



State of Tennessee
Division of Health Care Finance and Administration

Tennessee Technical Advisory Services
(TN TAS)

Test Management Plan

Version: 1.0
Submitted Date: March 31, 2016

TABLE OF CONTENTS

1. Executive Summary	6
2. Introduction.....	8
2.1. Purpose	8
2.2. Objective.....	8
2.3. Scope.....	8
2.4. Assumptions, Risks & Constraints.....	9
2.5. Referenced Documents	9
3. Test Management Plan Vision.....	11
3.1. Vision and Concept Overview	11
3.2. Strategy for Testing	12
3.3. Testing Approach.....	12
3.4. Test Management Framework.....	13
3.5. Test Management Objectives.....	14
4. Test Management Methodology.....	15
4.1. Test Life Cycle	15
4.2. Test Planning Phase.....	16
4.3. Test Preparation Phase	20
4.4. Test Execution Phase: Unit and System	24
4.5. Test Execution Phase: User Acceptance Testing (UAT).....	30
4.6. Test Execution Phase: Operational Readiness Testing (ORT).....	33
4.7. Test Execution Phase: Beta Testing	36
4.8. O&M Testing.....	40
5. Test Management Standards.....	44
5.1. Test type Plans	44
5.2. Test Schedule.....	45
5.3. Test Resources.....	46
5.4. Test Cases.....	46
5.5. Test Case Documentation and Results	47
5.6. Role Based Access Testing.....	47
6. Test Environment Management	48
6.1. Test Environments.....	48
6.2. Test Environment Management Process	48

7. Test Data Management	49
7.1. Test Data Management Process	49
7.2. Test Data Identification	49
7.3. Test Data Requests	50
7.4. Extract/Transform/Load Test Data	50
7.5. Test Data Assignment and Documentation	50
7.6. Test Data Archiving and Decommissioning.....	51
8. Defect Management.....	52
8.1. Defect Management Process	52
8.2. Defect Status	53
8.3. Defect Management and Test Communication	54
9. Test Tools, Technology and Reporting.....	55
9.1. Test Tool Evaluation and Onboarding Process	55
9.2. Test Tools	55
9.3. Test Environment Monitoring.....	56
9.4. Test Reporting	56
9.5. Test Status Reporting	57
10. Test Metrics and Evaluation	58
10.1. Test Management Metrics.....	58
10.2. Test Evaluation	61
11. Test Management Plan Implementation, Operations and Governance	62
11.1. Change Management.....	62
11.2. Governance.....	62
11.3. Communication Management	62
Appendix A: Acronyms and Definitions	64
Appendix B: SDLC RACI Chart Role Definition	71
Appendix C: Document Stage Definitions.....	73
Appendix D: Project Test Management Plan Template.....	74
Appendix E: Release Test Plan Template.....	86

TABLE OF FIGURES

Figure 1: Test Management Context.....	11
Figure 2: Test Life Cycle	15
Figure 3: Test Environment Management Process.....	48
Figure 4: Test Data Management Process	49
Figure 5: Defect Management Process Flow.....	52
Figure 6: Test Tool Evaluation and Onboarding Process	55
Figure 7: Example of a Test Management Dashboard	58
Figure 8: Test Planning - Test Type Review Process (A)	79
Figure 9: Test Planning - Test Type Review Process (B)	82
Figure 10: Defect Status Lifecycle Workflow	84

TABLE OF TABLES

Table 1: References Documents	9
Table 2: Test Activities by Test Phase within SDLC Phase	13
Table 3: Test Planning RACI Chart	18
Table 4: Test Preparation RACI Chart.....	22
Table 5: Unit and System Test Execution RACI Chart.....	26
Table 6: User Acceptance Testing Execution RACI Chart.....	31
Table 7: Operational Readiness Testing Execution RACI Chart.....	34
Table 8: Beta Testing Execution RACI Chart	36
Table 9: O&M RACI Chart.....	41
Table 10: Acronyms	64
Table 11: Glossary of Terms	65
Table 12: RACI Participants Definition	71
Table 13: In Scope Functional and Non-Functional Test Types	77
Table 14: Out of Scope Test Types.....	78
Table 15: High Level Test Plan by Release.....	85

1. Executive Summary

The Test Management Plan is a guide for the SI Contractor to develop the Project Test Management Plan for the Medicaid Modernization Program (MMP). This document provides a framework for the management of testing activities throughout the Test Life Cycle. This document includes guidelines for developing testing strategy, approach and standards established for testing specific to MMP projects. SI Contractors must reference the Test Management Plan to develop the Project Test Management Plan and the Release Test Plan (if applicable).

Test Management Plan Vision

The SI Contractor must develop the Project Test Management Plan to address the testing needs of the project while leveraging industry standard approaches and methodologies, aligning to the Centers for Medicare and Medicaid Services (CMS) guidance, along with International Software Testing Qualifications Board (ISTQB) and the American Software Testing Qualifications Board (ASTQB) standards. The Project Test Management Plan must clearly articulate the testing strategy, overall approach to testing, and the testing details required to test each release.

Framework and Methodology

The test framework and methodology aligns to the MMP System Development Life Cycle (SDLC) and it describes guidance for the SI Contractor to reference throughout the Test Life Cycle of testing activities, deliverables, roles and responsibilities. The SI Contractor must use the templates when developing the Project Test Management Plan. SI Contractor must document the plan for testing activities throughout the entire Test Life Cycle, such as regression, automation, and manual testing.

Environment, Data, and Defect Management

The testing components and capabilities documented in the Project Test Management Plan is critical to test execution, communication, and coordination. Test environments, test data, and defect management must be well defined and documented to allow effective management and control of testing. Sensitive data used for testing must be protected and must follow all State and Federal regulations. Proper defect management process is required to allow proper communication and defect resolution delivery.

Tools, Technology and Reporting

For MMP projects, tools and technology must be appropriately evaluated and selected to support the execution of the testing. SI Contractor must refer to the defined processes for tools, technologies, and reporting.

Metrics and Evaluation

SI Contractor must refer to the metrics guidance in this document to create and provide to the State for proper test management progress tracking. The State expects the SI Contractor to produce testing metrics for proper evaluation.

SI Contractor must schedule meetings with SPMO, IV&V, and the State to discuss progress, status, issues, escalation, and concerns related directly to test management on a regular basis.

Implementation, Operations, and Governance

SI Contractor must comply and communication changes that impact testing activities and deliverables.

Governance is in place to support and enable the management testing activities include, but may not be limited to, the Project Program Steering Committee, and the Technical Architecture Review Board. These governing bodies are described in detail as part of the Program Governance Management Plan and the communication is established in the overarching Change Management Plan and Communication Management Plan.

2. Introduction

The Test Management Plan is a guide for SI Contractor to develop the Project Test Management Plan for the MMP. The standards used to define the Test Management Plan are based on standards established by the International Software Testing Qualifications Board (ISTQB) and the American Software Testing Qualifications Board (ASTQB). The testing framework and methodology are aligned to Centers for Medicare and Medicaid Services (CMS) guidance.

If an MMP contractor proposes an alternate approach for a project than what is presented in this Management Plan, that approach must satisfy the requirements of this plan, and the contractor must provide justification for the proposed approach as well as demonstrate how the requirements are satisfied.

2.1. Purpose

The purpose of the Test Management Plan is to provide the SI Contractor with guidance, standards, and a framework to develop the Test Management Plan for MMP projects. This Management Plan will also ensure that Project Test Management Plans are developed based on leading practices.

This document provides specifications for the contents required in the Project Test Management Plan, including but not limited to:

- A framework for managing testing activities throughout the Life Case of the project including deliverables and artifacts.
- Roles and responsibilities of stakeholder through the Test Life Cycle.

2.2. Objective

The Test Management Plan will help ensure that the testing activities are performed and managed effectively and accurately throughout the Test Life Cycle for MMP projects.

2.3. Scope

In Scope

The Test Management Plan includes guidance for:

- Methodology that ensures that MMP projects follow defined testing standards and processes including roles and responsibilities.
- Testing components and capabilities that need to be in place to execute the methodology.
- Tools and technologies used to manage testing and associated reporting requirements.
- Metrics required to measure the effectiveness of project level test execution.

Out-of-Scope

The Test Management Plan does not include guidance for testing of non-system related items.

2.4. Assumptions, Risks & Constraints

Assumptions

The life cycle and standard presented in this document assumes that development is based on a waterfall or iterative software development methodology and would need to be updated if an Agile or other software development methodology is used.

Risks

Contractors may have Testing frameworks that need to be aligned with the standards in this document. This risk must be mitigated by aligning a Contractor’s method with these standards at the end of the Planning Phase of a project, prior to the Build Phase.

Constraints

Schedule, resource and budgetary constraints may impact the management of testing activities and should be addressed at the onset of the project.

2.5. Referenced Documents

The Test Management Plan is interrelated with other deliverable as illustrated in the table below. These deliverables have been considered in the design of the Test Management Plan and will continue to be aligned to these and other relevant MMP management plans in future iterations.

Table 1: References Documents

#	Document Name	Content Overview
1	Request for Qualifications for Systems Integration (SI) Services RFQ # 32101-15557	RFQ defining the State’s requirements for an SI Contractor to develop, operate and maintain the Tennessee Eligibility Determination System (TEDS).
2	Centers for Medicare & Medicaid Services (CMS) Testing Framework Overview	CMS guidelines for testing functions to be performed during the Development, Test, Implementation, and Operations and Maintenance (O&M) Phases of the CMS Integrated IT Investment and System Life Cycle Framework.
3	Requirements Management Plan	Provides guidance for developing requirements

#	Document Name	Content Overview
		specifications.
4	Interface/Integration Management Plan	Provides guidance for developing interface/integration project and release level plans.
5	Integrated Operations & Maintenance Planning/Deployment Management Plan	Provides guidance for developing an operations and maintenance plan at a project level.
6	Project and SDLC Management Plan	Provides guidance for developing Project and SDLC Management Plans.

3. Test Management Plan Vision

This section describes the approach for defining the testing framework for addressing the testing needs of the State while leveraging testing approaches and methodologies that are based on CMS guidance and industry leading practices.

3.1. Vision and Concept Overview

The SDLC framework as described in the Project and SDLC Management Plan, is used to manage the activities and deliverables associated to executing a system implementation, such as Tennessee Eligibility Determination System (TEDS). Within the MMP SDLC framework key activities and deliverables are required for testing and will be delivered through the Test Framework. The following figure illustrates the concept of the Test Framework in the context of the MMP SDLC.

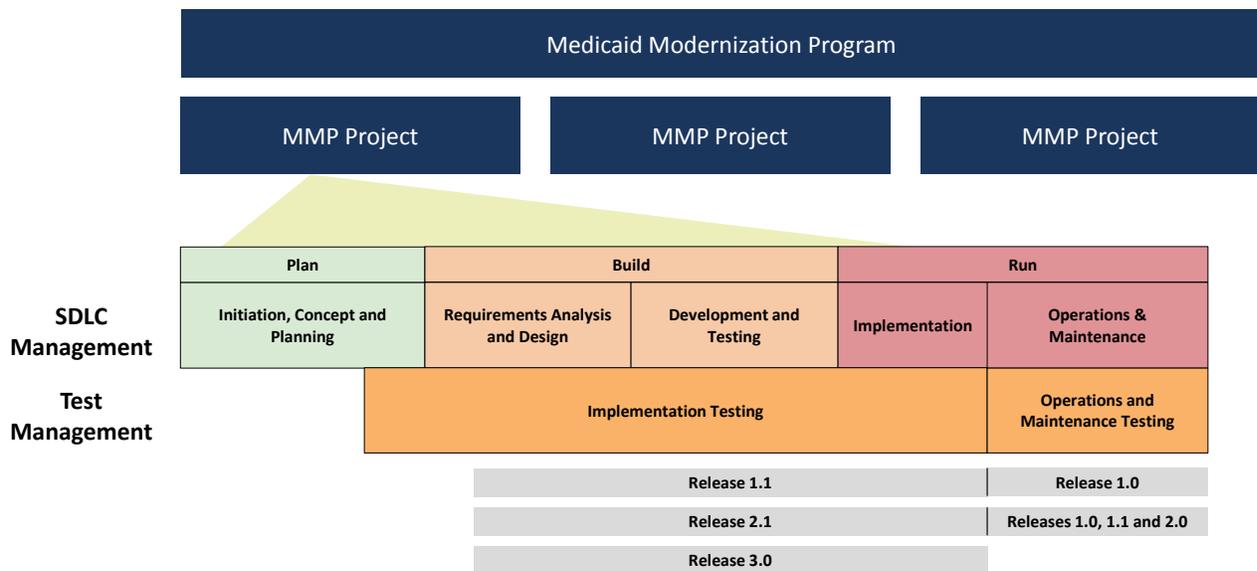


Figure 1: Test Management Context

Implementation testing commences in the latter stages of the Plan phase of the SDLC and continues through to implementation. In alignment to the project release strategy, implementation testing will take place for each release of the project. A **Release Test Plan**, will be developed for each release of the project based on guidance provided by the **Test Management Plan**. The SI Contractor is responsible developing an 'all inclusive' Test Management Plan.

- **Test Management Plan** will focus on the overall approach to testing, including high-level descriptions of how each of the release, functional, and non-functional tests fit within the overall plan. This project level plan will be defined in the Planning Phase of the SDLC.
- **Release Test Plan** will focus on the approach to testing each of the functional and non-functional tests that will be executed during the particular release. The SI Contractor may determine additional test planning documents are required for functional, non-functional, and Operations and Maintenance (O&M) specific test to document the approach for their respective domain. This Release Test Plan will be defined in the Analysis Phase of the SDLC in accordance with the overarching project level release strategy.

O&M testing of a release begins when a fix or maintenance release is required for the current release that is in production. Test cases for a release currently in production should be made available to any new release that is in development to ensure fixes to the current release are added to the code in the next release.

3.2. Strategy for Testing

This document provides guidance to MMP projects to enable them to follow test strategies that encompass successful test objectives. To inform the development of a Test Management Plan, a Test Strategy should be established.

Test Strategy is one of full coverage, meaning Test Plans (as part of the Release Test Plan) will encompass all functionality and business process requirements, security, and non-functional requirements as approved and documented. This strategy will also outline the schedule and test cases which each Test Management Plan and subsequent Release Test Plan should align to.

Test Cases must provide one hundred percent traceability to the approved functional and non-functional requirements developed in accordance with the Requirements Management Plan. As appropriate, SI Testers will manually generate both valid and invalid test data for testing to ensure the system functions as expected

3.3. Testing Approach

Individual Test Plans for each of the functional and non-functional test types will be delivered during the MMP SDLC Design Phase. Test cases will be created by mapping the functionality and detailed test steps to the business processes and non-functional requirements, and will include expected results. All test results, except for Unit Test, will be captured and documented in the Test Management Tool. Testing defects will be captured, documented, and resolved, as appropriate. The test environments will be updated as required, and the test case retested to confirm valid resolution. Test Accreditation and Analysis Reports will be delivered for each of the test types and these reports will contain the specific testing results. Test Plans, Test Cases, and Test Summary and Analysis Reports will be subject to review by all appropriate State stakeholders.

3.4. Test Management Framework

The Test Management Framework, as described in Section 4, specifies a methodology that defines the activities and deliverables required when executing the Test Life Cycle and is aligned to the MMP SDLC. This alignment, testing phases, and associated activities are described at a high level in the following table.

Table 2: Test Activities by Test Phase within SDLC Phase

SDLC Phase(s)	Test Phase	Test Activity
Planning & Design	Planning	Using the Program Test Management Plan as a guide, a high level Project Test Management Plan is developed, with as much details as available at this early phase of the SDLC, regarding objectives, assumptions, acceptance criteria, timeframes, responsibilities, etc. Detailed Test Plans are developed for all testing to update objectives, assumptions, acceptance criteria, timeframes, responsibilities, etc., adding the detail required to resource, schedule and execute the required testing activities.
Development	Preparation	Test scripts, test cases and test data are developed for all tests. For each test case, the expected results, testing process, and participant roles are identified and documented.
Test & Implementation	Execution	In the SDLC Test Phase, the system is tested to confirm that it meets both functional and non-functional requirements. The functional testing confirms that the system supports business processes and produces the expected business results as defined by the functional requirements. The non-functional testing confirms that the system meets operational requirements such as performance, availability, reliability, capacity, security, and accessibility. System defects discovered during testing are managed to ensure that they are fixed, the related test cases re-run to confirm, and regression testing is performed to ensure there are no unintended consequences introduced by defect remediation. Management of the testing and defect resolution processes is supported by summary reports and status reporting. Persistent defect issues are identified through root cause analysis (RCA). In the SDLC Implementation Phase, testing is done to help ensure a smooth transition of the user-accepted system into production mode, not only from a technology perspective, but from people and process perspectives as well.

SDLC Phase(s)	Test Phase	Test Activity
Operations and Maintenance	Operations and Maintenance Testing	During Operations and Maintenance testing, the Test Management Plan should emphasize that testing is done to help ensure seamless upgrades, modifications, defect fixes, and infrastructure changes in production mode.

3.5. Test Management Objectives

The overall objective of the Test Management Framework and described methodology is to not only ensure a quality product, but that the functional and non-functional aspects of the application function in accordance with approved business requirements. In addition, the following key objectives are common to all Test Phases:

- Successfully test the business processes and core functionality (processing transactions against the baseline application configuration, including interfaces).
- Ensure that modules of the application work together to provide the functionality required:
 - Validate that the baseline requirements have been met.
 - Validate that the various work stream requirements work together.
 - Validate the flow of data between upstream and downstream systems.
- Successfully test the development components.
- Successfully test transaction processing against converted data.
- Stress test ensuring that the system is scalable and meets performance, availability, and capacity requirements.
- Complete the testing on time to meet the planned rollout date and ensure a smooth deployment.
- Define, develop, and execute tests that sufficiently meet approved functional, non-functional, and business process.
- Document test results, defects identified, and all defects resolved in accordance to the Defect Management process.
- Ensure that test Entrance and Exit Criteria are successfully achieved.

