PHASE 7: TEST PHASE

The Test Phase focuses on an empirical investigation in which the results describe the quality of the system: testing cannot confirm a system functions properly under all conditions but can establish that it fails under specific conditions. A well-known maxim in software development is the earlier a defect is found in the development process the less expensive the fix to the code. Testing early in the system life cycle reduces risks such as schedule delays or cost overruns due to incomplete or unacceptable components. In the Test Phase, testing of the system proves that the system meets all requirements, including those for performance and security. The in-depth security testing of this phase identifies any parts of the system that will not satisfy accreditation criteria. Finally, acceptance testing confirms the developed system satisfies the end users who identified the business need and the requirements. Multiple-release projects require multiple iterations of the Test Phase—one for each release.

1.0 OBJECTIVE/GOALS

Objectives

Successful completion of the Test Phase should comprise:

- Proof through system, security, and user acceptance tests that the system meets all requirements, functions according to design parameters, and satisfies all business, technical, and management stakeholders
- Assurance that the system functions as described in the Operations Manual and User Manual
- Conversion of data from the legacy system
- Approval to progress to the Implementation Phase

Goals

The purpose of the Test Phase is to guarantee that system successfully built and tested in the Development Phase meet all requirements and design parameters. After being tested and accepted, the system moves to the Implementation Phase.

2.0 DELIVERABLES AND APPROVALS

SDLC deliverables help State agencies successfully plan, execute, and control IT projects by providing a framework to ensure that all aspects of the project are properly and consistently defined, planned, and communicated. The SDLC templates provide a clear structure of required content along with boilerplate language agencies may utilize and customize. State agencies may use formats other than the templates, as long as the deliverables include all required content.

The development and distribution of SDLC deliverables:

- Ensure common understanding among Development Team members and stakeholders,
- Serve as a reminder of specified plans as projects become increasingly complex,
- Provide agency senior management and other State officials insight into project risks and ongoing performance,
- Encourage the execution of repeatable and consistent processes,
- Facilitate the implementation of project management and agency IT best practices, and
- Result in a comprehensive record of project performance useful for many purposes (e.g. staff knowledge transfer, budgetary and other assessment activities, lessons learned).

During the development of documentation, the Development Team should:

- Write comprehensive, easy to understand documents with no redundant information.
- Develop an organized document repository for critical project information, so Development Team members can easily access, store, and reference project documents and other deliverables from all life cycle phases.
- Implement routine deliverable reviews to correct inaccuracy, incompleteness, and ambiguities.
- Recognize that sample templates for deliverables are available; agencies might accept deliverables in different formats as long as all required information is present. The content of these deliverables might expand or shrink depending on the size, scope, and complexity of the project.
- Recycle or reference information from earlier documents where possible and beneficial.

The following is a listing of deliverables required of all projects for this phase of work. Deliverables need to be updated for each iteration of the Test Phase.

Deliverables	Goals	Developed By	Approved By
Test Analysis Approval Determination – summarizes the system's perceived readiness and is attached to the Test Analysis Report as a final result of the test reviews.	• Document the perceived production-readiness of the system • Serve as an input to the project Readiness Document described below	Development Team	Project Sponsor Agency CIO
Test Problem Reports – document problems encountered during testing; are also attached to the Test Analysis Report.	• Document detailed results of testing	Development Team	Project Sponsor Agency CIO
Information Technology Systems Certification & Accreditation – includes completion of a Security Risk Assessment, Sensitive System Security Plan, Security Operating Procedures, Security Test and Evaluation, and Certification Statements.	• Assess technical and non-technical safeguards to determine the extent to which the system meets security requirements • Obtain formal declaration by a Designated Approval Authority (DAA) that an information system is approved to operate in a particular security mode using a prescribed set of safeguards at an acceptable level of risk	Development Team	Project Sponsor Agency CIO

