of Quality Assurance:

* **Unit Testing:** Performed by developers to ensure code and configuration performed as defined in requirements.
* **Integration Testing:**Verifies that different applications that communicate with each other do so in the correct manner and continue to after new applications are deployed.
* **Functional End-To-End Testing:** End-to-End testing verifies the application flows from beginning to end in the manner that it was designed and intended to.
* **User Acceptance Testing (UAT):** Usually the last step before deployment. End Users perform testing with real world scenarios to make sure the application handles tasks as specified in the requirements.
* **Regression Testing:** Verifies that the pre-existing applications and code continue to function as required when the new application is deployed.

These are some general Quality Assurance:

* Write test cases during design and prior to the start of configuration or coding
* Ensure that User Stories have acceptance criteria and these test cases are included in the unit test cases
* Leverage automated testing and test classes wherever possible
* Do not write test classes simply for code coverage, but to ensure that error conditions are adequately handled

## Types of Testing

### Unit Testing

Unit testing refers to localized testing performed by individual developers on a small unit of code to ensure that the code meets the original specification. Unit testing is done by the original developer of the code to verify the code works on its own, before it is built into the larger application and with other code. This section provides guidelines on unit testing so developers can produce effective code efficiently without a lot of rework. Unit Testing is conducted in the Development sandbox or Configuration sandbox.

There are a few ways to perform unit testing for code. One way is to manually perform functionality testing on the front-end; another is to write code to run tests. The best practice is to utilize test code whenever possible to perform unit testing, so more test scenarios can be written into and tested by the test code.

The following is a Testing Checklist that should be followed:

1. **Code Coverage:** Salesforce requires that **at least 75%** of Apex code be covered by unit tests in order to deploy code to production sandboxes. In addition, all triggers must have some test coverage. As a best practice, the Development Team will work to achieve **90-100% code coverage, where possible**. Please note that calls to System.debug are not counted as part of Apex code coverage in unit tests.
2. **Best Practices:** For more information, please refer to the following video on Apex Test Coverage Best Practices <http://developer.force.com/df_session?id=a0J300000009wdyEAA>
3. **Work in Small Batches:** Unit test should be done at the smallest testable level so each unit of code is tested and validated, before a developer moves on to writing additional code. Salesforce best practices recommend that developers continuously test as code is developed. This helps catch and correct mistakes before a larger batch of code is written based on nonfunctional code. More complex code requires more test iterations. As a result, bug fixing becomes more time-consuming. With this in mind, developers should work in small batches, making sure each additional unit of code is fully functional before moving on to the next batch.

### Integration Testing

Integration testing ensures all downstream and upstream applications and systems are functional and operating as intended. Integration testing also allows for testing of code across multiple developers and teams. This step validates all integration points, and verifies that all code inter-dependencies are met. Integration Testing will take place in the VA Partial Sandbox.

As part of the Functional Integration Testing, a document that defines the steps followed to complete testing which will be used as a guideline by the team performing testing.

Integration Testing validates the interface between different units of code. It identifies any assumptions made about the behavior of code that communicates with other code or programs. It is highly recommended that integration testing take place in the QA sandbox. Integration Testing should be conducted across different teams, after code has been merged. This helps ensure that code is not in conflict in the sandbox.

### Functional End-to-End Testing

Functional End-to-End (E2E) testing helps ensure all requirements are met and the application is business ready. The documentation for Integration and Functional E2E Testing is similar.  The example test document in the  Integration Test Guidelines provides more information.

E2E testing is conducted on the complete, integrated system. It helps verify that all functional requirements are met. E2E testing is performed after integration testing in the same sandbox.

### Regression Testing

Regression Testing ensures that the capability of pre-existing features remains whenever new functionality is deployed. This is necessary to ensure that changes do not negatively affect existing functionality. If possible, regression testing should be performed by the project team(s) that developed and tested the capability. If this is not possible, the COE, product owners, and/or others familiar with the original functionality should conduct the regression testing.

### Quality Assurance Testing

Quality Assurance Testing ensures that developers have coded and made and modifications that work as designed. This testing is typically performed by a test team that is separate from the development team.

## Quality Assurance Environments

VA will have five Salesforce environment: Development, Quality Assurance (QA), Development SitBox, UAT, and the Production org.