4. Test Management Methodology

This section describes the different phases of testing, roles and responsibilities, activities, deliverables, and Entry and Exit Criteria between phases.

4.1. Test Life Cycle

The Test Life Cycle, as part of the MMP SDLC, is the basis of the Test Management methodology that will be applied to MMP projects.

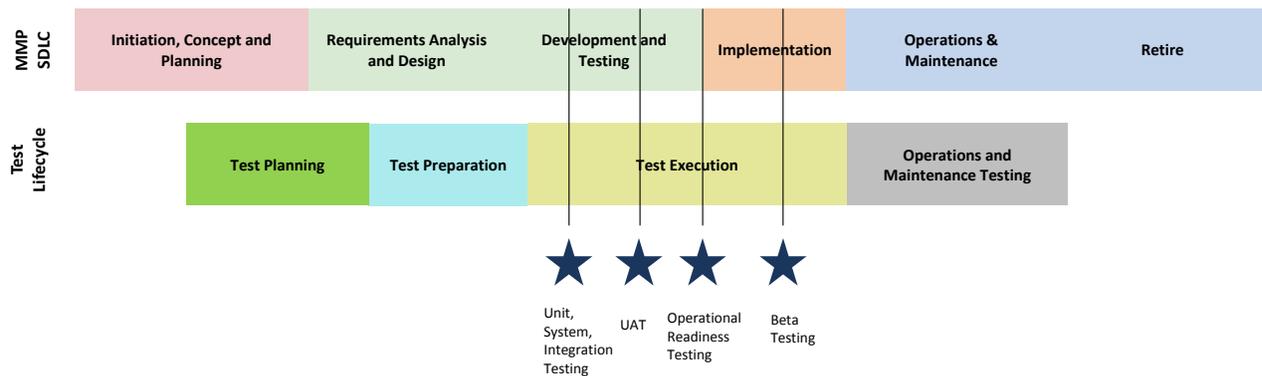


Figure 2: Test Life Cycle

During the Design, Development, and Implementation (DDI) Cycle of a project, the Test Life Cycle is comprised of the Planning, Preparation, and Execution phases, collectively known as Implementation Testing. Implementation testing can be comprised of more than one iteration separated into releases. Once any one release of a proposed solution is implemented, testing is switched to O&M Testing.

O&M Testing takes place in the O&M Phase of the MMP SDLC, and will be focused on one or more iterations that are currently operating in production. It is important that future iterations or defect remediation introduced into a solution during the O&M Phase have the appropriate regression testing performed to ensure that the introduction of new features does not negatively impact current operations.

Each MMP project is required to develop a Test Management Plan that describes the overall approach to testing a solution, and a Release Test Plan that describes the specific testing that will take place during a particular release. Included in the Release Test Plan will be sections that describe the plans for each of the functional test types that will be included in the respective release. While being developed, both test plan types must describe the activities that the four major phases in the Test Life Cycle: Planning, Preparation, Execution, and O&M Testing.

Different types of testing must take place through the entire Test Life Cycle and are described below:

Regression Testing

Regression testing must take place throughout the entire Test Life Cycle and is to be performed to ensure no unintended consequences introduced by defect remediation. This includes regression testing pre and post go live, especially during O&M. SI Contractor must perform regression testing when any changes (direct and indirect) are introduced to the application and interfaces that would impact the end-to-end business process.

The SI Contractor must create the Regression Test Plan and the State Test Manager must review the Regression Test Plan to ensure that the Regression Test Plan is appropriate throughout the Test Life Cycle. The SI Contractor must provide the results to the State Test Manager on a periodic basis to ensure results are aligned to the project progress.

Manual Testing

In the early stages of testing when functionalities and configurations are developed and ready for testing, SI Contractors must test manually to ensure that functionalities are functioning as intended and that the test cases are documented with the appropriated level of details to allow automated test scripts to be developed.

The SI Contractor must coordinate with the State Test Manager to review and determine the Manual Test approach for each test types.

Automation Testing

The SI Contractor must include an Automation Test Plan in the Project Test Management Plan and review it with the State Test Manager. The Automation Test Plan must document the approach, schedule, scope, test environment, automation tools, and test data required.

As the application stabilizes and the regression test inventory increases, and the timeliness of validating test results after defect fixes or changes to the system environment, SI must coordinate with the State Test Manager to proceed with the Automation testing.

4.2. Test Planning Phase

The purpose of the Test Planning Phase is to identify the methodologies and approaches that will be used for the project or release.

For the Test Management Plan, the SI Contractor must detail the approach, tools, resources, and activities for managing the schedule, testing scope, test environments, and test data throughout the life of the project.

For the Release Test Plan, the SI Contractor must describe the approach, resources, and activities necessary for managing the release's schedule, scope, test environments, and test data for executing each functional and non-functional test type that will be included in the particular Release Test Plan.

The SI Contractor's Project and Test Managers will be responsible to coordinate with the Technical Advisory Services (TAS) Contractor, State Test Manager, and any owners of data needed for testing to produce a State-approved Test Management Plan prior to the project's

Design Phase of the MMP SDLC. In similar fashion, the SI Contractor will produce a State-approved Release Test Plan before the project's Development Phase of the MMP SDLC.

The SI Contractor must complete a detailed capacity/performance plan which includes clearly articulated business drivers (economy, population, Federal regulation, Medicaid expansion, etc.) relative to the Eligibility Modernization Project (EMP).

During this Phase, the SI Contractor must establish the Defect Management, Test Data Management, and Test Environment Management Processes that detail not only the processes that will be used to manage defects, test data, and test environments, but also the approaches and resources that will be used to manage such processes.

The SI Contractor must also develop and perform an internal formal Peer Review process and review with business stakeholders and the State and SI Test Managers. The SI must detail the process by which they will document, incorporate, and get approval of the feedback provided during the Peer Review process into their test cases. The SI Contractor will also ensure that there is representation from the development and/or architectural groups to participate in the review sessions.

Completion Criteria

- The SI Contractor must produce a preliminary Test Management Plan.
- If a release approach is determined, the SI Contractor must produce a preliminary Release Test Plan.
- As part of the Test Management Plan, the SI Contractor must define the process for Defect Management, Test Data Management, and Test Environment Management.
- The SI Contractor must define a Peer Review and stakeholder review process for test cases and scripts as part of the Test Management Plan.
- The SI Contractor must complete a Capacity Plan based on business drivers.
- The SI Contractor must have State-approved systems in place for managing defects and test cases that provide for traceability and interoperability with current State requirements and enterprise architecture systems.

RACI definitions can be found in Appendix B of this document.

Table 3: Test Planning RACI Chart

Test Planning	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Activities									
Develop a Test Management Plan that details the approach, tools, resources, and activities for managing the schedule, scope, test environments, and test data throughout the life of the project.	C	A	C	I	I	R	SR	C	C
Develop a Release Test Plan that details the approach, tools, resources, functional test types to be tested, and activities for managing the schedule, scope, test environments, and test data throughout the release of the project.	C	A	C	I	I	R	SR	C	C
Coordinate with the TAS Contractor, State Test Manager, and any owners of data to appropriately capture the processes and activities needed to	I	A	C	I	I	R	C	I	C

Test Planning	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
manage test data in the Test Management Plan									
Acquire, stand-up, and configure tools for managing test cases and defects	I	A	I	I	I	R	I	I	C
Define the process for updating the Requirements Traceability Matrix with necessary test information	I	A	I	I	I	R	SR	I	C
Engage with the STS Infrastructure Team to define test environment needs	I	A	I	I	I	R	C	I	C
Deliverables Produced or Updated During This Phase									
Test Management Plan (Preliminary*)	C	A	C	I	I	R	SR	C	C
Release Test Plan (Preliminary)	C	A	C	I	I	R	SR	C	C
Requirements Traceability Matrix (Updated)	I	A	I	I	I	R	SR	I	C

Test Planning	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Tool Training Material (Preliminary)	I	A	I	I	I	R	I	I	C
Risk Register (Updated)	I	C	I	A	I	R	C	SR	C
Issue Log (Updated)	I	C	I	A	I	R	C	SR	C
Assumptions Log (Updated)	I	C	I	A	I	R	C	SR	C
Decisions Log (Updated)	I	C	I	A	I	R	C	SR	C

*See Appendix C for Document Stage Definitions

4.3. Test Preparation Phase

The purpose of the Test Preparation Phase is to prepare the appropriate resources, systems, and processes so that they are in place for the Test Execution Phase.

The SI Contractor will establish and onboard the Test Team, ensuring that the team is knowledgeable on the processes, tools, and approaches that will be used for testing.

During this phase the SI Contractor will create test cases, including functional, non-functional, and User Acceptance Test (UAT) and get those test cases peer reviewed, and reviewed and approved by the appropriate State or MMP Contractor resources. The SI Contractor will ensure that the test cases are developed to demonstrate both positive and negative conditions. The SI Contractor will submit all test cases to TAS for review on behalf of the State prior to the State Test Manager’s review. The SI Contractor must get non-functional test cases reviewed and approved by the State’s Security SME and/or Infrastructure SME.

The SI Contractor must confirm that the build and deployment of the test environments are complete and ready for testing as part of this phase. In addition, the SI Contractor must provide

the appropriate processes and resources to ensure that test data is created and available to the necessary test resources for manual or automated processes. This should include, at a minimum, verifying that human-to-human, human-to-system, and system-to-system processes are in place, and appropriate personnel have access to systems.

The SI Contractor must train appropriate State and MMP Contractors on SI Contractor tools that will be used to manage test cases, defects, and test reporting.

Entry Criteria

- State-approved systems are in place for managing defects and test cases that provide for traceability and interoperability with current State requirements and enterprise architecture systems.
- A preliminary Test Management Plan or Release Test Plan.
- Test Plan for capacity and performance.
- The plan for acquiring and using test data is clearly articulated and executable as explained in the Test Management or Release Test Plan.

Completion Criteria

- Baselined Test Management Plan or Release Test Plan.
- SI Contractor's Test Team is established and ready to execute tests.
- SI peer reviewed and State-approved Test Cases and Scripts that have been properly traced to requirements.
- SI peer reviewed and State-approved UAT Test Cases and Scripts.
- State and MMP Contractors have necessary access to SI Contractor tools for test case and defect management.
- The appropriate test data is ready for testing.
- The appropriate Test Environments are stood-up and available for testing.

Table 4: Test Preparation RACI Chart

Test Preparation	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Activities									
Create Test Cases	C	A	C	I	I	R	C	I	C
Peer Review Test Cases	C	A	C	I	I	R	SR	I	C
Prepare Test Data	C	A	C	I	I	R	I	I	C
Confirm the build, configuration, and deployment of Test Environments	I	A	I	I	I	R	C	I	C
Perform training on SI Contractor tools for managing test cases and defects	I	A	I	I	I	R	I	I	C
Deliverables Produced or Updated During This Phase									
Test Management Plan (Baselined)	C	A	C	C	I	R	SR	C	C
Release Test Plan (Baselined)	C	A	C	C	I	R	SR	C	C
Test Cases (Prior to Peer Review) (Preliminary)	C	A	C	C	I	R	SR	C	C

Test Preparation	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Test Cases (After Peer Review) (Final)	C	A	C	C	I	R	SR	C	C
Unit, System, Integration Test Scripts (Prior to Peer Review) (Preliminary)	C	A	C	C	I	R	SR	C	C
Unit, System, Integration Test Scripts (After Peer Review) (Final)	C	A	C	C	I	R	SR	C	C
Requirements Traceability Matrix (Updated)	C	A	C	C	I	R	SR	C	C
Beta Test Plan (Preliminary)	C	A	C	C	I	R	SR	C	C
Operational Readiness Plan (Preliminary)	C	A	C	C	I	R	SR	C	C
Tool Training Material (Final)	C	A	C	C	I	R	SR	C	C
Risk Register (Updated)	I	C	I	A	I	R	C	SR	C
Issue Log (Updated)	I	C	I	A	I	R	C	SR	C
Assumptions Log (Updated)	I	C	I	A	I	R	C	SR	C
Decisions Log (Updated)	I	C	I	A	I	R	C	SR	C

4.4. Test Execution Phase: Unit and System

This phase cannot start until the MMP SDLC Validation Readiness Review (VRR) Gate Review has been completed.

The purpose of the Unit and System Phase is to execute and monitor test activities, perform test status reporting, and document test results. System integration testing (SIT) is a high-level software testing process in which the SI testers verify that all related systems can operate in coordination with other systems in the same environment. The system testing process ensures that all subcomponents are integrated successfully to provide expected results.

After the test cases and Cycles are documented, approved, loaded into the Test Management Tool, and the necessary development objects and configuration have been moved to the test environment, the tests are executed by the SI Test Team. The SI Contractor Test Team executes test Cycles according to the test scheduled and by following the test steps associated with each test case. The SI Tester will compare the test results against the expected results. The tester validates the expected test results and documents the actual results by filling in the results within the test management system. If results are not as expected, a defect is initiated.

The SI Test Manager and Lead are responsible for reporting defects to the correct team for resolution, and ensuring that the defect is resolved and retested. Test results are submitted to the SI Test Manager and SI Lead for review and SI sign-off. The approver for test results is the State Test Manager for UAT, Beta, and ORT, otherwise it is the SI Test Manager for any SI testing.

The SI Contractor will log all testing related issues in the project Issue Log. The issues will be reviewed on a regular bases to assess and define the appropriate mitigation plan and owner. Defects will be logged in the Defect Management Tool for triage to determine whether it is a true defect, and whether it needs to be fixed and retested. Any defects that uncovers an issue in requirements, such as missing or misunderstood requirements, will require that the SI Test Manager to escalate the issue to the appropriate resources for a determination on the need for a requirements change. If the outcome of such review results in a requirement that needs to be changed, that requirement change must be approved by the State. The SI Test Manager must update the defect status as appropriate.

The SI Contractor will ensure that the appropriate support resources are available for supporting the testing effort. This can include the execution of batch jobs, such as reports, and data transfers between applications. Any changes to the agreed upon support schedule will require a formal request and approval process between all stakeholders of the change, including the State Test Manager and State Project Manager.

The SI Contractor will ensure that the Test Manager and Test Leads require that testers document the results in a secure manner, each step of the test case in which an action returns a response and is verifiable. Testers must attach appropriate artifacts, including, but not limited to screen shots, reports, mouse clicks/keyboard entry, or query results, that show the completion of each step.

In conjunction with the State's Security SME, and State Test Manager, the SI Contractor must test role-based access test cases against the solution. The SI Contractor must meet with the State Test Manager and State's Security SME to obtain State approval of the roles-based testing plan and the roles being added to the system prior to the configuring of any such roles. The Contractor must certify that role-based permissions are compliant with State, MARS-E, HIPAA, and Segregation of Duties, business roles as defined by business requirements and any relevant regulation rules prior to the start of UAT.

Non-functional testing happens during this phase. The solution is subjected to performance, security, capacity, availability, volume, accessibility, stress, penetration, recovery, and Operational Readiness Testing. The solution must successfully pass any criteria for each of these non-functional tests as described in the Test Management Plan and Release Test Plan.

Entry Criteria

- The SI Contractor has successfully passed the VRR Gate Review
- The Test cases and scripts have been peer reviewed and State-approved
- The SI Contractor's test team has been formed and properly trained on the test tools
- The SI Contractor has a completed, State-approved Test Management Plan or Release Test Plan

Completion Criteria

- The SI Contractor has successfully tested all approved cases and scripts
- All defects have been logged in the defect management tool, fixed where appropriate, and have a disposition; any open defects have been approved by the State
- Testing results show that the solution meet the testing thresholds for successfully test cases and scripts
- Non-functional testing has taken place

Table 5: Unit and System Test Execution RACI Chart

Unit and System Test Execution	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Activities									
Generate a Testing Support Schedule	I	A	I	C	I	R	I	C	C
Execute Test Cases	C	A	C	C	I	R	C	C	C
Document test case results <ul style="list-style-type: none"> Functional/Non-Functional Positive/Negative 	C	A	C	C	I	R	C	C	C
Open and Resolve Test Defects <ul style="list-style-type: none"> Log Defect Triage Defect Communicate Defect Create Fix Package and Deliver Fix Test Fix Regression Update/Close Defect Get Sign-off that defect is fixed 	C	A	C	C	I	R	C	C	C
Provide mitigation plans for defects that cannot be fixed due to budget, missed	A	C	C	C	I	R	C	C	C

Unit and System Test Execution	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
requirements, or other scenarios outside of the SI Contractor's control									
If needed, execute the Change Management processes for defects that result in requirements changes	C	A	C	C	I	R	SR	C	C
Verify Test Results <ul style="list-style-type: none"> Functional/Non-Functional Positive/Negative Outcomes 	C	A	C	C	I	R	SR	C	C
Obtain Approval/Signoff of Test Results	C	A	C	C	I	R	I	C	C
Obtain Approval/Signoff of Test Results	C	A	C	C	I	-	-	-	-
Generate the System Readiness Cert. for UAT	C	A	C	C	I	R	I	C	C
Deliverables Produced or Updated During This Phase									
Test Management Plan (Final)	C	A	C	C	I	R	SR	C	C
Release Test Plan (Final)	C	A	C	C	I	R	SR	C	C

Unit and System Test Execution	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Unit, System, Integration Test Results (Final) Section 508 Assessment Package Test Summary Reports Test Reports	C	A	C	C	I	R	SR	C	C
System Readiness Cert. for UAT (Final) <ul style="list-style-type: none"> Section 508 Assessment Package UAT Test Summary Report UAT Test Reports UAT Test Scripts 	C	A	C	C	I	R	I	C	C
Requirements Traceability Matrix (Updated)	C	A	I	C	I	R	SR	C	C
Testing Support Schedule (Final)	C	A	I	C	I	R	I	C	C
Beta Test Plan (Updated)	C	A	C	C	I	R	SR	C	C
Operational Readiness Plan (Updated)	C	A	C	C	I	R	SR	C	C
O&M Test Plan (Preliminary)	C	A	C	C	I	R	C	C	C

Unit and System Test Execution	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
<ul style="list-style-type: none"> • Regression Test Approach • Automation Test Approach • Negative Test Approach • Test Environments Used Restore Procedures									
Risk Register (Updated)	I	C	I	A	I	R	C	SR	C
Issue Log (Updated)	I	C	I	A	I	R	C	SR	C
Assumptions Log (Updated)	I	C	I	A	I	R	C	SR	C
Decisions Log (Updated)	I	C	I	A	I	R	C	SR	C

4.5. Test Execution Phase: User Acceptance Testing (UAT)

This phase cannot start until the MMP SDLC Implementation Readiness Review (IRR) Gate Review has been completed.