Deliverables	Goals	Developed By	Approved By
Defect Log – tracks and summarizes in a tabular format defects or bugs found during testing. Defects may be documented via multiple commercially available bug tracking tools or manually in a spreadsheet.	• Allow team members to track reported bugs, or defects • Clearly communicate summary of defects found • Record facts regarding known bugs, such as times reported, individuals who reported them, defect statuses, and team members responsible for addressing the bugs	Development Team	N/A – The Defect Log does not require approval.
Readiness Document – consolidates summary information regarding the current status of the system and the project and provides decision makers with the information necessary to make a "Go-No Go" decision. It should include a checklist for all work products, User Acceptance Test results, other quality control checks such a peer review, and results of the system walkthroughs.	• Provide information necessary to make the "Go-No Go" decision • Consolidate status information regarding the effective completion of the project and achievement of project objectives and SDLC requirements • Affirm achievement of all deliverable acceptance criteria	Development Team	Agency CIO

All deliverables other than those identified as Updates should be developed in this phase. Deliverables identified as Updates should be revisited and enhanced as necessary as prescribed in this phase.

Deliverables produced during this phase must be reviewed in detail and should follow the approval path as defined in the above table (for each iteration). A signature page or section should accompany each deliverable requiring approval. DoIT will periodically request copies of these documents as part of its oversight responsibilities.

3.0 ROLES AND RESPONSIBILITIES

The following personnel participate in the work activities during this phase:

- Project Sponsor
- Executive Sponsor
- Agency CIO
- Project Manager

- Development Team
- Project Stakeholders

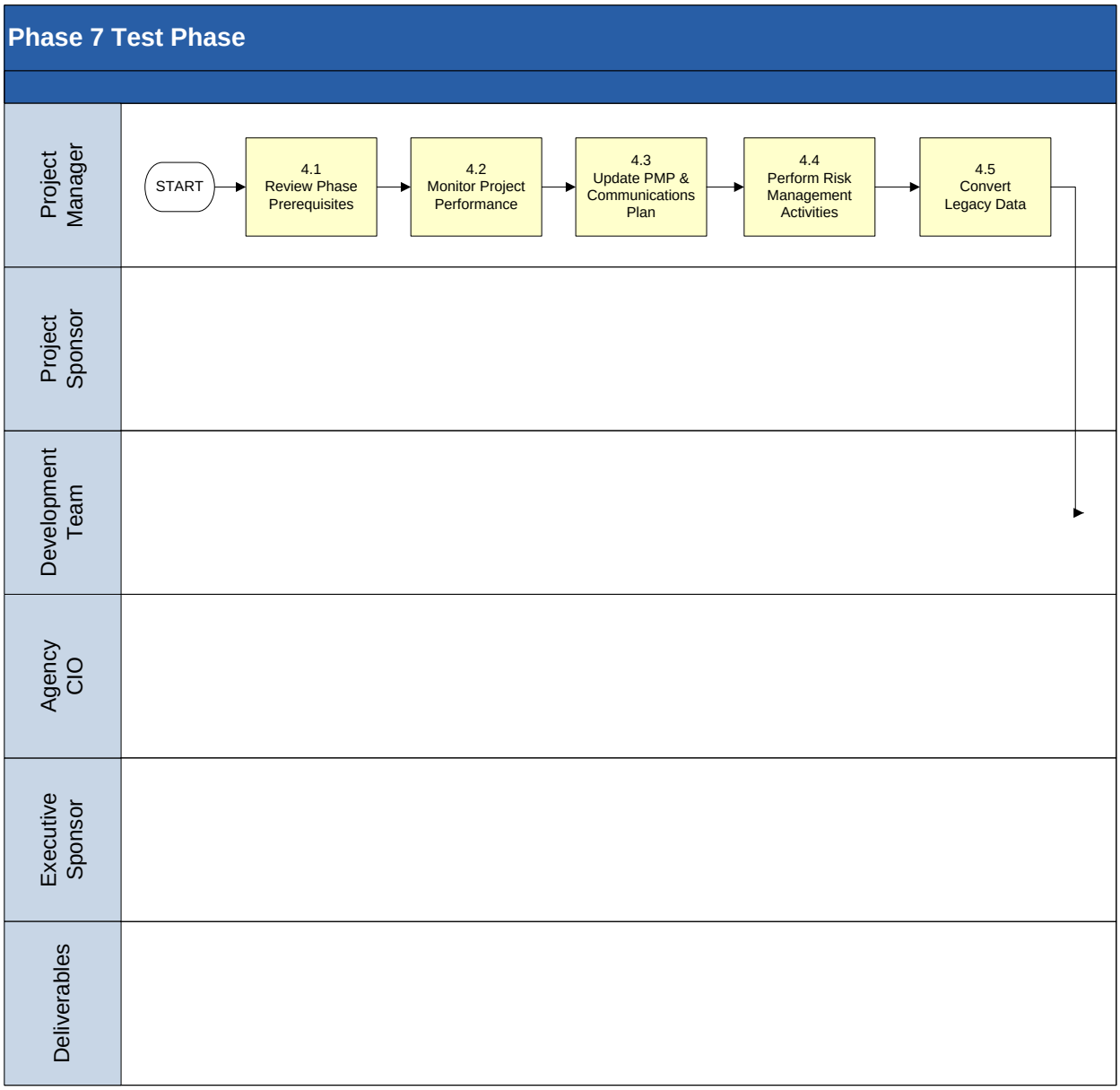
RACI Key
Responsible – Describes role that executes the activities to achieve the task.
Accountable – Describes roles that own the quality of the deliverable and sign off on work that Responsible provides.
Consulted – Describes roles that provide subject matter expertise.
Informed – Describes roles that receive information about the task.