* **Dev Box:** The Development sandbox is used by the Development team for all of their code. This environment is used to develop applications. This is all where the team tests defect fixes and unit testing.
* **QA:** The QA sandbox
* **Dev SitBox:** This environment supports system integration testing
* **UAT:** The User Acceptance Testing sandbox is where VA will test the new functionality of Salesforce prior to deployment to production. UAT ensures functionalities and components operate as expected by VA and that any discrepancies are reported to the Center of Excellence (COE).
* **Production:** The Production org is where all deployed configurations and code reside. This is where end users perform their duties in Salesforce.

## Sandbox Management

Sandboxes are managed by a select group of VA System Administrators. The sandboxes are refreshed based on a schedule that is published by this group of Admins. There are five types of Sandboxes:

* **Full Sandbox:** Used for staging and refreshed monthly
* **Partial Sandbox:** Use for UAT, Integration, and Change Set Testing. This sandbox is refreshed monthly.
* **Development Professional Sandboxes:** There are five of these used for testing code, RDT, and Carenow SIT. These are refreshed monthly.
* **Development Sandboxes:** Used for developers and supplied as needed. Refreshes occur on an as needed basis.

## Testing Team Roles

There are different roles depending on the stage of testing. The Test team roles are:

* **Development Team:** The development team writes code. This team also conducts Unit Testing and supports all other testing as needed.
* **QA Tester:** The QA tester performs integration testing, end-to-end testing and integration testing. They also support the development team with their efforts as necessary.
* **Salesforce Administrator:** The Salesforce Administrator ensures individuals have appropriate access to all Salesforce environments. He coordinates the sandbox refresh schedules and maintenance of the sandboxes.
* **Peer-to-Peer Review:** Allows the owner of the work product to define and present deliverables to a subset of the team. This ensures all necessary details, findings, and suggestions are accurate and consistent, In addition, this collaboration within the team provides knowledge sharing and a channel for ideas to be added to a solution.

## Test Plan Guidelines

When new functionality is added, the team submits a test plan to the VA for review and as an artifact. An example of a Test Plan table follows.

|  |  |
| --- | --- |
| **Field Name** | **Description** |
| Test Case Number | For Tracking Purposes |
| Use Case | ID of related Use Case or User Story |
| Description | Description of Test Case |
| Preconditions | List any conditions that must be met prior to test scenario starting |
| Inputs | List any inputs needed for the test scenario |
| Expected Outputs | The expected outputs resulting from the test case, if any. |
| Results | The expected result from the test case |
| Post Conditions | The expected state the test case will create. |
| Comments | Any comments if necessary. |

There are additional guidelines specific for UAT. These include but are not limited to:

* Step-by-Step instructions for testers to for each testing scenario
* Expected test results
* Does the test pass or fail per the user?
* Comments

## Release Schedule

VA System Administrators provide a schedule that is updated weekly with the release schedule for deployments. This schedule provides information related to User Story due dates and Deployments up to a year in advance.

Training

# Intended Audience

The primary audience for this section are the personnel at The Department of Veterans Affairs (VA) involved with delivering Salesforce application training to users. The secondary audience is the VA Salesforce Administrators that support users and applications.

# Overview

This document addresses key training methodologies to utilize when a new application is deployed in the VA Salesforce organization. Additionally, this document supports a training plan intended to achieve the learning objectives of the Users who are targeted for new applications pushed in VA Salesforce. It identifies the key milestones, delivery format, and curriculum for each new application.

# VA Training Process

*VA has an internal Training process that the COE needs clarification about. This section will be updated when the information is provided.*

## Center of Excellence Best Practices

This section describes a high-level approach to best practices for training in a Salesforce environment. Included here are the methodology overview, delivery methods, benefits of Train-the-Trainer Training, and training materials.

### Training Methodology Overview

Training creates awareness, builds understanding, and promotes adoption among end users. Training methodology encompasses: Training Approach, Goals and Objectives, Assumptions and Scope, Audience Group, Training Curriculum, Training Development, Training Delivery Methods, Training Delivery, and Training Environment.