The purpose of the User Acceptance Testing Phase is to allow the State UAT Testing Team and other business testers the opportunity to validate the solution.

The SI Contractor will have resources available to assist the State during the execution of the UAT testing. The SI Contractor will provide written UAT test cases and scripts including test cases that are strung together to represent use case scenarios that are representative of common and complex business scenarios executing over time. The SI Contractor will help update the Requirements Traceability Matrix and support the Defect Management process where needed, such as triaging the defects. The TAS will directly support the State during this phase.

During this phase, The State UAT testing team will perform and document the results of UAT testing. Once the State is satisfied with the UAT test results, ORT testing may commence.

Successful outcome of UAT testing means that the State UAT team approves the solution for implementation into Beta/production environment. The State will provide their approval or rejection of the solution through the Formal Acceptance Test Report.

Entry Criteria

- The SI Contractor has successfully passed the IRR Gate Review, and the state has accepted the solution into UAT testing.
- The SI Contractor has provided approved UAT Test Cases and Scripts.
- The SI Contractor has provided test signoff and certification including a list of outstanding defects that have been given approval to be deployed into UAT.
- The State Test Manager has given authorization to deploy into the UAT environment.
- The State and other MMP Contractors have access to the tool housing the UAT Test Cases and Scripts.

Completion Criteria

- All UAT testing has completed successfully.
- All critical and blocking defects have been resolved.
- All non-critical defects have a remediation or mitigation plan, or has a State-approved work around.
- The State UAT testing team approves the solution to be implemented in to the Beta/production environment.

Table 6: User Acceptance Testing Execution RACI Chart

User Acceptance Testing Execution	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Activities									
Execute Test Cases and Scripts and Document UAT Test Results	A	R	C	I	I	SR	SR	I	C
Open and Resolve Test Defects <ul style="list-style-type: none"> • Log Defect • Triage Defect • Communicate Defect • Create Fix • Package and Deliver Fix • Test Fix • Regression Test • Update/Close Defect • Get Sign-off that defect is fixed 	C	A	C	I	I	R	SR	I	C
Provide mitigation plans for defects that cannot be fixed due to budget, missed requirements, or other scenarios outside of the SI Contractor's control	C	A	C	I	I	R	SR	I	C
If needed, execute the	C	A	C	I	I	R	SR	I	C

User Acceptance Testing Execution	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Change Management processes for defects that result in requirements that need a change									
Verify UAT Test Results	A	R	C	I	I	C	SR	I	C
Develop the Formal Acceptance Test Report	A	SR	C	I	I	R	SR	I	C
Deliverables Produced or Updated During This Phase									
Formal Acceptance Test Report (Updated)	A	SR	C	C	I	R	I	C	C
Requirements Traceability Matrix (Final)	I	A	I	C	I	R	SR	C	C
Beta Test Plan (Final)	C	A	C	C	I	R	SR	C	C
Operational Readiness Plan (Final)	C	A	C	C	I	R	SR	C	C
O&M Test Plan (Updated) <ul style="list-style-type: none"> Regression Test Approach Automation Test Approach Negative Test Approach Test Environments Used Restore Procedures	C	A	C	C	I	R	C	C	C

User Acceptance Testing Execution	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Risk Register (Updated)	I	C	I	A	I	R	C	SR	C
Issue Log (Updated)	I	C	I	A	I	R	C	SR	C
Assumptions Log (Updated)	I	C	I	A	I	R	C	SR	C
Decisions Log (Updated)	I	C	I	A	I	R	C	SR	C

4.6. Test Execution Phase: Operational Readiness Testing (ORT)

This phase cannot start until the MMP SDLC Implementation Readiness Review (IRR) Gate Review has been completed.

The purpose of the Operational Readiness Testing (ORT) Phase is to verify and validate that the intended environment for Beta Testing successfully meets all criteria for pre-production readiness that is appropriately comprehensive and inclusive of all elements required to support applications deployment. This Test Phase may overlap with the Beta Test Phase and can begin as soon as the infrastructure for the production environment is ready.

This testing focuses on the operational readiness of the system that is to become part of the production environment. Functional testing is limited to a small subset of tests that will verify the non-functional aspects of the system. Operational Readiness Testing includes, but is not limited to checking the backup/restore facilities, IT Disaster Recovery procedures, maintenance tasks and periodic checking of security vulnerabilities.

The SI Contractor will work in conjunction with HCFA Security and Infrastructure SMEs, STS, and any other HCFA resources or Contractors to perform ORT including key non-functional test types and procedures in a production-like environment. The SI Contractor must verify that the operations personnel are able to perform the methods and procedures as documented in the O&M Runbook (e.g. responding to alerts, handing failovers, Disaster Recovery procedures).

The SI Contractor must document that the expected Operational Level Agreement and Service Level A requirements are met.

The successful outcome of ORT means that the environment and application is production ready for the Beta Test phase. Once ORT testing has ascertained that the Beta environment is sufficient for implementation, the SI Contractor can begin their deployment implementation process of the application in the Beta test environment. The State Beta testing teams can begin performing Beta Testing in a production environment running in parallel with the legacy system.

Entry Criteria

- The SI Contractor has successfully passed the IRR Gate Review, and the state has accepted the solution into ORT testing.
- The SI Contractor has provided the ORT Test Cases and Scripts.
- The State and other MMP Contractors have access to the tool housing the ORT Test Cases and Scripts.

Completion Criteria

- All ORT testing has completed successfully.
- All critical and blocking defects have been resolved.
- All non-critical defects have been remediation or mitigation plan, or has a State-approved work around.
- The O&M Runbook has been developed and approved by the state.

Table 7: Operational Readiness Testing Execution RACI Chart

Operational Readiness Testing Phase	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Activities									
Execute Test Cases and Scripts and Document ORT Test Results	A	R	C	I	I	SR	SR	I	C

<p>Open and Resolve Test Defects</p> <ul style="list-style-type: none"> • Log Defect • Triage Defect • Communicate Defect • Create Fix • Package and Deliver Fix • Test Fix • Regression Test • Update/Close Defect • Get Sign-off that defect is fixed 	C	A	C	I	I	R	SR	I	C
<p>Provide mitigation plans for defects that cannot be fixed due to budget, missed requirements, or other scenarios outside of the SI Contractor's control</p>	C	A	C	I	I	R	SR	I	C
<p>If needed, execute the Change Management processes for defects that result in requirements that need a change</p>	C	A	C	I	I	R	SR	I	C
<p>Verify ORT Test Results</p>	A	R	C	I	I	C	SR	I	C
Deliverables Produced or Updated During This Phase									
<p>Operational Readiness Report (Final)</p>	C	A	C	C	I	R	SR	C	C
<p>O&M Test Plan (Updated)</p> <ul style="list-style-type: none"> • Regression Test Approach • Automation Test Approach • Negative Test Approach • Test Environments Used • Restore Procedures 	C	A	C	C	I	R	C	C	C
<p>O&M Runbook (Baselined)</p>	C	A	C	C	I	R	C	C	C

Risk Register (Updated)	I	C	I	A	I	R	C	SR	C
Issue Log (Updated)	I	C	I	A	I	R	C	SR	C
Assumptions Log (Updated)	I	C	I	A	I	R	C	SR	C
Decisions Log (Updated)	I	C	I	A	I	R	C	SR	C

4.7. Test Execution Phase: Beta Testing

This phase cannot start until the MMP SDLC Operational Readiness Review (ORR) Gate Review has been completed.

The purpose of the Beta Testing Phase is to test the implementation of the developed application in a production environment with converted data and live interfaces. During Beta Testing the business users validate specific and unique business scenarios. Completion of Beta Testing is necessary for a go-live decision to production.

Beta testing can only start once UAT has been successfully performed. HCFA IS will coordinate Beta Testing. During this phase HCFA Business will have a focus on scenario-based testing and will look for predictable outcomes. HCFA IS will focus on End-To-End testing including interface testing.

The SI Contractor will support this testing through clarification on intended functionality, defect management, environment support, and any test result validation.

Entry Criteria

- The SI Contractor has successfully passed the ORR Gate Review, and the State has accepted the solution into UAT testing.
- The Beta/production environment is ready
- The SI Contractor has provided the Beta Test Cases and Scripts.
- The State and other MMP Contractors have access to the tool housing the Beta Test Cases and Scripts.
- The O&M team must be fully staffed, trained and operationally ready

Completion Criteria

- All testing has completed successfully.
- UAT, ORT and Beta Testing has been successfully completed.
- All critical and blocking defects have been resolved or have a State-approved work around
- All non-critical defects have a remediation or mitigation plan, or has a State-approved work around.
- The State accepts the solution through the Formal Acceptance Test Report.

Table 8: Beta Testing Execution RACI Chart

Beta Testing Phase	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Activities									
Execute Test Cases and Scripts and Document Beta Test Results End-to-end interface testing	A	R	C	I	I	SR	SR	I	C
Execute Test Cases and Scripts and Document Beta Test Results Scenario testing – eligibility & predictable results	AR	SR	C	I	I	SR	SR	I	C
Open and Resolve Test Defects <ul style="list-style-type: none"> Log Defect Triage Defect Communicate Defect Create Fix Package and Deliver Fix Test Fix Regression Test Update/Close Defect Get Sign-off that defect is fixed 	C	A	C	I	I	R	SR	I	C
Provide mitigation plans for defects that cannot be fixed due to budget, missed	C	A	C	I	I	R	SR	I	C

Beta Testing Phase	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
requirements, or other scenarios outside of the SI Contractor's control									
If needed, execute the Change Management processes for defects that result in requirements that need a change	C	A	C	I	I	R	SR	I	C
Verify Beta Test Results	R	A	C	I	I	C	SR	I	C
O&M Test Plan (Baselined) <ul style="list-style-type: none"> Regression Test Approach Automation Test Approach Negative Test Approach Test Environments Used Restore Procedures	C	A	C	C	I	R	C	C	C
Risk Register (Updated)	I	C	I	A	I	R	C	SR	C
Issue Log (Updated)	I	C	I	A	I	R	C	SR	C
Assumptions Log (Updated)	I	C	I	A	I	R	C	SR	I
Decisions Log (Updated)	I	C	I	A	I	R	C	SR	I
Deliverables Produced or Updated During This Phase									

Beta Testing Phase	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
O&M Test Plan (Baselined) <ul style="list-style-type: none"> • Regression Test Approach • Automation Test Approach • Negative Test Approach • Test Environments Used Restore Procedures	C	A	C	C	I	R	C	C	C
Risk Register (Updated)	I	C	I	A	I	R	C	SR	C
Issue Log (Updated)	I	C	I	A	I	R	C	SR	C
Assumptions Log (Updated)	I	C	I	A	I	R	C	SR	C
Decisions Log (Updated)	I	C	I	A	I	R	C	SR	C

4.8. O&M Testing

The purpose of the O&M Testing is to execute, monitor, and report on the ongoing testing of maintenance releases or fixes that happen during O&M.

The SI Contractor must perform regression testing to ensure that any changes implemented in production do not negatively impact existing functionality. This includes, but is not limited to, changes in the infrastructure, database, indirect application, and software or platform upgrades. The SI Contractor will provide automated execution of test cases for regression testing after any fix or maintenance release.

The SI Contractor must update the O&M Runbook to include any changes to the tools, processes and approaches to regression testing, automation testing, and negative testing. In addition, the SI Contractor will maintain and update the O&M regression and automation test scripts, alerts, and monitoring tools.

When executing O&M testing, the scope must include, but is not limited to the testing of appropriate alerts and monitoring, end-to-end execution of the methods and procedures to communicate alerts and monitoring notifications, and restore procedures.

Any new test cases required for new functionality or new non-functional requirements will be made available to testers working on the next release under development. Maintenance releases may be required for a variety of reasons including fixing bugs or security flaws.

The SI Contractor should look for ways to continuously improve their testing capabilities and automated tools.

The Release Test Plan must ensure that the schedule for testing has been coordinated and approved with the SPMO. The Release Test Plan must be approved by the State Test Manager before the project enters the Development Phase of the MMP SDLC.

Entry Criteria

- The solution has been implemented in Production and Beta environments.
- The O&M Test Plan is complete and State-approved.

Completion Criteria

- All test results have been State-approved.

Table 9: O&M RACI Chart

Test Operations and Maintenance	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Activities									
Execute Test Scripts <ul style="list-style-type: none"> • Regression • Automation • Negative 	C	A	C	I	I	R	C	I	C
Document test cases and Cycles <ul style="list-style-type: none"> • Regression • Automation • Negative 	C	A	C	I	I	R	C	I	C
Open and Resolve Test Defects <ul style="list-style-type: none"> • Log Defect • Triage Defect • Communicate Defect • Create Fix • Package and Deliver Fix • Test Fix • Regression Test • Update/Close Defect • Get Sign-off that defect is fixed 	C	A	C	I	I	R	C	I	C

Test Operations and Maintenance	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
Provide mitigation plans for defects that cannot be fixed due to budget, missed requirements, or other scenarios outside of the SI Contractor's control	C	A	C	I	I	R	C	I	C
If needed, execute the Change Management processes for defects that result in requirements that need a change	C	A	C	I	I	R	C	I	C
Verify Test Results <ul style="list-style-type: none"> Regression Automation Negative 	C	A	C	I	I	R	C	I	C
Perform Smoke Testing after each new release	C	A	C	I	I	R	C	I	C
Perform Root Cause Analysis for any defects found as part of regression testing	C	A	C	I	I	R	C	I	C
Deliverables Produced or Updated During This Phase									
O&M Test Plan (Final)	C	A	C	I	I	R	C	I	C

Test Operations and Maintenance	State					Contractor			
	HCFA Business	HCFA IS	HCFA Enterprise Security	Program & Project Management	STS (Infrastructure)	SI	TAS	SPMO	IV&V
<ul style="list-style-type: none"> Regression Test Approach Automation Test Approach Negative Test Approach Test Environments Used Restore Procedures 									
O&M Test Results (Final) <ul style="list-style-type: none"> Test Summary Reports Test Reports 	C	A	C	I	I	R	C	I	C
Risk Register (Updated)	I	C	I	A	I	R	C	SR	C
Issue Log (Updated)	I	C	I	A	I	R	C	SR	C
Assumptions Log (Updated)	I	C	I	A	I	R	C	SR	C
Decisions Log (Updated)	I	C	I	A	I	R	C	SR	C

5. Test Management Standards

This section describes standard definitions for test documents and components used in the Test Life Cycle.

5.1. Test type Plans

This section describes the different types of tests that must be included in a Release Test Plan. The types of tests that have to be completed during an MMP project is determined by the PPA.

The SI Contractor must describe the different types of functional tests that will be executed as a part of the Release Test Plan according to the PPA in the Test Management Plan. The different types of test plans are:

Functional Test Types

- Unit Test
- System Test
- System Integration Test
- End-To-End Test
- Regression Test
- Negative Test
- User Acceptance Test
- Operational Readiness Test
- Beta Test
- Functional Test
- Enterprise Test
- Smoke/Sanity Test
- GUI software Test
- GUI Navigation Test
- Exploratory Test
- Usability Test
- Error Handling Test
- Compatibility Test
- Boundary Test
- Alert/monitoring Test
- Ad-hoc Test

Non-Functional Test Types

- Performance Test
- Security Test
- Capacity Test
- Accessibility Test
- Volume Test

- Recovery Test
- Stress Test
- Operational Readiness Test
- Penetration Test
- Reliability Test
- Scalability Test

O&M Test Types

- Can be a combination of all of the above that are applicable to the maintenance release or fix that is being tested.

Each Release Test Plan must include the following sections:

1. Test Approach, including:
 - a) Regression Test Approach
 - b) Negative Test Approach
2. Automation Test Approach
3. Test Schedule
4. Test Scope
 - a) Types of functional testing to be performed, this must include Reports, Interfaces, Configuration, Enhancements, Forms and Work Flows, and Security
 - b) For O&M testing includes, but is not limited to a combination of all of the above that are applicable to the maintenance release or fix that is being tested.
5. Performance Benchmarks (Non-Functional only)
6. Planned Test Execution Environment
7. Test Data Requirement
8. Test Execution Tools
 - a) For Non-Functional and O&M Test Plans, e.g. needed to simulate a production load, or security tools that simulate penetration
9. Test Type Entry and Exit Criteria
10. Required Staffing Model
11. Transition / Training Plan

5.2. Test Schedule

Each individual Release Test Plan must include a detailed schedule of the testing activities. This test schedule must be rolled up into the Release Plan. A high-level schedule for all testing must be included in the Test Management Plan. All test schedules are coordinated within the Project Management Plan.