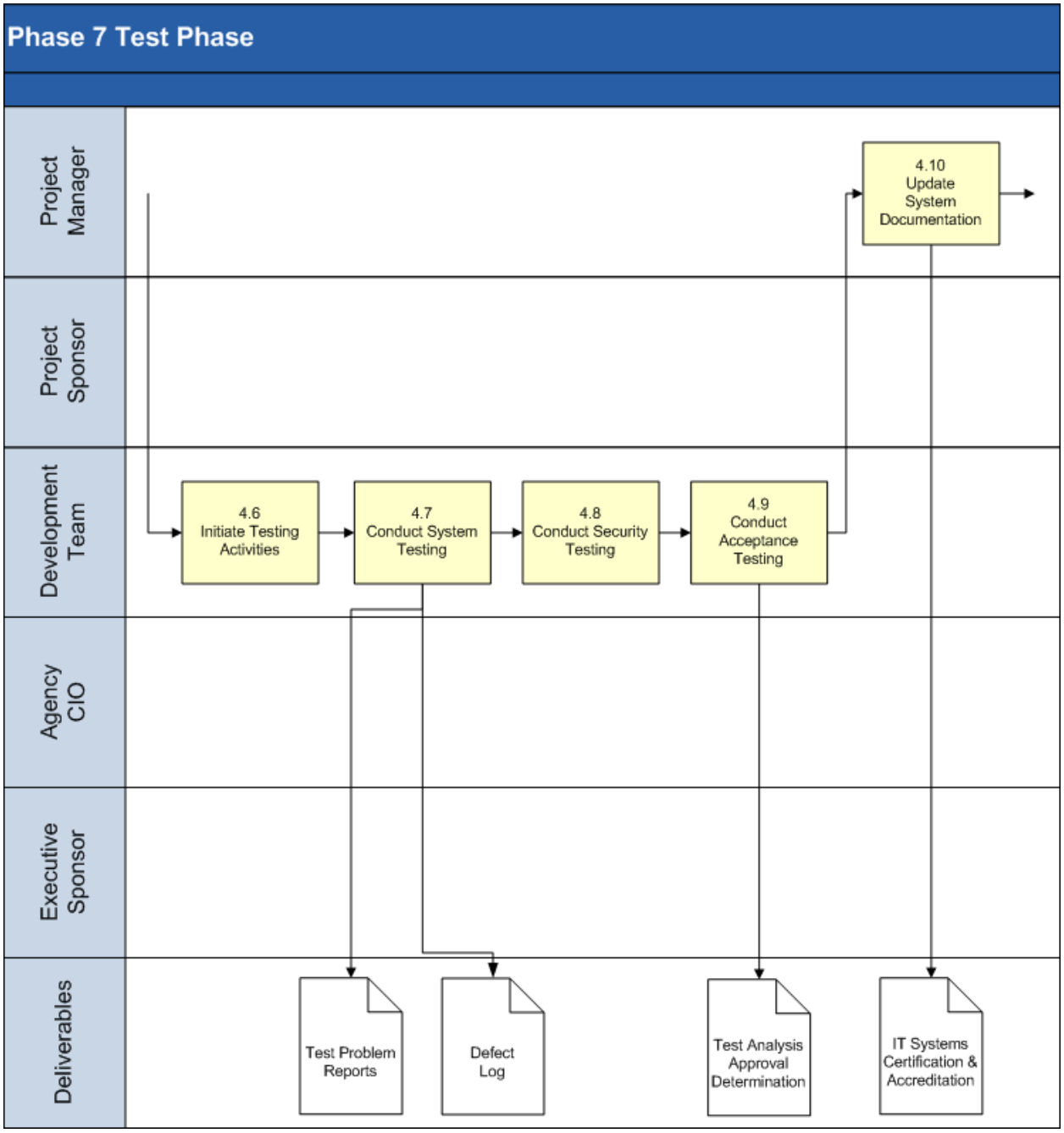
Deliverable	Executive Sponsor	Project Sponsor	Agency CIO	Project Manager	Development Team	Project Stakeholders
Test Analysis Approval Determination	I	I	A	R	I	I
Test Problem Reports	I	I	A	R	I	C
Information Technology Systems Certification & Accreditation	I	I	A	R	I	I
Defect Log	I	I	A	R	I	C
Readiness Document	I	I	A	R	I	C