The following should be considered in the training methodology:

* The key trainers in each business function
* Content and delivery methods
* Modules and learning objectives for different audiences
* Training activities
* Training environment requirements to support demonstrations, exercises, and simulations
* How training will be supported
* How to measure training effectiveness

Trainers are responsible for training a defined list of users. Training and user adoption strategies should be based on defined success criteria and training rationale should be modular. As a result, this enables reuse across functions and user groups. Below is an example of a form that can be used to identify the type of training and materials needed for new applications and updates.

**VA Salesforce <Insert Application Name>**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| App Version | Quantity | Training Resource | Special Notes  (Complexities, Dependencies, Issues, Risks | Draft Delivery | Go-Live Delivery |
| *Identify if this is new or an update.* | *How many people need training?* | *What training method to use? (In-person, brown bag, quick reference guide,etc)* | *Identify any issues with the training or materials.* | *Date* | *Date* |

### Training Delivery Methodologies

Different training can be more effective depending on the audience and the training objective. The following are examples of training that might be used at VA.

* Train-the-Trainer
* Individual hands-on instruction
* Hands-on classroom style instructor-led training
* Seminar-style group demonstration
* Computer based training/Recorded Training

#### Train the trainer

Of particular note, Train-the-Trainer has the highest usage and success rate. The Train-the-Trainer model enables experienced personnel to show less-experienced instructors how to deliver courses, workshops, and seminars. Usually, a new instructor first observes a training event led by the course designer or subject-matter expert. These training sessions typically have less students than the courses the trainers will eventually teach.

A Train-the-Trainer workshop can build a pool of competent instructors who can then teach the material to others by:

* Instructing small groups of individuals in a focused training environment
* Preparing trainers to train individuals and respond to participants’ questions
* Preparing trainers to lead activities and reinforce learning concepts

#### End User Guides

Certain materials promote Salesforce adoption. These can include:

* **Administrator Guides**: Provides users with System administrator profile information on managing users and configuring the system when new applications are deployed
* **Standard/New User Guides:** Provides existing and new users documentation on the “how to” of any new application pushed into the organization. This documentation includes guidelines and release notes of application training and usage.

This table describes additional materials and training that can assist with user adoption.

|  |  |
| --- | --- |
| **Type** | **Description** |
| **Brown Bag Lunch Seminar** | **Drive adoption through regularly recurring scheduled sessions to provide information and answer questions** |
| **Quick Reference Guide** | **A one-page document that summarizes access to major functions of an application** |
| **System User Guide** | **A short step-by-step guide to customized user functions. (Typically these users have attended Salesforce Admin training)** |
| **On-boarding Guide** | **A short guide that may include contact and help desk information, setup instructions, policies, and training schedules** |
| **Process Workflows** | **A step-by-step flow diagram that graphically depicts who performs different steps in a business process. These can be one-page job aides or can be included in workbooks and system user guides** |
| **Release Notes** | **Provide information to users when new applications are deployed and when updates are made to existing applications. They include details about features and uses for the application and are are written and posted on VA internal sites** |

Data Management

# Intended Audience

This section is intended for VA Salesforce Administrators involved with Data Management in their Salesforce applications.

# High Level Overview

This document outlines the policies relate to backups, archiving, audit logs , and compliance with the National Archives and Records Administration’s (NARA) guidelines. NARA has established guidelines for Federal agencies to preserve certain records and documentation.

# Data Management

## VA Data Backups

VA performs Salesforce Data backups on a regularly scheduled basis to ensure that information is retained and for system recovery purposes.

## Data Archiving

In accordance with NARA guidelines, VA exports data directly from Salesforce in a NARA-approved electronic format. The retention periods for data varies based on the type of data. More information can be found in the NARA Records Control Schedule (RCS).

## Audit Logs

VA maintains and manages an audit log to track Salesforce record access and user account information. Salesforce maintains the login history of users. However, it does not store all the record changes and activity for each record.

# Salesforce Center of Excellence Best Practices

## Exporting Data

VA must comply with NARA guidelines. This requires exporting data from Salesforce at regularly scheduled intervals. To determine data to be backed up, VA considers business requirements in the context of NARA guidelines to identify specific objects and corresponding fields that must be exported and archived.