Testing support must be scheduled to adequately support testers during test execution, including running of batch jobs such as reports and data transfers between applications.

5.3. Test Resources

The SI Contractor must provide a test team organizational structure, outlining the resources to be allocated to testing. The testing resources are to be identified at the individual level and should include, at a minimum, all test personnel.

5.4. Test Cases

Test cases are used to validate business processes and functional requirements. Test cases are developed in order to test positive, as well as, negative conditions related to business requirements.

At a minimum every test case will contain the following information:

- Test Definition (Purpose)
- Key category attributes such as modules, business areas, and functional areas.
- Test Conditions
- Tester Role (with appropriate security permissions)
- Test Script: Specific steps to follow. Industry standard for a quality test script is 15-20 steps. Each test script could test 4-5 requirements.
- Specific Inputs and/or Test Data necessary to execute the test (Prerequisite)
- Validation Script
- Expected Results
- Test execution details (Execution Date, Executed by, Status, etc.)

For consistency across the entire MMP, a common Test Case unique identification (ID) numbering scheme is required. In addition to tracking and reporting relevancy, the ID must provide traceability between the test cases and the approved business processes and their associated requirements as documented in the Test Case Management Tool and the Requirements Management Plan.

Test cases must outline test conditions, test steps, test data, and expected results for a given test Cycle/case combination. All test cases must be peer reviewed and quality checked to ensure all test cases are complete and of a certain quality. Test case quality is to be tracked for review. Furthermore, test cases must be created so that they can be retested without reaching out to the original creator. This requires that test cases are easily understandable and accurately details the steps required to produce the expected result.

Where applicable, test cases will be developed based on already defined test cases that have been reviewed and can be reused. The SI Contractor will leverage best practices during the test case creation process while making modifications as needed to incorporate tests of specific requirements.

5.5. Test Case Documentation and Results

Test Cases upon execution will conclude with either a Pass, Fail, or Pass with Exception Result. This result will be based on the tester's comparison of the test cases expected results to the actual results received during execution. During testing, SI Testers are to document, in a secure manner, the results of each step in which an action returns a response and is verifiable. Agreed upon documentation techniques are to be followed throughout each testing phase, and are to include the capture of results using artifacts such as:

- Screen Shots
- Reports
- Mouse Clicks/Keyboard Entry
- Query Results

At the completion of each testing phase, all test cases must have been executed and an acceptable test result achieved, based on the test plan Exit Criteria. Test Cases resulting in a 'Passed with an Exception' status will require formal approval from all testing stakeholders, including the SI Test Manager prior to obtaining approval from the State Test Manager. It is also recommended that this review be done on a weekly basis to help reduce the time to review 'Passed with Exception' results.

5.6. Role Based Access Testing

The SI Contractor must execute Role Based Access Testing and validate that the intended security permissions can prevent unauthorized end users from accessing information and performing functionalities within the tested role in the application. The SI Contractor must coordinate with the State's Security SME and the State Test Manager to understand the role based requirements. The SI Contractor must schedule meetings with the State's Security SME and the State Test Manager to review and approve the role based testing approach prior to configuring the roles. The SI Contractor must verify and validate that the role based permission are compliant with the State, MARS-E, HIPAA, Segregation of Duties, business roles as defined by Business Requirements, and any relevant regulation rules prior to the start of UAT.

6. Test Environment Management

This section describes the testing types to be executed in specific environments. For MMP, HCFA requires that UAT is to be conducted in a secure environment. This secure environment is to simulate the Production environment.

6.1. Test Environments

Testing will be conducted in dedicated, controlled test environments as to ensure that developers do not have direct access to these environments. The State has ownership of verifying that the environment is ready for the start of testing and that it has been configured according to the Test Environment Design Plan. The State will also monitor the status of environmental components and schedule changes in the environment during testing to minimize downtime.

Test environment access is to be managed through assignment of test environment roles only to authorized project team members in order to prevent unauthorized data entry into the testing environment that could compromise the test results. Test environment roles may change throughout the Test Life Cycle as appropriate to the Test Phase.

6.2. Test Environment Management Process

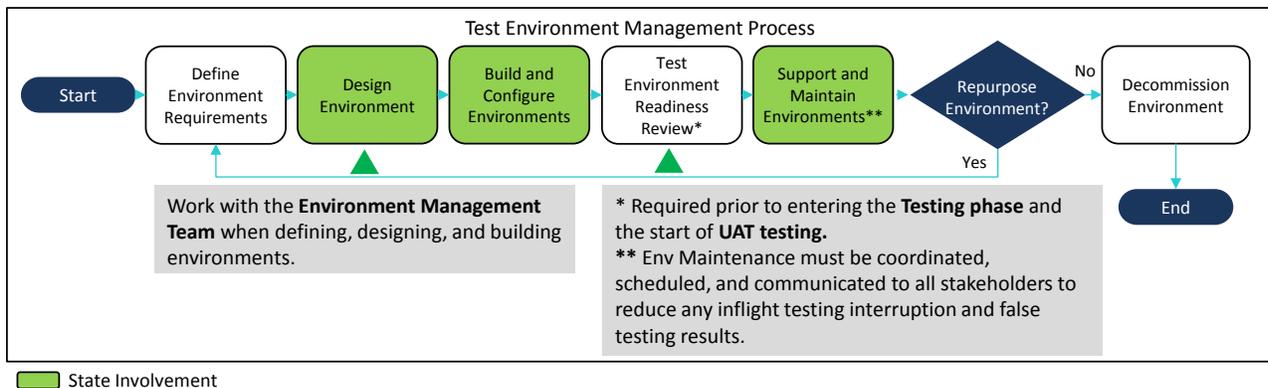


Figure 3: Test Environment Management Process

The SI Contractor is accountable for defining environment requirements and configuring the environment to accept the software. The State, SI Contractor, and STS must work together to establish the test environments.

7. Test Data Management

As part of the Release Test Plan, the SI Contractor must complete a Test Data Management section that specifies the processes and procedures that will be used for test data management.

This section also provides guidance on how to incorporate a Test Data Management section and address the requirement to protect sensitive data. The SI Contractor must work with the State to get approval to access test data, and then work with HCFA data base administrators (DBAs) to extract, transform and load data into the test environments.

7.1. Test Data Management Process

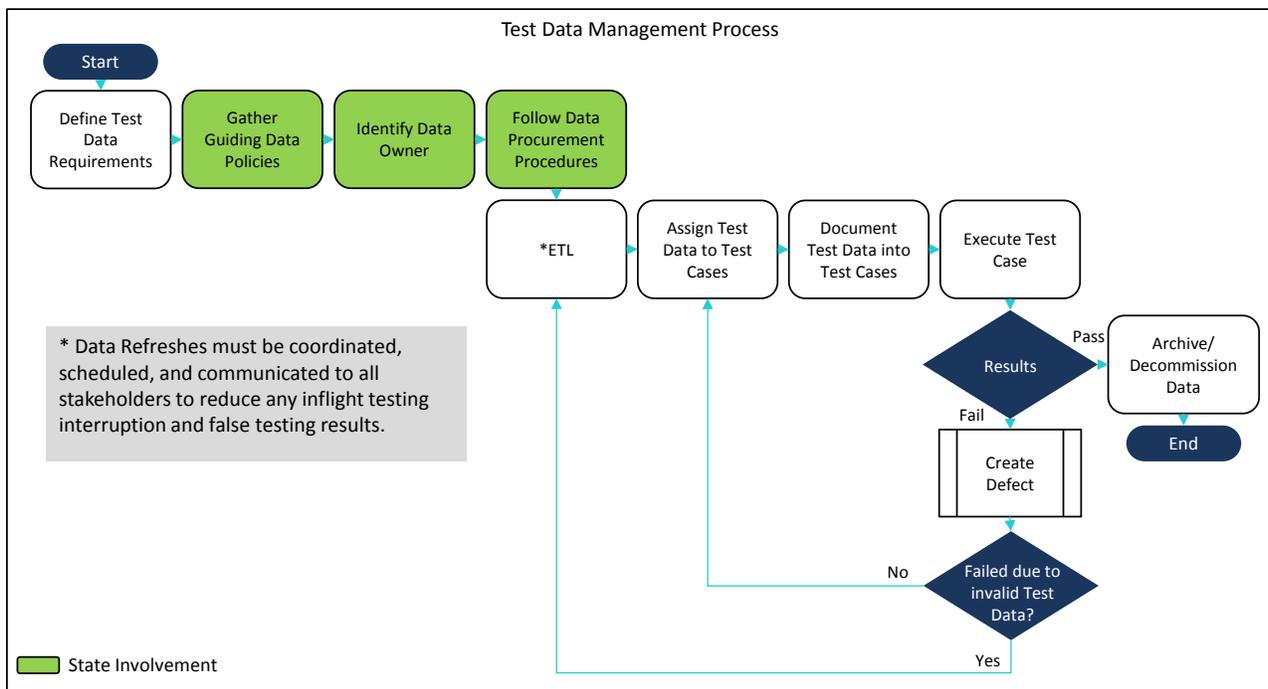


Figure 4: Test Data Management Process

7.2. Test Data Identification

The creation of test cases will be based on the combination of business requirements and business process flows. Throughout test case creation, the SI Test Manager and SI Testers will be responsible for identifying test data and environment requirements and specifications where required for each step in the test case. Testers will work with the State to confirm the appropriate data requirements. A significant amount of the identified test data may need to be created through test case execution.

User input and sample transactions are to be gathered and utilized to satisfy requirements. The Test Leads are to be consulted for suggestions regarding sources of realistic test data. Test

data requirements are determined by the transactions being processed in a test case. Test data accuracy is critical to comprehensive testing to avoid identification of invalid defects.

Gold Copy is a snap shot of the production data that has been prepared to use for testing purposes. Sensitive information is properly protected from unauthorized testers. Gold Copy is also an integrated suite of test data that allows proper referential integrity that allows end to end testing. It should be maintained on a regular basis and includes master data and application configurations.

SI Contractor must include the approach for managing Gold Copy test data in the Data Management Plan. This must include the process to manage, maintain, deploy, and communicate the Gold Copy. In addition, the SI Contractor must define the test data that will be used for each of the test types in their Release Test Plan.

7.3. Test Data Requests

The Test Team will work with all necessary parties to hand off identified test data requirements to the responsible parties for test data creation. This process will be completed with appropriate lead time to accommodate the turnaround time needed to complete the work to satisfy the request. Test Data requests will require a formal request and approval process involving the State Test Manager, data owner, State's Security SME and potentially any technical data base administrator (DBA).

7.4. Extract/Transform/Load Test Data

Once test data requirements have been communicated through the request process, the responsible test data team will extract, transform and load the test data based upon the test data requirements and design specifications into the appropriate environment.

The use of any data must be compliant with any State and Federal regulations.

7.5. Test Data Assignment and Documentation

Results from the Extract, Transform and Load effort will be communicated back to the original requestors and documented so that it may be cross-referenced to the proper test cases/test steps. Where possible, the cross-reference should be completed before execution, but, when test data is not created (or known) until immediately prior to execution, the cross-reference must be completed prior to test execution. This cross-reference is an important component of test support.

Test data once extracted by the SI Contractor is to be reassigned to the appropriate test case/test step for execution. The tester will be responsible for identifying missing test case data and notifying all parties responsible for providing the data. Parties providing data in response to data requests will be responsible for expedited delivery of correct or missing data in the event that data is found to be incorrect or missing.

7.6. Test Data Archiving and Decommissioning

Necessary Archiving and Decommissioning Standards must be identified by the SI Contractor, and must be in compliance with organizational, regulatory and governmental requirements to ensure data and test results are archived in the appropriate manner. It is important to refer back to the test data management strategy for any mandated archiving requirements.

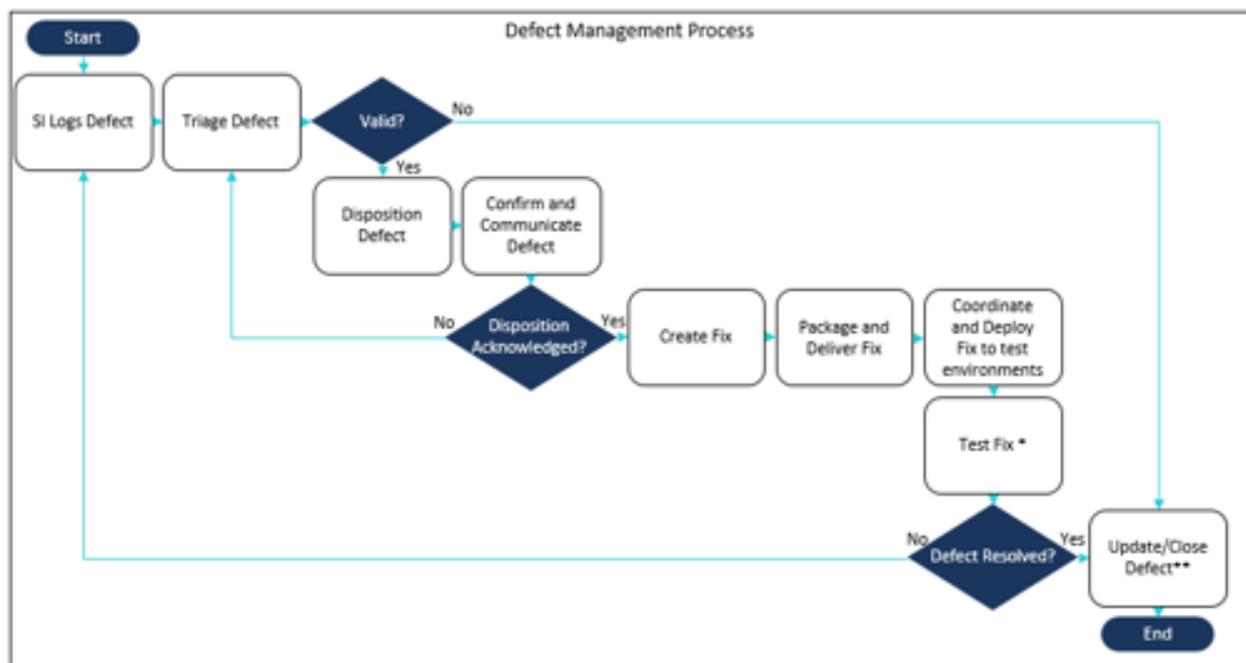
When directed, the entire data store is to be archived or deleted by the SI Contractor. Organizational and regulatory requirements usually drive archival versus decommissioning requirements. Decommissioning a database involves taking the entire data store and/or system offline versus a subset of records. Additionally, a decommissioned database may or may not have an archived equivalent.

As part of the overall Environment Strategy approach, any test data generated or used during the System Integration Test Phase may not be purged from the test environment since that environment could be repurposed for the next Test Phase or release. The test data is to be leveraged to the extent possible during UAT, to reduce any duplication of effort to create new test data.

8. Defect Management

As part of the Release Test Plan, the SI Contractor must complete a Defect Management section that defines how defects will be managed over their Life Case, from identification, categorization, logging, review, resolution, and approved closure. The SI Contractor must communicate the description, status and resolution of defects according to the communication schedule. This section describes the guidance for adding a Defect Management section to the Release Test Plan.

8.1. Defect Management Process



* Note: During UAT, Business Users identify defects and test defect fixes, while SI enters and updates defect logs and statuses.

** Note: Specified filed will be required to be updated in the event a defect is closed. The following fields have been identified as being mandatory:

- Status
- Description (Updated)
- Resolution Action
- Resolution Type

Figure 5: Defect Management Process Flow

A defect will be created for each testing incident (i.e., actual test result does not meet expected result) and maintained/managed in the Defect Management Tool. A defect provides detailed incident information for the time/date of incident occurrence, the test case, test step, test environment, and data used in executing the test step. Additionally, any relevant information regarding the problem (e.g., screen prints, reports, etc.) is to be included. The defect documentation must also include the resolution notes associated with the efforts performed to

successfully investigate, analyze, and resolve the defect, as well as the root cause and final defect resolution.

Once the defect is resolved the tester must fully retest the test case(s) or the series of test steps which caused the unexpected result. To verify the retest of the defect, the team member will provide supporting documentation (e.g., screen prints, reports, etc.) validating the successful retest. In addition, the defect status will need to be updated to reflect the root cause and resolution. Regression testing must be performed to confirm that the defect remediation has not introduced any unintentional consequences.

8.2. Defect Status

The Defect Management Tool provided by the SI Contractor must have the capability of recording status codes for defect tracking.

A SI Manager (typically the SI Test Manager or SI Development Manager or their designate) will assume responsibilities of ensuring defects as a whole are addressed and their resolution is progressing at an acceptable rate.

On a summary level, testing incidents will be consolidated, tracked, and reported for each of the Test Phases in a Defect Management Report. These reports become an audit trail of program and configuration changes throughout the Test Life Cycle. The reports will also include the defect severity, status, and summary resolution of each defect.

The Defect Management Report must include at a minimum:

- Summary information
 - Total number of test cases
 - Percentage of test cases completed and closed
 - Number of defects resulting from testing and category
 - Percentage of defects compared to the total number of applicable test steps performed
- Module/ Functional Area
- Defect ID# – Unique defect identification number
- Defect Status
- Defect Severity
- Description – high level description of the defect
- Nature of Defect – summary description of the unexpected result
- Initiator – person who executed the test case and identified the defect
- Owner – person who owns the defect resolution and all required follow-up actions
- Open – date that the defect was opened for resolution
- Closed – date that the defect was successfully resolved and closed
- Root Cause – a description of the underlying cause of the defect occurring
- Resolution Action – summary of defect resolution action required to close the issue

8.3. Defect Management and Test Communication

Management of the defects involves the proper coordination and dissemination of information as well as any decision-making related to test issues. The information dissemination and decision-making will be coordinated by the SI or State Tester and involve the Test Managers and Test Lead. The Test Lead monitors the open defects to ensure proper severity level, owner, tracking, coordination of retest, and closure.