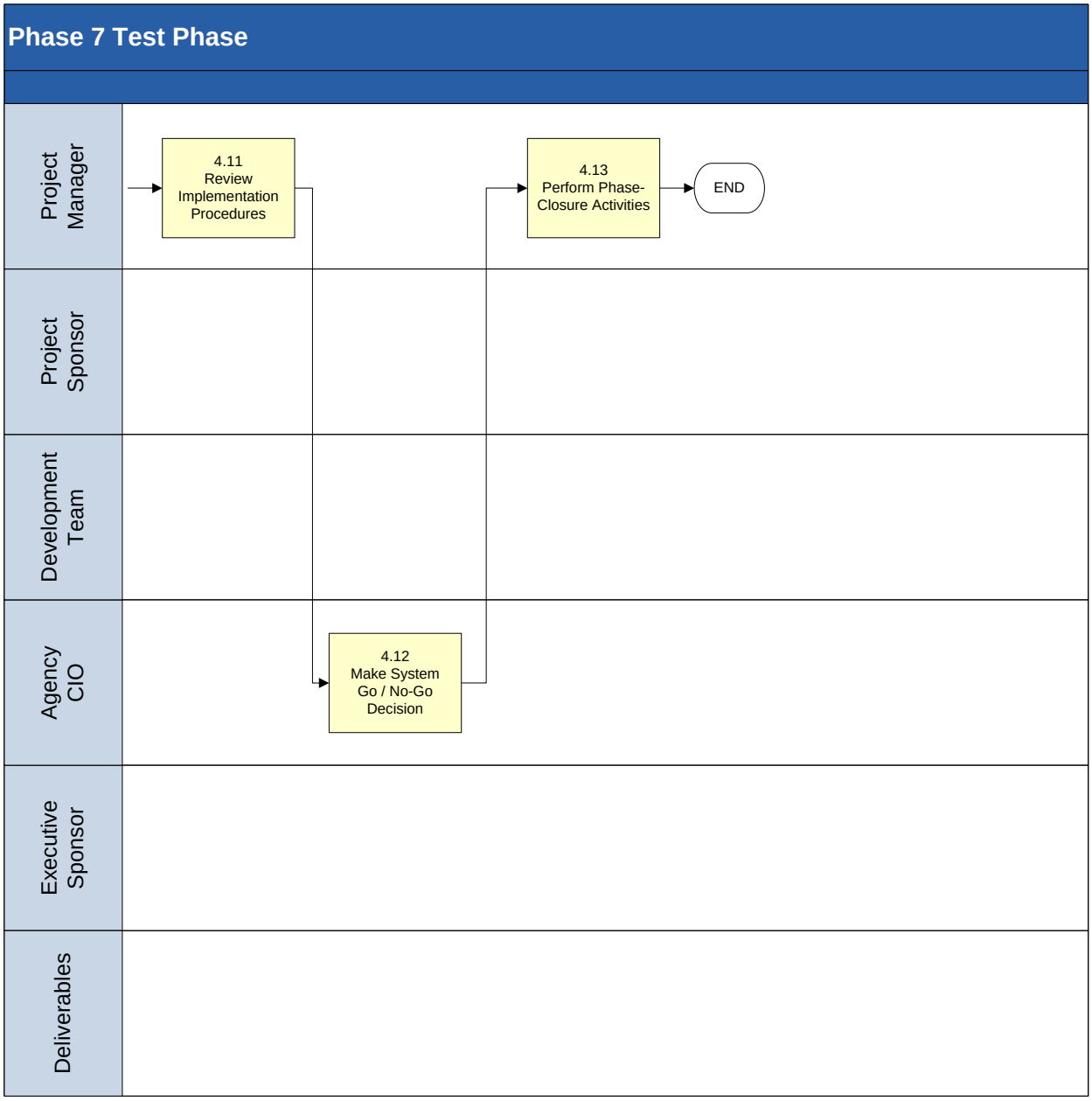
Possible RACI Matrix

The Roles and Responsibilities page has detailed descriptions of these roles and their associated responsibilities.

4.0 TASKS AND ACTIVITIES




Phase 7 Test Phase
Project Manager
Project Sponsor
Development Team
Agency CIO
Executive Sponsor
Deliverables
4.6 Initiate Testing Activities
4.7 Conduct System Testing
4.8 Conduct Security Testing
4.9 Conduct Acceptance Testing
4.10 Update System Documentation
Test Problem Reports
Defect Log
Test Analysis Approval Determination
IT Systems Certification & Accreditation



4.1 **Review Phase Prerequisites.**

The Project Manager ensures the following prerequisites for this phase have been completed:

- The PMP is current, and the schedule showing the target termination date for the system is current. All software and hardware items or units have been constructed and tested.
- Unit and integration test plans and result are final.
- The Conversion Plan for migrating data completely and accurately from the legacy system to the new system is complete.

During the Test Phase, the Development Team frequently may discover problems with interfaces, data structure, and functions that require repair. Ensure that all changes made to

shared processes are also made to the corresponding models and components kept in the Maryland EA Repository.

The Project Manager should confirm and review any testing tools and defect tracking mechanisms and the change management tool used in the software development.

4.2 **Monitor Project Performance.**

The Project Manager monitors project performance by gathering status information about:

- All changes to baseline data
- Change management information
- Activity progress with status details
- List of complete and incomplete deliverables
- Activities initiated and finished
- Testing performed and test results
- Estimated time to completion
- Resource utilization data
- Changes to project scope

The Project Manager also organizes and oversees systematic quality management reviews of project work as a part of monitoring the project performance.

To measure project effort at all phases of the life cycle, the Project Manager establishes timelines and metrics for success at each phase of work when planning project tasks.

The *PMBOK*, fourth edition, provides additional details on controlling project work in sections 4.4 and 4.5 and on project scope control in section 5.5.

4.3 **Update PMP and Communication Management Plan.**

The Project Manager updates the PMP routinely (at least quarterly) to ensure the PMP reflects project performance accurately. Review project performance controls and risks for deviations from the baseline.