## Data Export Applications

Salesforce requires third-party applications to support data exports. Two recommended data export applications for Salesforce include:

* **Jitterbit:** A free data migration third-party application used to automate importing and exporting of data that is not Personally identifiable Information (PII).
* **Data Loader:** A Salesforce native application that is used for importing and exporting data.

There are other data export applications for Salesf

User Account Maintenance

# Intended Audience and Overview

The purpose of this section is to ensure System Administrators with user management permissions follow proper The Department of Veterans Affairs (VA) Salesforce configuration settings and rules, and best practices related to user accounts.

The processes in this section ensure only valid business users are added and provided access to the VA Salesforce platform. This process is enforced by the Salesforce Center of Excellence (COE) to ensure standard and consistent experience across all VA Salesforce applications. This User Account Management document defines processes of user accounts management in the VA Salesforce platform. The following processes define and suggest user management practices for licenses, profiles, roles, and general permissions that should be granted as new users are on-boarded or have their permissions modified.

# VA Specific User Management Policies

## Account/License Types

Salesforce Account and License types are used as a starting block to provide access to users within the Salesforce environment. Additional permissions are granted via profiles, roles and permission sets. VA uses limited profiles and roles. The majority of permissions are granted via permission sets. VA Salesforce will use the types in the following table.

|  |  |
| --- | --- |
| **Account Types** | **Purpose** |
| User Internal | Assigned to an individual with an internal Salesforce license. |
| User Community | Assigned to an individual with a Salesforce Community license. Community users have limited access to specific functions in Salesforce. |
| Integration | Explicitly for integration with Salesforce (Web services). |
| User – Sandbox | Gives user access to a test, training, or development environments. By default, users with this access have matching credentials in Production. |
| Administrative User | Assigned to individuals who deploy changes to Salesforce production. |
| Emergency | Temporary or emergency accounts authorized for an accreditation boundary. |
| Guest/Anonymous/Shared | Not allowed by VA. |

**Table 1 - Account Types**

## Profiles

A profile is a collection of permissions and settings that is instrumental in determining a user’s functional access (apps, tabs, object-level permissions), how information is displayed to the user (page layouts, record types, field-level security), and a wide range of other permissions. Profile naming should be standardized across all profiles. The format is: “Application – Role – Permissions” (example, “NPC – User – R”). Profiles are minimally  used at VA.

## Roles

A role controls the level of visibility that users have into your organization’s data. Users at any given role level can view, edit, and report on all data owned by or shared with users below them in the hierarchy, unless your organization’s sharing model for an object specifies otherwise. Roles are minimally used at VA.

## Public Groups

A group consists of a set of users. A group can contain individual users, other groups, or the users in a particular role or territory. It can also contain the users in a particular role or territory plus all the users below that role or territory in the hierarchy. Administrators and delegated administrators can create public groups.

## Permission Sets

Permission sets are a collection of settings assigned to a user to grant them privileges in addition to the existing privileges on a user’s profile. Permission Set naming will be standardized across all permission sets. The format is “Application – Role – Permissions” (example, “Customer Management – User – CRE”).

## Queues

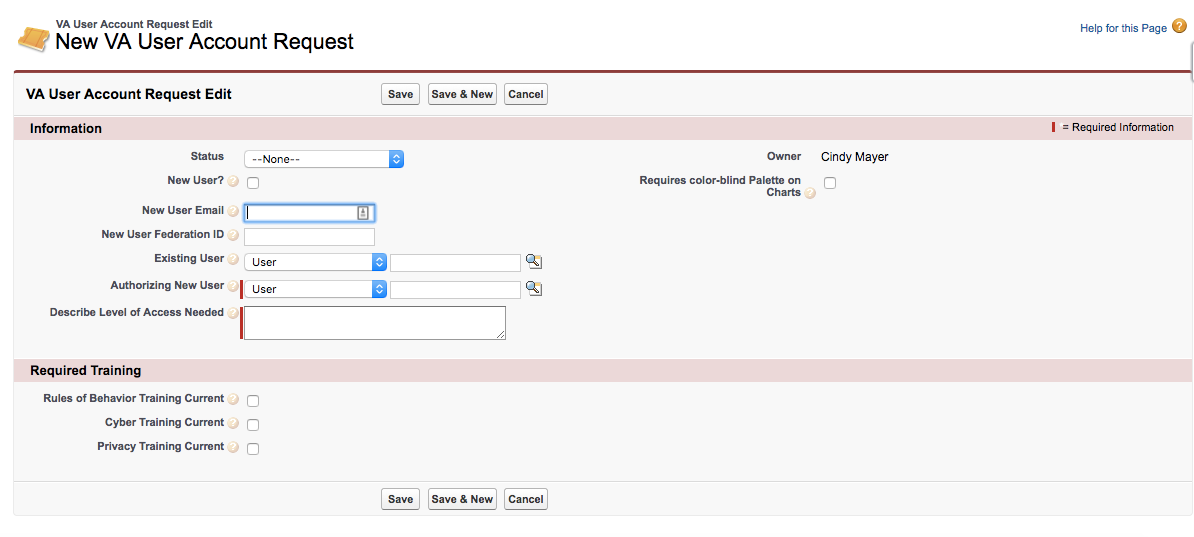
Queues help teams manage leads, cases, service contracts, and custom objects. Once records are placed in a queue manually or through an automatic case or lead assignment rule, records remain there until they're assigned to a user or taken by one of the queue members.