Frequent communication through regular meetings and reports is a best practice during testing. Defect reporting and when applicable, defect specific meetings, are to be conducted in parallel with test reporting and meeting efforts.

9. Test Tools, Technology and Reporting

As part of the Release Test Plan, the SI Contractor must complete a Tools, Technology, and Reporting section. This section of the document provides guidance on the tools and technologies that can be used to perform Test Management.

9.1. Test Tool Evaluation and Onboarding Process

If an SI Contractor introduces new test tools during an MMP project, the Contractor must adhere to the following process in order for the State to approve the tool.

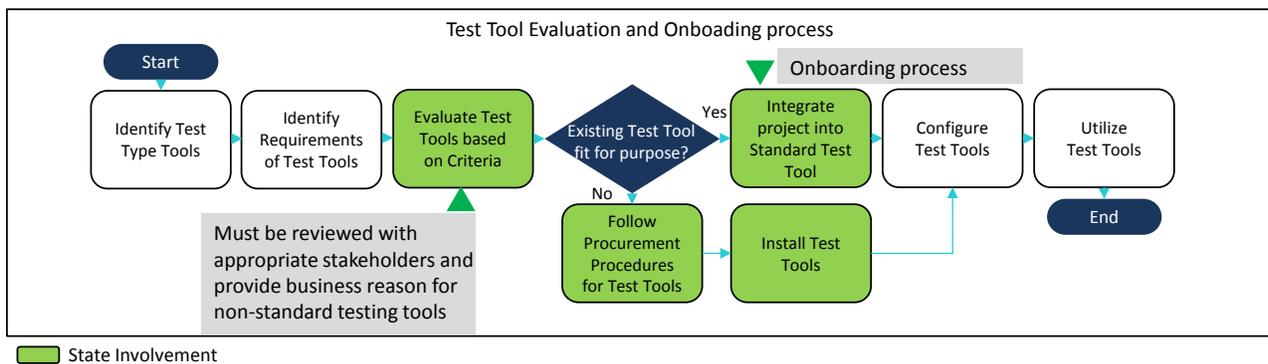


Figure 6: Test Tool Evaluation and Onboarding Process

9.2. Test Tools

The Test Management Tool will be used as the baseline against which the approved business processes and their associated requirements will be manually linked to the test cases. Test Cases are to be consolidated and manually tracked for each Test Phase at a summary level for ease of status reporting and management oversight. The standard test tools include, but are not limited to the following functionality:

- Master test case repository and Test labs for all test types (unit, system, integration, UAT, etc.)
- Tool to trace test cases to requirements
- Defect repository and management
- Tool to trace retests to a defect
- Centralized source for producing testing metrics

The SI Contractor will need to provide UAT and Beta testers access to the agreed upon test and defect management tools prior to and during UAT and Beta execution. The SI will also need to provide necessary documentation and training material in order for the State to properly operate the management tools.

9.3. Test Environment Monitoring

Production-like infrastructure monitoring must be in place with SLA's for downtime, etc. Environment performance is to be monitored and as environment issues are identified, there may be specific performance monitoring required.

9.4. Test Reporting

The Test Managers will review the test reports provided by the SI, as they are delivered. These reports will be evaluated and testing progress will be assessed. There are several levels of Test Reporting:

Daily Progress Reports: Compiled daily during test execution providing day to day progress

Weekly Test Status Reports: Compiled after one week of test execution to provide a short term snapshot of progress.

Test Completion Summary Reports: Reports generated on completion of a testing effort, detailing the overall status of testing.

Analytical Reports: Reports generated over a period of time to evaluate data driven statistics such as Defect Resolution Effectiveness and Root Cause Analysis

Any other reports as requested by the State

Daily Execution Progress Reports

Daily execution progress reports will be created by the SI Test Lead and will be provided to the Test Managers and Project Managers. Additionally, the daily execution report will be posted on the project website. The daily status report will have execution progress percentage, open defects and impact to execution schedule, and major risks / issues.

This execution progress report will be generated based on Test Case Management Tool execution data.

The SI Contractor is required to provide an example of the Test Execution Progress Report.

Weekly Test Status Report

The Weekly Test Status Report will consist of overall progress for the week, outstanding defects and impacts to the overall project. To streamline efforts, it should be prepared through the Project Management Tool or other preferred tool that shows the status of meeting the project's major milestones. The Test Team Lead produces these reports weekly. Weekly overall execution progress status will be provided to the Test Managers and Project Managers.

This status report will be generated based on the Test Case Management Tool data and identify open high risks and issues.

The SI Contractor is required to provide an example of the Weekly Test Status Report.

The report should include, at a minimum, at a minimum the following topics:

- Project Name
- Project Stage
- Test Stage
- Accomplishments
- Upcoming major tasks and milestones
- Test Status
- Report Ending Period
- Key communications
- Issues
- Risks

Test Completion Summary and Analytical Reports

At the conclusion of each test effort, a Test Completion Summary Report and additional Analytical Reports will be prepared by the Test Lead to document test results and satisfaction of exit or acceptance criteria.

The Test Completion Summary Report and Analytical Reports will be generated based on the Test Case Management Tool data. It should identify open high risks and issues and will be used as input into the test approval and sign-off.

The SI Contractor is required to provide an example of the Test Completion Summary and Test Analytical Report.

The reports should include, at a minimum, at a minimum the following topics:

- Test Completion Summary
- Defect Resolution Efficiency (DRE)
- Root Cause Analysis (RCA)
- Release Conclusion
- Test Type conclusions
- Hand-over information
- Recommendations
- Issues
- Risks
-

9.5. Test Status Reporting

Test Management involves proper coordination and dissemination of information as well as any decision making related to test issues. The information dissemination and decision-making will be coordinated by the SI Test Lead and the SI and State Test Managers. The SI Test Lead monitors the testing status to ensure proper progress and will report through the agreed upon communication channels.

10. Test Metrics and Evaluation

This section describes how test executions are to be evaluated and the evaluation results reported.

10.1. Test Management Metrics

This section describes what the SI Contractor is to measure and how, in terms of test management efficiency and effectiveness, to guide test management process improvement. The metrics defined in this section are intended to define the data that are to be collected by the SI Contractor during test execution, to enable production of the reports defined in section 8, and for the dashboard described in this section.

Test Management Dashboard

A test management dashboard is to be provided to illustrate progress of test case execution against planned execution and illustrate outstanding critical and high defects over a period time relative to a major milestone.

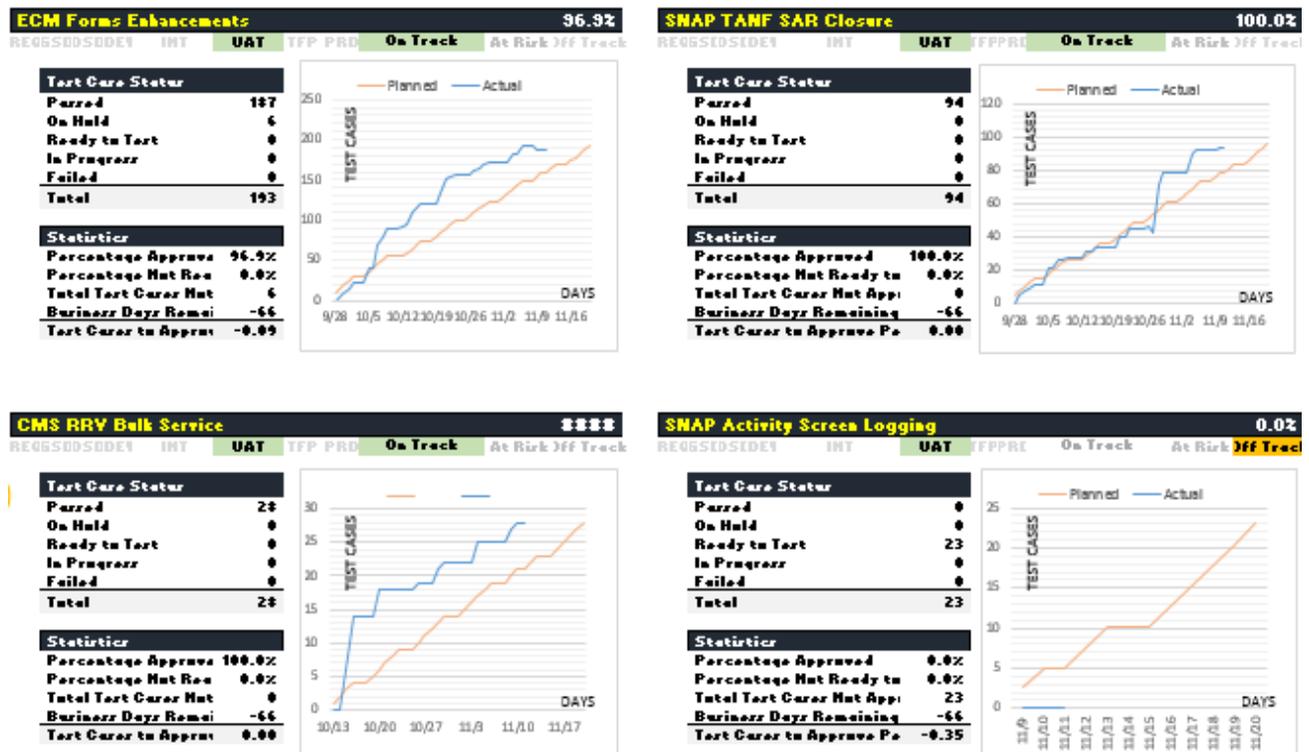


Figure 7: Example of a Test Management Dashboard

Key Test Execution Metrics

There are two types of test execution metrics that are to be provided on at least a daily and a weekly basis.

Testing Progress Metrics

These are metrics that enable the progress of test execution to be monitored, and to help in managing to the testing schedule. The metrics in this category are:

- Number of test cases executed (planned vs actual)
- Number of test cases outstanding (testable) (planned vs actual)
- Number of test cases outstanding (not testable – have been run, but are stopped because of a defect) (planned vs actual)
- Number of test cases passed by functional areas / requirements
- Number of test cases passed with exception
- Number of test cases failed

Defect Metrics

These are metrics that enable the progress of defect resolution to be monitored, and to help in identifying and removing potential bottlenecks in the process. The metrics in this category are:

- Number of defects, by severity and by application area or module impacted, for each defect status category (new, opened, under investigation, etc.)
- Defect Cycle times, by severity and by application area or module impacted, showing the average amount of time a defect remains in a particular stage of a defect resolution Cycle
- Average number of attempts to fix a defect (i.e., fix/retest iterations), by severity and by application area or module impacted

Test Completion Summary Metrics

These are metrics that measure testing and defect resolution effectiveness, and the extent of the impact of the root causes of defects. They are used to identify areas of the SDLC where improvements can be made to reduce the incidence of defects on future releases and other application development projects. As such, they are typically applied sometime after testing has been completed and the application has gone live. These metrics may be applied sooner if there is some urgency in making SDLC process improvements. The metrics in this category are:

- Defect Resolution Efficiency (DRE)
- Defect Leakage
- Root Cause Analysis (RCA) by modules/application areas

These metrics are complex and require additional statistical defect data points that are not listed in section as follows:

- Data logs (date and time) of status changes to defects
- To whom a defect is assigned
- The root cause(s) of a defect

Other Metrics

Other metrics to be considered for inclusion in a Project Test Management Plan are:

- Test Completeness
 - Percentage Test Coverage – Percentage of Number of Test Requirements Vs Functional Areas/Requirements
 - Number of Test Cases Planned Vs Ready for Execution
 - Percent Completion
- Testing Efficiency and Effectiveness
 - Review Efficiency (RE)
 - Test Case Efficiency
 - Test Efficiency and Test Effectiveness
 - Total Time Spent on Preparation Vs Estimated Time
 - Test Efficiency Percentage of test cases passed
 - Defects identified in operational phase vs. prior to deployment
 - Total Unable to Test
 - Defect Discovery Rate
- Test Management
 - Number of Test Cases Executed Vs Test Cases Planned
 - Schedule Adherence
- Developer Effectiveness
 - Defect Density
 - Cumulative Weighted Defect Density
 - Defect Severity Index
 - Percentage Rework
 - First Run Fail Rate
 - Total Number of Defects Closed Vs Total Number of Defects Reopened
 - Total Number of Defects Raised and Closed per period
 - Percentage Test Cases Failed by Module
 - Percentage Test Cases Passed
- Statistical
 - Test Execution Time
 - Number of Test Cases Passed, Failed and Blocked
 - Overall Fail Rate
 - First run fail rate
 - Total Number of Test Cases Passed by Functional Areas / Requirements
 - Defect Distribution Totals by Severity per period
 - Defect Distribution Totals by Functional Areas / Requirements by Severity Period.

10.2. Test Evaluation

In order to monitor test progress, meetings with the SI Contractor, SPMO and the State will be scheduled to discuss progress, status, issues, escalation, and concerns. The frequency will depend upon the needs of the project as defined in the detailed test plan. In addition, test metrics will be collected and reported, including the number of test cases passed vs. failed, test coverage requirements, and percentage of defects open vs. closed over test execution period.

The performance of Test Case Execution and Defect Resolution will be assessed by key stakeholders during test case execution and at the end of each test effort. The frequency of performance evaluations will be defined and documented in the Test Management Plan. The frequency of testing evaluation and reporting may vary per release and testing initiative based on complexity, size, and volume amongst other factors.

Additional Ad Hoc reporting may be requested by the State in order to closely monitor test case execution that may have fallen behind schedule.

11. Test Management Plan Implementation, Operations and Governance

This section describes how the Test Management will be implemented during MMP projects, and operated as a capability within the organization.

Key capabilities need to be in place to enable the management of the Test Management Plan, framework, methodology, along with associated activities and deliverables.

- Change Management
- Governance
- Communication Management

This section will provide additional detail describing the above mentioned capabilities.

11.1. Change Management

A change management process must be in place to manage changes relevant to and impacting testing activities and deliverables.

The SI Contractor will define a process that is aligned to the overall MMP change management expectations as outlined in the Change Management Plan. This Management Plan will provide guidance for the management of changes to testing deliverables, and artifacts.

11.2. Governance

If the Test Management Framework requires changes to its scope, standards, methodology, activities, tools etc., the Test Management Plan will go through appropriate change management controls including Technical Architecture Review Board (TARB). A detailed description and charter for the TARB is included in the Program Governance Management Plan.

11.3. Communication Management

A Communication Management Plan for Testing and Test management should be included as part of the overarching project level Communication Management Plan.

Specific to Testing and Test management, a Communication Management Plan should include, at minimum, the following:

- A mechanism to ensure hand offs between different teams or stakeholders involved in testing is managed (e.g., defect management communication)
- Controls for managing communication associated to the coordination and management of testing activities
- A plan or process for raising and escalating concerns, issues, decision points and for issue resolution

This communication management capability must be established to effectively communicate all testing activities through the entire SDLC.