Information distribution is one of the most important responsibilities of the Project Manager. The Project Manager reviews and updates the Communication Management Plan at least quarterly to document potential stakeholder changes. The Project Manager redistributes the updated PMP and risk management information according to the revised Communication Management Plan. *PMBOK*, fourth edition, section 10 contains additional details on project communications and information distribution.

4.4 **Perform Risk Management Activities.**

The Project Manager conducts risk assessments during the Test Phase; these activities include:

- Identification – determination of initial and emerging risks that might affect the project as well as each risk characteristic
- Risk Analysis – conducting quantitative and/or qualitative analysis of each identified risk. Usually, qualitative risk management techniques are most applicable for State projects. These

risk analysis methods, as well as the conditions under which each method might be used, are described in detail in section 11 of *PMBOK*.

- Response Planning – planning of methods for developing mitigation, transfer, or avoidance strategies to reduce risk
- Monitoring and Control – tracking risks, monitoring residual risk, identifying new risks, executing response plans, and evaluating risk management effectiveness

These activities occur throughout the project duration to track and mitigate any new or updated project risks. *PMBOK*, fourth edition has details for risk management activities in section 11, particularly in sections 11.2 through 11.6.

4.5 **Convert Legacy Data.**

The Project Manager reviews and executes the Conversion Plan to migrate legacy data to the new system. The Development Team may repeat data migration and conversion for each iteration associated with a release to production. The migration involves entering data into the new system and verifying that the migrated data is correct. Having correct data in the new system is essential to its functioning as intended. A data migration plan is important for transformation, migration, and modernization projects; entirely new systems will likely have no data to convert and have no Conversion Plan.

A few points about data conversion:

- Information can be discarded easily, but adding information requires effort.
- Computers can only add information according to rules; people can add information in any manner.
- Converting data to a new format with additional features does not create the new information but adds space for it. A person must enter the new information.
- Some data conversion can occur directly; other data conversion, however, requires several time-consuming steps.
- Data conversion may result in the loss of information if the target format has fewer features than the source format.

One method of verifying data integrity is parallel operations during which the old system runs simultaneously with the new system. The output from each system is compared; if all is correct, the new system is certified. If the new system fails in any way, continue operations on the old system until all problems are resolved.

4.6 **Initiate Testing Activities.**

The Development Team ensures that all test data is loaded to test databases and prepares any internal or external interfaces. Testing activities need to be performed for each release.

4.7 **Conduct System Testing.**

The Development Team conducts the system tests according to the test plans and documents all results in the Test Analysis Report, Test Problem Reports, and Test Analysis Approval Determination. System testing is conducted on a complete, integrated system to determine compliance with all requirements. System testing includes tests to ensure that the developed

system meets all technical requirements, including performance requirements. The Development Team will repeat system testing for each iteration associated with a release to production.

Return any failed components to the developers for debugging; move the passing components on to security testing.

Testing may be one of two approaches:
- Static testing – conducting reviews, walkthroughs, and inspection
- Dynamic testing – executing the code for a set of test scripts

Testing may follow either a black box testing or a white box testing methodology.
- Black box testing approaches the system with no knowledge of the internal components, structure, or functions. Methods used include boundary value analysis, all-pairs testing, fuzz testing, model-based testing, and specification-based testing. Black box testing provides an unbiased opinion about the code but has the disadvantage of being blind to interconnections and the rest of the system.
- White box testing allows a tester to have knowledge of the internal code and structure of the system. Some methods of white box testing include fault injection methods, mutation testing, and static testing. Combining white box testing with black box testing allows evaluation of the completeness of the test suite and infrequently tested parts of the system and ensures critical functions have been tested.

Regression testing focuses on revealing software errors in functions that did work correctly but stopped working due to modifications. Regression testing typically involves repeating entire test scripts to ensure all functionality operates correctly after a unit or component has been modified.

The Development Team should prepare to perform non-functional tests such as load, usability, and security testing. During load testing, performance tests stress the system and indicate if the system or software can handle large quantities of data or end users. Usability testing checks if the user interface is easy to use and understandable. The Development Team can automate testing to expedite the process and ensure consistency.