# User Management Best Practices

## Creating & Maintaining Users

There are a few ways that Salesforce accounts are created. a community user account.  Community users can request full access by submitting a New User Account request. Managers may also submit a New User Account Request on behalf on a user. Both of these scenarios require submitting A New User Account Request within Salesforce. A brief description of the applications the user requires access to is included in this request. After submittal, a System Administrator will review the request, and validate the request. Once added, a user is created and/or assigned to the appropriate security group based on job requirements.

The following is a sample VA User Account Request form.



## Authenticating Accounts

VA does not allow users to perform any actions without appropriate identification and authentication. Each user must have an account and password to gain access to the domain and/or local system. No anonymous user, guest, or shared accounts are allowed or used within VA.

SSO Active Directory authentication is via a FIPS 140-2 compliant Virtual Private Network (VPN) connection.

## Deactivating Accounts

To prevent unauthorized access, a Systems Administrator reviews all accounts within VA Salesforce environments every 90 days for account activity. Based on SSP Control AU-2(3) requirements, a VA Salesforce account that is found to be inactive for 90 days is disabled.

Should an account need to be removed due to termination or reassignment of responsibilities, the account will be disabled at the same time (or just before) the employee is notified of their dismissal or upon receipt of resignation, and the VA Salesforce Off-Boarding process will be followed. At this point, the user no longer has access to the VA network and SSO and is therefore unable to access Salesforce.

If an Account is disabled and needs to be unlocked immediately or re-enabled, the user must contact the VA Salesforce Help Desk or a designated representative for assistance. Only Administrators with “Manage User” privileges have the authority to unlock accounts.

# Salesforce COE Best Practices Overview for User Management

The COE, its sub-teams, and related Program Management Office make sure that delivery risk is mitigated, that core standards for platform configuration and customization are upheld, and that a cohesive architectural vision is maintained; ultimately the goal is for the VA to rapidly achieve the core objectives of their business stakeholders and get value maximum from the implementation teams performing the work.

The COE, its sub-teams, and the related Program Management Office play a key role on multiple fronts, including but not limited to:

* Defining the strategy for all of VA’s Salesforce cloud platforms
* Enforcing a governance process that fosters an agile approach
* Improving communication and transparency between business units and development teams
* Providing oversight and technical guidance throughout each release
* Ensuring adherence to technology standards and common application of best-practices across the organization

Ensuring all strategic cloud initiatives are effectively implemented

Issue and Requirements Tracker

# Intended Audience

The primary audience of this section are the Salesforce System Administrators, Project Team Members, and Configuration Management Teams at the Department of Veterans Affairs.

../Downloads/Issue%20and%20Requirement%20Tracking%20-%20Final%20Draft%20-%20VA%20COE.pdf

## Developer Responsibility Tier 2 Defect Resolution

Level: Severe - Users cannot perform essential core functions and there is no workaround

Level: Moderate - Users cannot perform their functions and there is a workaround but it takes the user extra steps to perform this workaround which hinders productivity

Level: Low - Users cannot perform the task in the manner as stated in requirements/User stories but there is an acceptable workaround that can be used until the next release window.