Appendix A: Acronyms and Definitions

Table 10: Acronyms

Acronym	Definition
BRD	Business Requirements Document
CHIP	Children’s Health Insurance Program
CMS	Centers for Medicare and Medicaid Services
DED	Deliverable Expectations Document
EA	Enterprise Architecture
EA–BOM	Enterprise Architecture Business Operating Model
ELC	Enterprise Life Cycle
EM	Eligibility Modernization
EMP	Eligibility Modernization Project
HCFA	Health Care Finance and Administration
HHS	Health and Human Services
HIPAA	Health Insurance Portability and Accountability Act
ISSC	Information Systems Steering Committee
IT	Information Technology
IT–ELC	Information Technology Enterprise Life Cycle
IV&V	Independent Verification & Validation
MMIS	Medicaid Management Information System
MMP	Medicaid Modernization Program
NHSIA	National Human Services Interoperability Architecture
NIST	National Institutes of Standards and Technology
OeHI	Office of eHealth Initiatives
O&M Runbook	Operations and Maintenance Runbook
PGMP	Program Governance Management Plan
PMO	Project Management Office
PPACA	Patient Protection and Affordable Care Act

Acronym	Definition
PPM	Project Portfolio Management
RACI	Responsible, Accountable, Consulted, Informed
RFQ	Request for Qualifications
RTM	Requirements Traceability Matrix
SDLC	System Development Life Cycle
SI	Systems Integrator
SPMO	Strategic Program Management Office
SRS	System Requirements Specification
STS	Strategic Technology Solutions
TARB	Technical Architecture Review Board
TAS	Technical Advisory Services
TBSM	Tennessee Business Solutions Methodology
XLC	Expedited Life Cycle

Table 11: Glossary of Terms

Term	Definition
Accessibility Test	Accessibility Testing is to ensure that the product is compliant with applicable Section 508 Accessibility Standards identified in the completed Section 508 Product Assessment. Software products (whether COTS, Government Off-the-Shelf (GOTS), or custom-developed software applications) must adhere to Section 508 accessibility and other regulatory requirements governing the use of EIT in accordance with the CMS Policy for Section 508 Compliance. Accessibility Testing is required if the business application has a user interface or produces electronic output for direct access or use by federal employees or the public.
Ad-hoc Test	Testing performed without planning, but using defined test cases.
Alert/monitoring Test	Alert/Monitoring Testing is the type of testing that is done where you purposely end test scenarios/cases in actions that would result in a system alert/message, ensuring that the correct actions are taken at that time. Alerts could be from a user perspective (telling the user that they provided the wrong SSN b/c it begins with 999), or on the system side (the system has received error code 01239, and the system knows how to handle that

Term	Definition
	error code). Monitoring testing is to validate that, for known errors/messages, the system should be ready to monitor for those and intercept them and react to them appropriately.
Automation Test	Software testing that utilizes a variety of tools to automate the testing process and when the importance of having a person manually testing is diminished. Automated testing still requires a skilled quality assurance professional with knowledge of the automation tool and the software being tested to set up the tests.
Availability Test	Availability Testing is primarily concerned with measuring and helping minimize the actual repair time, reducing the repair time, or MTTR will increase available uptime.
Beta Test	Beta testing of the solution is run in a production environment and in parallel with the legacy production system. It allows a comparison of the results of processing between the legacy system and the new solution based on live data in the parallel production environments. It ensures that business sponsors are able to validate that the delivered solution in the target production environment supports all business requirements.
Boundary Test	Boundary Testing consists of testing the extremes of the input domain, e.g. maximum, minimum, just inside/outside boundaries, typical values, and error values
Business Rules Test	Assesses the ability of the system to execute key business rules that were configured based on the requirements. The main objectives are to ensure that all configurable business rules function properly and that all reasonable error conditions are recognized by the application system.
Capacity Test	Testing to determine how many users and/or transactions a given system will support and still meet performance goals.
Compatibility Test	Compatibility Testing validates how well a software performs in a particular hardware/software/operating system/network environment. Backward Compatibility Testing tests the application or software in old or previous versions. Forward Compatibility Testing tests the application or software in new or upcoming versions.
Component Integration Test	Component Integration Testing validates that all software components interact with one another correctly.
Conversion Test	Testing will focus on validating that each conversion program module is successfully completing the task as intended; verify that the data that has been converted to the target application is accurate and functioning properly. Verify that the data can properly flow through the application. Test the end-to-end process of converting data into the new applications

Term	Definition
	and extracting data from legacy.
End-to-End Test	End-to-End Testing tests all of the business application's access or touch points, and data, across multiple business applications and systems, front to back (horizontal) and top to bottom (vertical), to ensure business processes are successfully completed. Testing will be conducted on a complete, integrated set of business applications and systems to evaluate their compliance with specified requirements, and to evaluate whether the business applications and systems interoperate correctly, pass data and control correctly to one another, and store data correctly.
Enterprise Test	Enterprise Testing tests all enterprise business applications that may have direct or indirect touch-points across multiple business applications and systems. Testing will be conducted on a complete, integrated set of enterprise business applications and systems to evaluate their compliance with specified requirements, and to evaluate whether the business applications and systems interoperate correctly, pass data and control correctly to one another, and store data correctly.
Error Handling Test	Assesses the ability of the system to properly process erroneous transactions. The main objectives are to ensure that all reasonably anticipated error conditions are recognizable by the application system, accountability for processing errors has been assigned and that the procedures provide a high probability that the error will be properly corrected, and that reasonable control is maintained over errors during the correction process.
Exploratory Test	Unscripted, improvisational testing that emphasizes the personal freedom and responsibility of the individual tester to continually optimize the quality of his/her work by treating test-related learning, test design, test execution, and test result interpretation as mutually supportive activities that run in parallel throughout the project.
Functional Test	Assesses the input/output functions of a business application against pre-defined functional and data requirements. Each and every functionality of the system is tested by providing appropriate input, verifying the output and comparing the actual results with the expected results. Types of functional testing include: Unit Testing, Smoke Testing, Sanity Testing, Integration Testing, White box testing, Black Box testing, User Acceptance testing, Regression Testing
GUI Navigation Test	Validates the system logic behind when a user navigates from one screen to another. In a GUI system, at each time frame there is an active screen interacting with the user. The active screen when triggered by specific event, will disappear or be deactivated and another one will be loaded in or

Term	Definition
	activated. The two screens are logically connected by the event and such a scenario where the screen focus is shifted is called screen navigation.
GUI Software Test	Testing through the use of a graphical user interface, to ensure it meets agreed upon specifications as defined prior to software development. GUI testing evaluates design elements such as layout, colors, fonts, font sizes, labels, text boxes, text formatting, captions, buttons, lists, icons, links and content.
Interface Test	Interface Testing tests all of the business application's access or touch points, and data, across multiple business applications and systems, front to back (horizontal) and top to bottom (vertical), to ensure business processes are successfully completed. Testing will be conducted on a complete, integrated set of business applications and systems to evaluate their compliance with specified requirements, and to evaluate whether the business applications and systems interoperate correctly, pass data and control correctly to one another, and store data correctly. This testing function is sometimes referred to as End-to-End Integration Testing.
Negative Test	A Negative test will assess the response of the system outside of normal parameters and is designed to assess the system's ability to successfully perform error handling with the unexpected input. The tester uses invalid inputs or imitates unexpected user behavior to expose potential errors and system risk.
Operational Readiness Test	Ensures that the operational readiness is appropriately comprehensive and inclusive of all elements required to support applications deployment. Operational Readiness Testing may include checking the backup/restore facilities, IT Disaster Recovery procedures, maintenance tasks and periodic checking of security vulnerabilities.
Performance Test	A type of testing performed to confirm that the application and technical environment will properly support the anticipated increased transaction volumes, according to an acceptable set of response or elapsed time metrics. It assesses the capacity and throughput of a business application and/or infrastructure in processing time, CPU utilization, network utilization, and memory and storage capacities relative to expected normal (average and peak) user and processing load as defined in the system's requirements document and/or Operation Manual (OM) document. Performance testing can be used to establish a baseline against which future performance tests can be compared against.
Quality Control Test	Testing that determines if the solution is performing and adheres to the predetermined requirements and expectations. The testing will verify the solution was developed as outlined in the documented requirements.

Term	Definition
Recovery Test	Validates how well an application is able to recover from crashes, hardware failures and other similar problems. Recovery testing is the forced failure of the software in a variety of ways to verify that recovery is properly performed.
Regression Test	Selective re-testing to validate that modifications have not caused unintended functional or data results and that the application still complies with its specific requirements.
Reliability Test	Monitors the operational availability of business applications and/or infrastructure, problems/incidents, performance/service level, and capacity utilization of production systems, and will validate the gathered data against expected results (documented in the system's requirement document and/or Operation Manual (OM) document) to ensure that the implemented application or infrastructure performs as expected in production. This testing function is sometimes referred to as Reliability Validation, Burn in Period, Reliability Test, or Extended Reliability Test.
Scalability Test	Testing of a software application for determining its ability to maintain satisfactory performance when any of its non-functional capability requirements is scaled up - be it the user load supported, transaction rate, data volumes, etc. Scalability testing is a subset of performance testing.
Security Compliance Test	A Security Test & Evaluation (ST&E) will validate all applicable security controls defined in the CMS Policy for the Information Security Program. ST&E validates that business application or infrastructure are implemented correctly, operate as intended, and produce the desired outcome with respect to meeting the security requirements for the application or infrastructure. ST&E may include vulnerability scanning, penetration testing, and/or testing security standards and policy.
Smoke/sanity Test	Smoke Tests are shallow and wide, testing all areas of the application without getting deep in focus. Sanity tests are narrow and deep regression tests, testing one or a few areas of functionality.
System Integration Test (SIT)	Testing that validates that all modules and external systems perform together as a system and validates day to day transactions. This Test Phase leverages an end-to-end inter-module testing scheme that qualifies the system based upon the movement of data from the beginning to the end of the system via pre-defined business processes.
System Test	System Testing will validate that the integrated application meets the detailed functional requirements based on an end-to-end process. Test data is used to test each sub-process in an end-to-end format. All new or modified customizations, delivered functionality including application programs, reports, interfaces, etc. are system tested. The goal of this Test

Term	Definition
	Phase is to complete detailed functional testing of the application to ensure that all business requirements are tested in whole.
Unit Test	Unit testing is performed by the application developer/maintainer prior to individual code modules being released from development and moved into a testing area for the broader application. This is the lowest level of confirmation that a report, interface, conversion, enhancement or workflow is validated and functions as expected within the confines of the specification that was developed.
Usability Test	Testing technique typically performed by end users to verify the appropriate level of ease with which a user can learn to operate, prepare inputs for, and interpret outputs of a system or component.
User Acceptance Test	Assesses the overall functionality and interoperability of a business application's solution in an operational mode. UAT allows end users to use the solution in a manner that most resembles actual production use. Testing is performed against the Business Product/Code based on the user's requirements, and may include Training Artifacts and User Manual. UAT may also assess the user's experience with the application to determine if users are able to accomplish their tasks and goals satisfactorily and efficiently to help identify potential problems and possible improvements (i.e., usability testing). Success in UAT will result in a sign-off by the business owner, validating that the business application meets documented requirements.

Appendix B: SDLC RACI Chart Role Definition

This appendix defines roles and responsibilities that key stakeholders have in the SDLC phase activities and deliverables.

The following defines what each letter in the RACI acronym means:

(R) Responsible: Those who are primary responsible for the work to complete the task or deliverable. Only one party must be responsible for any activity, task, or deliverable.

(SR) Shared Responsibility: Those who are charged with completing some supporting work relative to the activity or task. There may be no, one, or multiple SR parties for any activity, task, or deliverable.

(A) Accountable: Those who are accountable for ensuring the correct and thorough completion of the task or deliverable. There should be only one Accountable party for any activity, task, or deliverable.

(C) Consulted: Those whose opinions and input are sought (two-way conversation).

(I) Informed: Those who are kept up to date on progress, often only on completion of the task or deliverable (one-way conversation).

Table 12: RACI Participants Definition

RACI Participants Definitions		
State	Program and Project Management	The management team that includes the Medicaid Modernization Program (MMP) Director and assigned Project Managers.
	HCFA Business	Organizational units that oversee the policies and operations of HCFA business functions, such as member services.
	HCFA IS	HCFA IS provides support for planning, design, implementation and operation of information technologies and methodologies.
	HCFA Enterprise Security	HCFA's enterprise security, includes HCFA & contractor resources responsible for reducing the risk of unauthorized access to systems and data.
	STS (Infrastructure)	Strategic Technology Solutions provides direction, planning, resources, execution, and coordination in managing the information systems needs of the State of Tennessee. STS is a division within the Department of Finance & Administration.

RACI Participants Definitions		
MMP Contractors	TAS	Technical Advisory Services supports and advises the State in completing the Medicaid Modernization Program (MMP) by offering Organizational Change Management and Training, Operations & Maintenance Planning, System Development Life Cycle Advisory Services, Quality Management, and Enterprise Architecture services.
	SPMO	The Strategic Program Management Office provides program and project management support to the State in completing the MMP
	IV&V	Independent Verification and Validation is an independent contractor responsible for verifying that any developed systems perform as designed and will continue to operate correctly in the future. IV&V provides objective evidence that all software requirements have been implemented correctly and completely. This includes evidence that the solution produces the intended results and that all functionality is traceable to solution requirements.
	SI	The System Integrator is responsible for the design, development, testing, implementation, and the operations and maintenance (O&M) of a new system to modernize and enhance eligibility determination, redetermination, and eligibility appeals for the State of Tennessee's Medicaid program (TennCare) and Children's Health Insurance Program (CHIP, known as CoverKids in Tennessee).

Appendix C: Document Stage Definitions

The following table defines the document statuses through the Testing Life Cycle. The source of these definitions are from the [CMS XLC Standard](#).

Acronym	Instance(s)	Definition
P	Preliminary	The first instance of an artifact that contributes to a stage gate review. The template for each review provides detailed expectations of that particular review.
B	Baseline	A version of the artifact that is under initial configuration management control. It is possible but usually difficult to change a baselined artifact. Such a change requires a change request, which ensures that implications to cost, schedule, and technical baselines are addressed. The expectation is that all sections of the artifact have been completed, reviewed, and approved in order to declare a baseline for the artifact.
U	Updated	Security information and artifacts are subject to continuous monitoring and update as needed and/or required.
F	Final	A baseline version of the artifact that is deemed complete and cannot be changed in later phases. It is deemed unchangeable for a particular release of a system. The expectation is that all sections of the artifact have been completed, reviewed, and approved. A final version of an artifact is used for handoff to Operations and Maintenance (O&M).

Appendix D: Project Test Management Plan Template

Project Test Management Plan Template	
Name	Project Test Management Plan Template
Alias(es)	Test Management Plan
Objective	The objective of the Project Test Management Plan Template is to document and track the necessary information required to effectively plan test execution of a software implementation on a consistent basis.
Definition	<p>The Project Test Management Plan Template outlines elements of a test plan, identifies who creates the test plan and how the test plan is used throughout the SDLC (pre and post go live).</p> <p>At the highest level the Project Test Management Plan Document describes:</p> <ul style="list-style-type: none"> • Introduction: provides an overview of the Test Management Plan document, including scope and objectives. • Overall Test Strategy and Approach: addresses test procedures, high level schedule, testing documentation and administration, infrastructure requirements, roles and responsibilities, and assumptions that are shared and common to all of the Tests and testing activities. • Release Test Plans are at a detailed level for the specific Test Types. It will contain the types of testing that will be executed and the associated strategy, approach, and processes to structure, execute, validate, and accept those Tests. Testing deliverables are defined as well as the roles and responsibilities specific to each Test Types. • Test Tracking and Reporting contains information on how testing progress is tracked and communicated, including major Project Plan milestones, the expected testing support schedule, and a list of stakeholder contacts.
Sample	A sample Project Test Management Plan Template is included in the following section.

Project Test Management Plan Template

Table of Contents

1	Introduction
1.1	Purpose
1.2	Scope
1.3	Assumptions, Risks & Constraints
2	Test Approach
2.1	Test Schedule
2.2	Test Resources
2.3	Test Case
3	Test Environment
4	Test Data
5	Test Execution
6	Test Reporting
7	Defect Management
8	High Level Test Plan
9	Test Tools

1 Introduction

The introduction should give a short summary of the project background. The intent is to help those involved in the project to a better understanding of the content of the test plan.

This document provides Project Test Management Plan Template details for <insert name> project. The scope of this document will describe the types of test activities that will be conducted, the strategies of each, tools and environments that will be necessary, data requirements, interfaces, defect tracking and other pertinent testing information.

1.1 Purpose

Instruction: Describe the purpose of the Project Test Management Plan Template. For example:

This Project Test Management Plan Template provides System Integrator (SI) professionals and other MMP participants the necessary information required to effectively plan test execution of a software implementation on a consistent basis throughout the project SDLC. It identifies at a high level the processes, procedures, and deliverables required to successfully test applications, and functionality as defined by the approved baseline business requirements and design documentation. Until all of the identified tests are successfully executed, validated, and the results accepted, the systems will not be migrated to the production environment.

This document must contain:

- Overall test strategy, approach, needs and issues
- Testing scope
- Resource requirements, including external resource dependencies
- Test environment requirements
- Test data requirements
- Overall scheduling and test execution management
- Progress & status monitoring and metrics
- Defect Management
- Entry and exit criteria for each test effort
- Testing tools

1.2 Scope

Instruction: Describe the functional and non-functional types of testing that will be in scope for the project. The test types listed in the in-scope table below are the standard test types to be executed during the testing phase of the SDLC. If any other test types in the table are considered unnecessary, they should be listed in an out-of-scope table with a justification for

each provided, and should be reviewed with the State Test Manager and Infrastructure (Non-Functional) SMEs as per the process outlined below.

Table 13: In Scope Functional and Non-Functional Test Types

IN SCOPE Functional Test Types	IN SCOPE Non-Functional Test Types
• Unit Test	• Performance Test
• System Test	• Security Test
• System Integration Test	• Capacity Test
• End-To-End Test	• Accessibility Test
• Regression Test	• Volume Test
• Negative Test	• Recovery Test
• User Acceptance Test	• Stress Test
• Operational Readiness Test	• Operational Readiness Test
• Beta Test	• Penetration Test
• Functional Test	• Reliability Test
• Enterprise Test	• Scalability Test
• Smoke/Sanity Test	
• GUI software Test	
• GUI Navigation Test	
• Exploratory Test	
• Usability Test	
• Error Handling Test	
• Compatibility Test	

IN SCOPE Functional Test Types	IN SCOPE Non-Functional Test Types
<ul style="list-style-type: none"> Boundary Test 	
<ul style="list-style-type: none"> Alert/monitoring Test 	
<ul style="list-style-type: none"> Ad-hoc Test 	

Of these other test types, those that are determined not necessary should be listed in the out-of-scope table with the justification for each documented. All-out-of scope test types should be reviewed with the State Test Manager and any relevant Infrastructure (Non-Functional) SMEs. A memo of acceptance from the State Test Manager and Infrastructure (Non-Functional) SMEs for the out-of-scope test types must be documented.

The following is an example of an out-of-scope test type table:

Table 14: Out of Scope Test Types

OUT OF SCOPE	Justification	Approver Name
Conversion	<i>No data conversion taking place as part of this project</i>	
Usability Testing	<i>This project is focused on automated services that don't require human interaction</i>	

Test Types are to be identified early in the test planning phase. This will allow for proper adjustment in subsequent test planning and preparation activities. Modifications to the test types prescribed above will require a formal request and approval process including all impacted stakeholders. Additionally, changes will require request and approval from the State Test Manager.