Regardless of the testing methodology, the Development Team updates the RTM to reflect all test results and ensure traceability back to the original requirements. When all testing is finished, an audit of the testing should show test results for every element of the system and traceability to its corresponding requirement.

4.8 **Conduct Security Testing.**

The Development Team again reloads the test databases, executes the security/penetration tests, and documents all test results. Return any failed components to the developers for debugging; move the passing components on to acceptance testing after all components have passed subsystem or integration and security testing. The Development Team will repeat security testing for each iteration associated with a release to production.

Test security controls prior to the system deployment to uncover all design and implementation flaws that might violate DoIT's security policy. Security testing involves numerous methods,

such as analyzing system design documentation, inspecting test documentation, and independently executing functional and penetration testing.

State policy for IT systems requires that all Executive Branch agencies certify and accredit IT systems and sites under their ownership and control. The Development Team should review the *DoIT Information Technology Security Certification and Accreditation Guidelines* and the project's SSCD for any actions necessary to enable the system to become certified and accredited prior to implementation. These documents are available at the DoIT State Information Technology Security Policy and Standards webpage.

4.9 **Conduct Acceptance Testing.**

The Development Team reloads the test databases to start and document the acceptance testing. Users participate in acceptance testing to confirm that the developed system meets all user requirements as stated in the FRD. Acceptance testing shall be done in a simulated user environment; the users use simulated or real target platforms and infrastructures. Review, rework, and retest any failed components. When all components pass acceptance testing, the system is ready for implementation. The Development Team will repeat acceptance testing for each iteration associated with a release to production.

4.10 **Update System Documentation.**

During the Test Phase, problems with interfaces, data structures, and functionality are frequently discovered and require fixes. The Project Manager ensures that the documentation reflects any changes from all previous phases as well as changes that occurred during this Phase. This documentation includes the Conversion Plan, the Operations or Systems Administration Manual, the Maintenance Manual, the Training Plan, and the User Manual. The Project Manager coordinates these updates.

4.11 **Review Implementation Procedures.**

Bearing in mind any modifications that result from testing, the Project Manager and Development Team review implementation procedures, including any necessary resources and information, for deploying the system in its target environment.

4.12 **Make System "Go-No Go" Decision.**

The Agency CIO and Project Sponsor decide whether to perform additional testing or to proceed to the next phase, Implementation. They review the system requirements, the user acceptance criteria, and the user acceptance test results and consult with others about the condition of the project and the state of completeness of the system. Using the requirements and acceptance criteria as a quantitative base, they consider other qualitative factors through a collaborative discussion to arrive at the "Go-No Go" decision. This critical project decision can move the system to a production state.

4.13 **Perform Phase-Closure Activities.**

The Project Manager and the Development Team prepare and present a project status review for the Agency CIO, Project Sponsor, Executive Sponsor, and other project stakeholders after completing all Test Phase tasks. This review addresses:

- Status of Test Phase activities
- Planning status for all subsequent life cycle phases, with significant detail about the next phase
- Status of resource availability
- Project scope control as described in the Project Scope Statement/System Boundary Document (SBD) and any required adjustments to the scope
- Changes to the project schedule and estimated completion date
- "Go-No Go" decision made to proceed to next phase, based on Test Phase information
- Verification that all changes are conducted in accordance with the approved Change Management Plan

The Project Manager compares actual project performance to the PMP and the projected cost of the project to determine any variances from the cost baseline during the phase-end review. The Project Manager also performs a comprehensive risk assessment of the project to update the Risk Register. The Project Manager updates the Maryland EA Repository with any new or updated components before beginning the next phase, Implementation.

The Project Manager must obtain deliverable approval signatures before proceeding to the Implementation Phase.

Update the project documentation repository upon completion of the phase-closure activities.

5.0 CONCLUSION

At the end of the Test Phase, the Development Team has completed a working, fully tested information system that meets all business and technical requirements. The approval of the Test Phase deliverables, the completion of the Test Phase project status review, and the approval to proceed to the next phase, signify the end of the Test Phase.