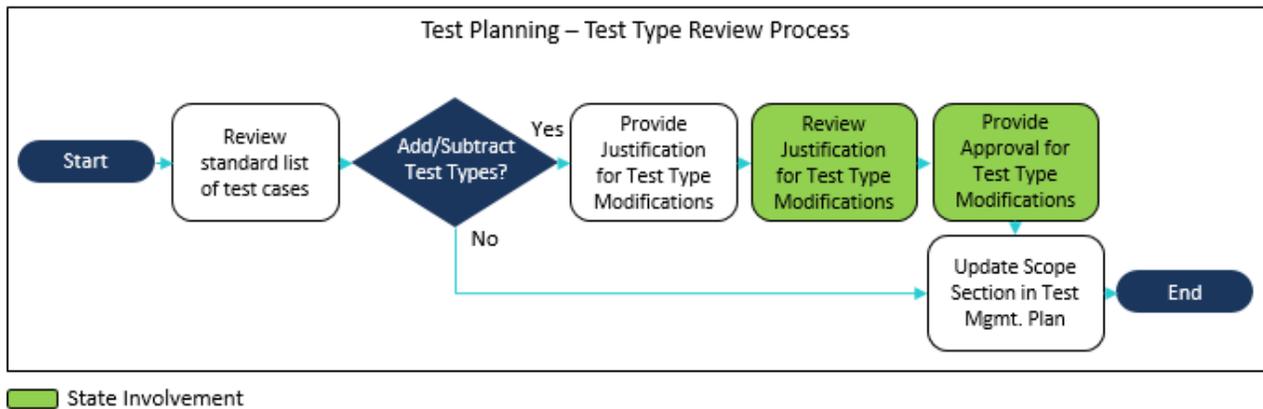


Figure 8: Test Planning - Test Type Review Process (A)

1.3 Assumptions, Risks & Constraints

Instruction: List the Assumptions, Risk & Constraints related to the software components. Areas to consider include but not limited to:

- *Delivery of software by development or third parties*
- *Changes in interfacing systems*
- *Lack of knowledge of supporting systems*
- *Complexity and stability of software and environment*
- *Past experiences*
- *Poor documentation of applications to be tested or supporting testing*
- *In adequate or lack of requirements to test*
- *Governance or Regulatory Standards Requirements*
- *Impacts, limitations and constraints to end user/customer*
- *Other impediments to verifying and validating system characteristics meet defined requirements. System characteristics and their verification and validation techniques are listed below in table format for reference.*

2 Test Approach

Instruction: Describes the testing approach and document the test objectives. If the complete solution is divided up into releases, describe the testing approach per release for all the release.

2.1 Test Schedule

Instruction: Coordinate with the SPMO / State Project Manager to determine and document the detailed testing schedule. Regular consultation between the SPMO / State Project Manager, State Test Manager must be implemented.

- Describe the remediation process and communication when the test schedule slips.
- Document the tool and the cadence of maintaining the test schedule.

2.2 Test Resources

Instruction: Describe the staffing needs (roles, qualifications, capacity, and when they are needed). This planning is not only for the test personnel, but should also include personnel for administrating the test infrastructure, developers and customers.

The staffing needs must encompass the resources required throughout the life of the project.

2.3 Test Case

Instruction: Test Cases are formal deliverables. Describe the follow for test cases:

- *Central location of the master test cases inventory, including tools and navigation path*
- *Which test case(s) address which requirement*
- *Description of the naming convention used*

The SI Contractor is required to create all test cases for each test type in scope. This includes test cases and scenarios for use by the State during User Acceptance Testing (UAT) and Beta Testing. The test cases will provide holistic coverage of interface performance in support of the eligibility determination process. All functional test cases must be reviewed by the TAS Contractor on the State's behalf and approved by the State Test Manager. All non-functional test cases must be reviewed and approved by the Security SME and or Infrastructure (NF) SME.

During test case development, a formal Peer Review process will be followed for the SI Contractor to collaborate with their business stakeholders, representative of the development and/or architectural group reviews, and the State and SI Test Managers. The goal of the Peer Review is to improve the quality of the reviewed test case artifacts by incorporating the business and Test Manager feedback early in the development stages. All business reviewers and the Test Managers should have reviewed the artifacts prior to their scheduled meeting and should come prepared with a list of questions/potential issues or recommended updates.

3 Test Environment

Instruction: Describe the test environment where each test type will be executed. Coordinate with the STS to procure the test environments. Document the process and procedure for communicating test environment support needs. Describe the coordination of environment backups.

The SI Contractor shall provide the environments below. Any deviation from the list of environments below must be supported by the SI Contractor's environment strategy and approach and approved by the State. The SI Contractor may utilize additional application environments based on their development methodology and their understanding of the project roadmap.

- Production Environment with Failover Clustering
- Production Support Environment with Failover Clustering
- Staging/Penetration Testing Environment with Failover Clustering
- Technical Sandbox Environment
- Development Environments
- Development Environment
 - Interface Development Environment
 - Unit/Automated Test Environment
 - Component Integration Test Environment
 - Conversion Development Environment
- Test Environments
 - System Integration Test Environment
 - Functional Test Environment
 - Automated Regression Test Environment
 - Interface Test Environment
 - Performance/Stress Test Environment with Failover Clustering
 - Conversion Test Environment
 - User Acceptance Test Environment
 - Quality Control Testing Environment
 - Beta Test Environment
- Training Environment
- Disaster Recovery Environment with Failover Clustering

4 Test Data

Describe the test data required to execute testing for each test type.

The SI Test Manager and SI Testers will be responsible for identifying test data requirements and specifications for each step in the test case where required. SI Testers will work with the data owner to confirm the appropriate data requirements. A significant amount of the identified test data may be created through test case execution.

5 Test Execution

At the completion of each testing phase, all test cases must have been executed and an acceptable test result achieved, based on the test plan Exit Criteria. Test Cases resulting in a 'Passed with an Exception' status will require formal approval from all testing stakeholders, including the SI Test Manager prior to obtaining approval from the State Test Manager. It is also recommended that this review be done on a weekly basis to help reduce the time to review 'Passed with Exception' results.

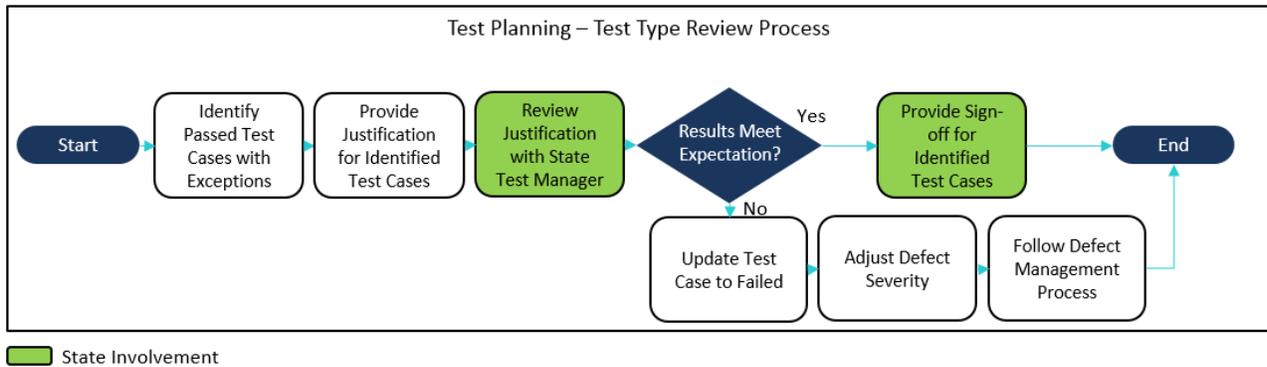


Figure 9: Test Planning - Test Type Review Process (B)

6 Test Reporting

Describe the reporting types with objectives, frequency, intended audience. SI Contractor must include the following types of reports and it must be formatted to the intended audience’s level of detail.

7 Defect Management

Document the SI Contractors intended defect management tool and describe the defect status workflow. The Defect Management Tool provided by the SI Contractor, shall have the capability of creating status codes for defect tracking. The following table is an example of possible status codes. The SI Contractor will update accordingly and review the different statuses with the Test Managers to determine if the statuses are adequate for test metric reporting.

Status	Description
New	By default, the status of the new defect is “New” when added by a tester.
Open	Triage team will change the status from “New” to “Open” after initial triage and review of the defect and assigns appropriate team (developer, infrastructure, and data.)
Investigate	Developer or resolver is assigned to investigate the defect further. The Developer or resolver will try to recreate the defect.
Fix in Progress	Developer or resolver was able to recreate the defect and accepts the responsibility to resolve the defect. Developer or resolver understands how to resolve the defect on a timely basis.
Fix Delivered	Developer or resolver was able to provide a defect fix to address the error
Fix Applied	Development managers update status to “Fix Applied” when the defect fix is applied to the appropriate environments for the tester to retest.

Status	Description
Retest	Development managers update status to “Retest” when the defect has been fixed and applied to the testing environment where the defect was found.
Failed	Tester update the status from “Retest” to “Failed” when the defect retest fails with the same error root cause. If the error is different from the original error, then close the original defect and open a new defect with the new error.
Retest Failed	Tester update the status from “Retest” to “Retest Failed” when the defect failed to be resolved after the initial delivered fix for the original error.
Retest Passed	Tester updates the defect status to “Retest Passed” after re-testing on the new build of the application is successful. Defect is updated with the appropriate documentation and the defect fields are populated correctly.
Rejected	Developer or resolver was not able to recreate the defect and does not accept the responsibility to resolve the defect. Tester has to review the comments and determine if the defect can be closed or a new change request is required. The tester may consult with the Test Lead, Test Managers, or the business client to clarify the intended result.
Closed	Tester updates the status “Closed” after re-testing on the new build of the application is successful and all the appropriate defect attributes are properly populated (e.g. Screen shots, root cause application, root cause category, ...)

The SI Contractor must document the defect status lifecycle workflow. Below is an example.

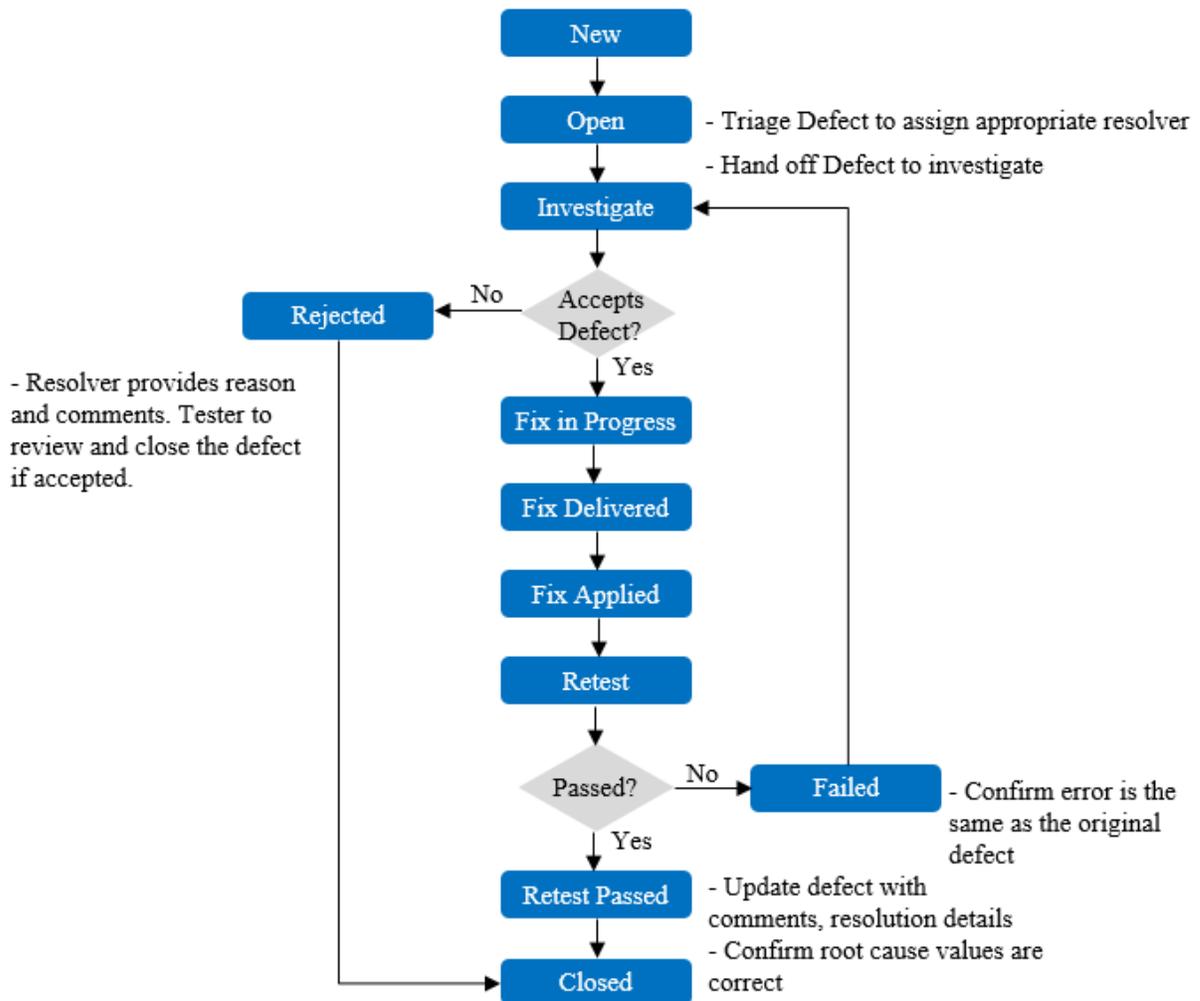


Figure 10: Defect Status Lifecycle Workflow

Describe the roles required to resolve and manage defects. When test problems are identified, the problem is reported and tracked for resolution and completion. The tester is ultimately responsible for tracking and coordinating resolution with the appropriate project individual at the granular level. The SI defect manager will maintain responsibilities of ensuring defects as a whole are addressed and progressing at an acceptable rate.

8 High Level Test Plan

Populate the following table for all the in scope features to be tested (by release if applicable).

Table 15: High Level Test Plan by Release

High Level Test Plan	Features / Scope
	<ul style="list-style-type: none"> • List reports, interfaces, configurations, enhancements, forms, and workflows • <i>Examples:</i> • <i>All forms/screens within each desktop.</i> • <i>All controls within each screen: tabs, entry fields, trigger controls (e.g. button, toolbars, etc), .</i> • <i>Customized field values within each form/screen.</i> • <i>Role based security</i>
	<ul style="list-style-type: none"> • <i>Examples:</i> • <i>All forms/screens within each mobile devices.</i> • <i>Advance reporting</i> • <i>Segregation of Duty</i>
	<ul style="list-style-type: none"> • <i>Examples:</i> • <i>Alerts and Monitoring</i> • <i>Email notification for open tasks</i>
	<ul style="list-style-type: none"> • <i>Examples:</i> • <i>Automation evaluation and qualifications</i>

9 Test tools

Describe the test tools that will be used for testing. These tools include management tools, analytical, automation, performance management tools. SI Contractor must work with the State Test Manager and respective domain SME to review and approve any new testing tools requested.

Appendix E: Release Test Plan Template

Release Test Plan Template	
Name	Release Test Plan Template
Alias(es)	Release Test Plan Template
Objective	The objective of the Release Test Plan Template is to document and track the necessary information required to effectively plan test execution of a software implementation on a consistent basis.
Definition	<p>The Release Test Plan Template outlines elements of a test plan, identifies who creates the test plan and how the test plan is used throughout the SDLC (pre and post go-live).</p> <p>At the highest level the Release Test Plan Document describes:</p> <ul style="list-style-type: none"> • Introduction: provides an overview of the Test Management Plan document, including scope and objectives. • Overall Test Strategy and Approach: addresses test procedures, high level schedule, testing documentation and administration, infrastructure requirements, roles and responsibilities, and assumptions that are shared and common to all of the Tests and testing activities. • Test Plans are at a detailed level for the specific Test Types. It will contain the type of testing that will be executed and the associated strategy, approach, and processes to structure, execute, validate, and accept these Tests. Testing deliverables are defined as well as the roles and responsibilities specific to each Test Types. • Test Tracking and Reporting contains information on how testing progress is tracked and communicated, including major Project Plan milestones, the expected testing support schedule, and a list of stakeholder contacts.
Sample	A sample Release Test Plan Template is included in the following section.

Release Test Plan Template

Table of Contents

1	Introduction
1.1	Purpose
1.2	Scope
1.3	Assumptions, Risks & Constraints
2	Test Approach
2.1	Test Schedule
2.2	Test Resources
2.3	Test Case
3	Test Environment
4	Test Data
5	Test Execution
6	Test Reporting
7	Defect Management
8	Test Plan Details For Release
9	Test Tools

1 Introduction

The introduction should give a short summary of the project background. The intent is to help those involved in the project to a better understanding of the content of the test plan.

This document provides Release Test Plan Template details for <insert name> project – Release <insert release number>. The scope of this document will describe the types of test activities that will be conducted, the strategies of each, tools and environments that will be necessary, data requirements, interfaces, defect tracking and other pertinent testing information.

1.1 Purpose

Instruction: Describe the purpose of the Release Test Plan Template. For example:

This Release Test Plan Template provides System Integrator (SI) professionals and other MMP participants the necessary information required to effectively plan test execution of a software implementation on a consistent basis throughout the project SDLC. It identifies at a high level the processes, procedures, and deliverables required to successfully test applications, and functionality as defined by the approved baseline business requirements and design documentation. Until all of the identified tests are successfully executed, validated, and the results accepted, the systems will not be migrated to the production environment.

This document must contain:

- Overall test strategy, approach, needs and issues
- Testing scope for the Release
- Resource requirements, including external resource dependencies
- Test environment requirements for the Release
- Test data requirements for the Release
- Overall Release scheduling and test execution management
- Progress & status monitoring and metrics
- Defect Management
- Entry and exit criteria for each test effort
- Testing tools

1.2 Scope

Instruction: Describe the functional and non-functional types of testing that will be in scope for the release. The test types listed in the in-scope table below are the standard test types to be executed during the testing phase of the SDLC. If any other test types in the table are considered unnecessary, they should be listed in an out-of-scope table with a justification for each provided, and should be reviewed with the State Test Manager and Infrastructure (Non-Functional) SMEs as per the process outlined below.

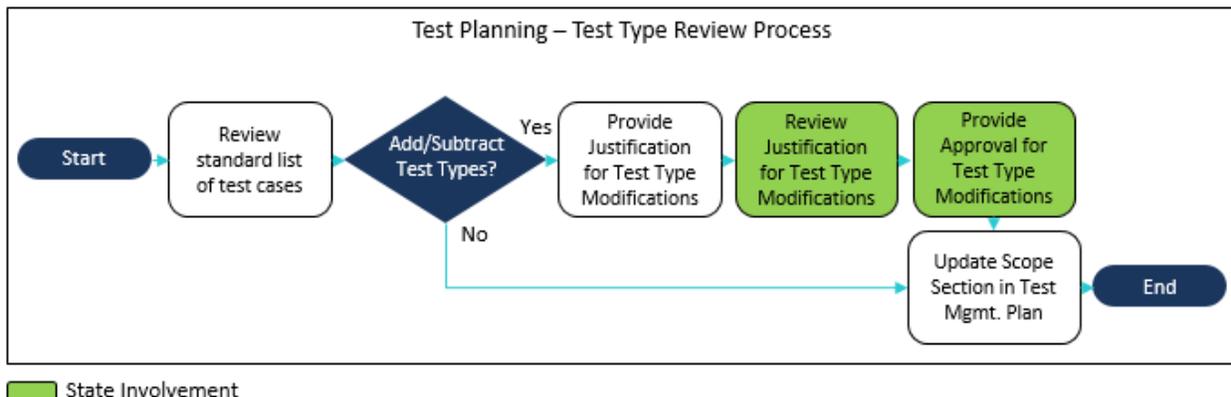
IN SCOPE Functional Test Types	IN SCOPE Non-Functional Test Types
• Unit Test	• Error Handling Test
• System Test	• Compatibility Test
• System Integration Test	• Boundary Test
• End-To-End Test	• Alert/monitoring Test
• Regression Test	• Ad-hoc Test
• Negative Test	• Security Test
• User Acceptance Test	• Capacity Test
• Operational Readiness Test	• Accessibility Test
• Beta Test	• Volume Test
• Functional Test	• Recovery Test
• Enterprise Test	• Stress Test
• Smoke/Sanity Test	• Operational Readiness Test
• GUI software Test	• Penetration Test
• GUI Navigation Test	• Reliability Test
• Exploratory Test	• Scalability Test
• Usability Test	•

Of these other test types, those that are determined not necessary should be listed in the out-of-scope table with the justification for each documented. All-out-of scope test types should be reviewed with the State Test Manager and any relevant Infrastructure (Non-Functional) SMEs. A memo of acceptance from the State Test Manager and Infrastructure (Non-Functional) SMEs for the out-of-scope test types must be documented.

The following is an example of an out-of-scope test type table:

OUT OF SCOPE	Justification	Approver Name
Conversion	No data conversion taking place as part of this release	
Usability Testing	This project is focused on automated services that don't require human interaction	

Test Types are to be identified early in the test planning phase. This will allow for proper adjustment in subsequent test planning and preparation activities. Modifications to the test types prescribed above will require a formal request and approval process including all impacted stakeholders. Additionally, changes will require request and approval from the State Test Manager.



1.3 Assumptions, Risks & Constraints

Instruction: List the Assumptions, Risk & Constraints related to the software components. Areas to consider include but not limited to:

- Delivery of software by development or third parties
- Changes in interfacing systems
- Lack of knowledge of supporting systems
- Complexity and stability of software and environment
- Past experiences
- Poor documentation of applications to be tested or supporting testing
- In adequate or lack of requirements to test
- Governance or Regulatory Standards Requirements
- Impacts, limitations and constraints to end user/customer

- *Other impediments to verifying and validating system characteristics meet defined requirements. System characteristics and their verification and validation techniques are listed below in table format for reference.*

2 Test Approach

Instruction: Describes the testing approach and document the test objectives. If the complete solution is divided up into releases, describe the testing approach per release for all the releases.

2.1 Test Schedule

Instruction: Coordinate with the SPMO / State Project Manager to determine and document the detailed testing schedule for the Release. Regular consultation between the SPMO / State Project Manager, State Test Manager must be implemented.

Describe the remediation process and communication when the test schedule slips.

Document the tool and the cadence of maintaining the test schedule.

2.2 Test Resources

Instruction: Describe the staffing needs (roles, qualifications, capacity, and when they are needed). This planning is not only for the test personnel, but should also include personnel for administrating the test infrastructure, developers and customers.

The staffing needs must encompass the resources required throughout the life of the project.

2.3 Test Case

Instruction: Test Cases are formal deliverables. Describe the follow for test cases:

- *Central location of the master test cases inventory, including tools and navigation path*
- *Which test case(s) address which requirement*
- *Description of the naming convention used*

The SI Contractor is required to create all test cases for each test type in scope. This includes test cases and scenarios for use by the State during User Acceptance Testing (UAT) and Beta Testing. The test cases will provide holistic coverage of interface performance in support of the eligibility determination process. All functional test cases must be reviewed by the TAS Contractor on the State's behalf and approved by the State Test Manager. All non-functional test cases must be reviewed and approved by the Security SME and or Infrastructure (NF) SME.

During test case development, a formal Peer Review process will be followed for the SI Contractor to collaborate with their business stakeholders, representative of the development and/or architectural group reviews, and the State and SI Test Managers. The goal of the Peer Review is to improve the quality of the reviewed test case artifacts by incorporating the business and Test Manager feedback early in the development stages. All business reviewers and the Test Managers should have reviewed the artifacts prior to their scheduled meeting and should come prepared with a list of questions/potential issues or recommended updates.

3 Test Environment

Instruction: Describe the test environment where each test type will be executed. Coordinate with the STS to procure the test environments. Document the process and procedure for communicating test environment support needs. Describe the coordination of environment backups.

The SI Contractor shall provide the environments below. Any deviation from the list of environments below must be supported by the SI Contractor's environment strategy and approach and approved by the State. The SI Contractor may utilize additional application environments based on their development methodology and their understanding of the project roadmap.

- *Production Environment with Failover Clustering*
- *Production Support Environment with Failover Clustering*
- *Staging/Penetration Testing Environment with Failover Clustering*
- *Technical Sandbox Environment*
- *Development Environments*
- *Development Environment*
- *Interface Development Environment*
- *Unit/Automated Test Environment*
- *Component Integration Test Environment*
- *Conversion Development Environment*
- *Test Environments*
- *System Integration Test Environment*
- *Functional Test Environment*
- *Automated Regression Test Environment*
- *Interface Test Environment*
- *Performance/Stress Test Environment with Failover Clustering*
- *Conversion Test Environment*
- *User Acceptance Test Environment*
- *Quality Control Testing Environment*
- *Beta Test Environment*

- Training Environment
- Disaster Recovery Environment with Failover Clustering

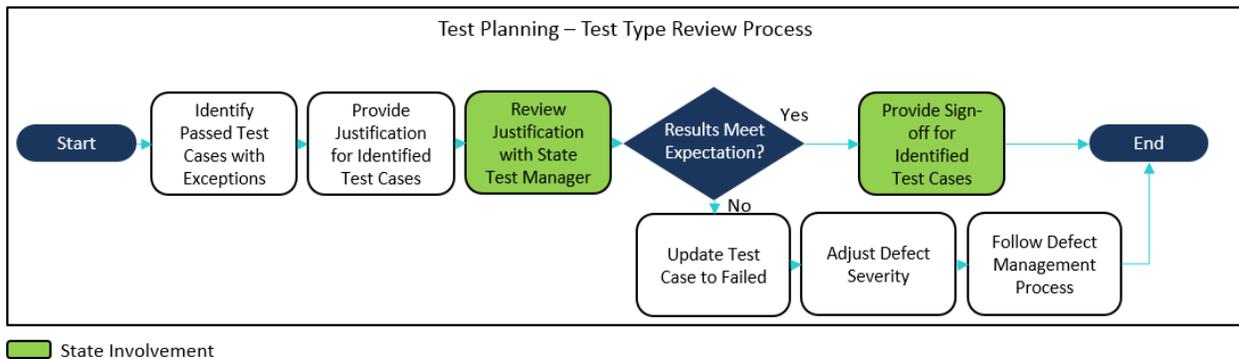
4 Test Data

Describe the test data required to execute testing for each test type.

The SI Test Manager and SI Testers will be responsible for identifying test data requirements and specifications for each step in the test case where required. SI Testers will work with the data owner to confirm the appropriate data requirements. A significant amount of the identified test data may be created through test case execution.

5 Test Execution

At the completion of each testing phase, all test cases must have been executed and an acceptable test result achieved, based on the test plan Exit Criteria. Test Cases resulting in a 'Passed with an Exception' status will require formal approval from all testing stakeholders, including the SI Test Manager prior to obtaining approval from the State Test Manager. It is also recommended that this review be done on a weekly basis to help reduce the time to review 'Passed with Exception' results.



6 Test Reporting

Describe the reporting types with objectives, frequency, intended audience. SI Contractor must include the following types of reports and it must be formatted to the intended audience's level of detail.

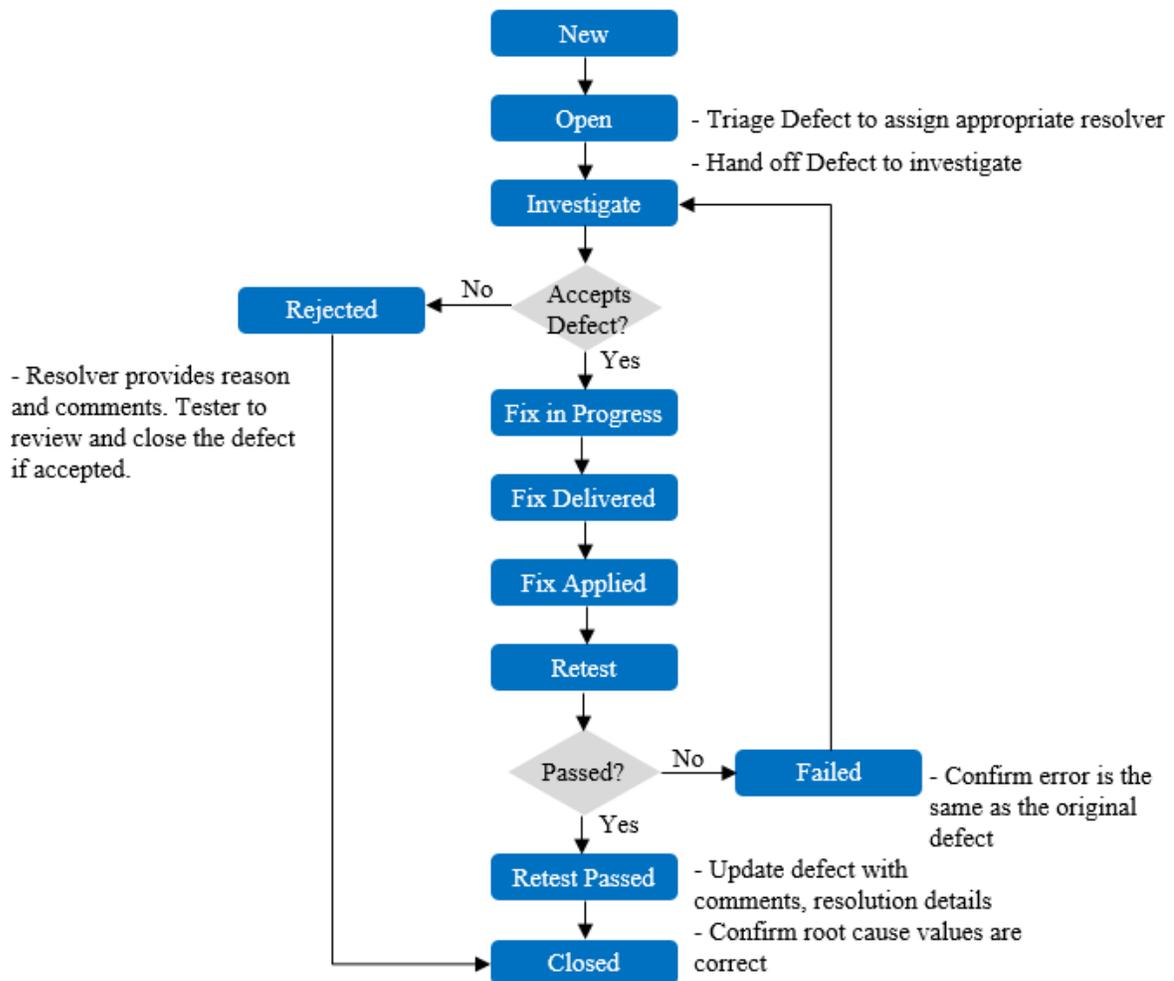
7 Defect Management

Document the SI Contractors intended defect management tool and describe the defect status workflow. The Defect Management Tool provided by the SI Contractor, shall have the capability of creating status codes for defect tracking. The following table is an example of possible status

codes. The SI Contractor will update accordingly and review the different statuses with the Test Managers to determine if the statuses are adequate for test metric reporting.

Status	Description
New	By default, the status of the new defect is “New” when added by a tester.
Open	Triage team will change the status from “New” to “Open” after initial triage and review of the defect and assigns appropriate team (developer, infrastructure, and data.)
Investigate	Developer or resolver is assigned to investigate the defect further. The Developer or resolver will try to recreate the defect.
Fix in Progress	Developer or resolver was able to recreate the defect and accepts the responsibility to resolve the defect. Developer or resolver understands how to resolve the defect on a timely basis.
Fix Delivered	Developer or resolver was able to provide a defect fix to address the error
Fix Applied	Development managers update status to “Fix Applied” when the defect fix is applied to the appropriate environments for the tester to retest.
Retest	Development managers update status to “Retest” when the defect has been fixed and applied to the testing environment where the defect was found.
Failed	Tester update the status from “Retest” to “Failed” when the defect retest fails with the same error root cause. If the error is different from the original error, then close the original defect and open a new defect with the new error.
Retest Failed	Tester update the status from “Retest” to “Retest Failed” when the defect failed to be resolved after the initial delivered fix for the original error.
Retest Passed	Tester updates the defect status to “Retest Passed” after re-testing on the new build of the application is successful. Defect is updated with the appropriate documentation and the defect fields are populated correctly.
Rejected	Developer or resolver was not able to recreate the defect and does not accept the responsibility to resolve the defect. Tester has to review the comments and determine if the defect can be closed or a new change request is required. The tester may consult with the Test Lead, Test Managers, or the business client to clarify the intended result.
Closed	Tester updates the status “Closed” after re-testing on the new build of the application is successful and all the appropriate defect attributes are properly populated (e.g. Screen shots, root cause application, root cause category)

The SI Contractor must document the defect status lifecycle workflow. Below is an example.



Describe the roles required to resolve and manage defects. When test problems are identified, the problem is reported and tracked for resolution and completion. The tester is ultimately responsible for tracking and coordinating resolution with the appropriate project individual at the granular level. The SI defect manager will maintain responsibilities of ensuring defects as a whole are addressed and progressing at an acceptable rate.

8 Test Plan Details For Release

Populate the following table for all the in scope Functional, Non-Functional, and O&M test types for this release.

[Test Type]	
Features / Scope for Release	<ul style="list-style-type: none"> • List reports, interfaces, configurations, enhancements, forms, and workflows • <i>Examples:</i> • <i>All forms/screens within each desktop.</i> • <i>All controls within each screen: tabs, entry fields, trigger controls (e.g. button, toolbars, etc.),</i> • <i>Customized field values within each form/screen.</i> • <i>Role based security</i>
Entry Criteria	<ul style="list-style-type: none"> • List entry criteria • <i>Examples:</i> • <i>Test requirements to be tested have been specified.</i> • <i>Functional and technical designs are signed off as per the IT Methodology.</i> • <i>Test schedule and scripts to be executed have been established.</i> • <i>Resources available and committed</i> • <i>Test environment is setup and smoke-tested</i> • <i>Data is loaded by the development team or test team in the test environment.</i> • <i>All build work is scheduled, clearly defined, and analysis if impact is completed.</i> • <i>Scope of build work being tested is defined in a signed off scope document</i> • <i>Walk through has been held with the test team.</i>
Exit Criteria	<ul style="list-style-type: none"> • List exit criteria

[Test Type]	
	<ul style="list-style-type: none"> • <i>Examples:</i> • <i>Exceptions must be approved by the project management and State Test Manager.</i> • <i>100 percent test script execution, 85 percent test script pass rate.</i> • <i>No outstanding defects prioritized as Critical or High.</i> • <i>Any outstanding defects fully documented as deferred and approved by the State Test Manager.</i>
Test Documentation	<ul style="list-style-type: none"> • Supporting documentation for each test case • Test execution and defect reports (Test Status Summary Report)
Environment	<ul style="list-style-type: none"> • List environment where testing will occur • Detail the equipment to be used, operating systems and application software (including versions) • Include a diagram if possible.
Test Data Requirements	<ul style="list-style-type: none"> • List and describe the test data requirement used for this test type
Testing Tool	<ul style="list-style-type: none"> • Document tool and how the tool will be used for testing
Defect Tracking	<ul style="list-style-type: none"> • Document defect management tool
Responsible Team(s)	<ul style="list-style-type: none"> • List tester team (SI Tester, UAT Tester, ...)

9 Test Tools

Describe the test tools that will be used for testing. These tools include management tools, analytical, automation, performance management tools. SI Contractor must work with the State Test Manager and respective domain SME to review and approve any new testing tools requested.