

## **PHASE 6: DEVELOPMENT PHASE**

The Development Phase features a key step in the project: system construction. The previous phases lay the foundation for system development; the following phases ensure that the product functions as required. To complete the Development Phase successfully, two elements are required: 1) a complete set of design specifications and 2) proper processes, standards, and tools. The focus of the Development Phase is on the preparation of a production-quality release of the selected COTS solution.

### **1.0 OBJECTIVE/GOALS**

#### **Objectives**

Successful completion of the Development Phase should comprise:

- Installing and building the system
- Testing and integrating the units into larger components
- Preparing the technical environment for the system
- Approval to progress to the Test Phase

#### **Goals**

The purpose of the Development Phase is to convert the system design prototyped in the Design Phase into a working information system that addresses all documented system requirements. At the end of this phase, the working system will enter the Test 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.

Deliverable	Goals	Developed By	Approved By
<b>Software Development Document</b> – details the provisions, including specific coding instructions, unit test cases, and scripts, for developing and/or modifying each unit or module of the system; also includes development procedures for issue tracking and configuration management and any other information that aids in building the functionality of the system.	<ul style="list-style-type: none"> <li>• Document all preparations related to the development and/or modification of each software unit and other items that will help explain the functionality of the software</li> <li>• Describe development methodologies</li> </ul>	Development Team	Agency CIO
<b>System (Application) Software</b> – file or set of files that contain the application or software code on discs or other media.	<ul style="list-style-type: none"> <li>• Provide software that meets the business needs and all requirements</li> </ul>	Development Team	Agency CIO
<b>Test Data</b> – contains dummy data used for testing; the data is not sensitive or live.	<ul style="list-style-type: none"> <li>• Provide sample representative data to use in test activities</li> <li>• Ensure that testing results simulate production results</li> </ul>	Development Team	Project Manager
<b>Integration Document</b> – describes the assembly and interaction of the hardware, software, and any other components of the system. This document is not applicable for projects that only implement SaaS.	<ul style="list-style-type: none"> <li>• Document planned approach and activities for the integration of hardware, software, and other system components</li> </ul>	Development Team	Agency CIO

Deliverable	Goals	Developed By	Approved By
<b>Test Analysis Report(s)</b> – presents a description of the unit tests and the results mapped to the system requirements; also identifies system capabilities and deficiencies.	<ul style="list-style-type: none"> <li>Record results of tests</li> <li>Present the capabilities and deficiencies for review</li> <li>Provide a means for assessing software progression to the next stage of development or testing</li> </ul>	Development Team	Project Sponsor Agency CIO
<b>Conversion Plan (Update)</b> – describes the strategies and approaches for converting/migrating data from an existing system to another hardware or software environment.	<ul style="list-style-type: none"> <li>Document all planned activities to ensure a smooth data conversion from a legacy system to a new environment</li> </ul>	Development Team	Project Sponsor Agency CIO
<b>Implementation Plan (Update)</b> – describes how the information system will be deployed as an operational system.	<ul style="list-style-type: none"> <li>Define all planned activities to ensure successful implementation to production operations</li> </ul>	Development Team	Project Sponsor Agency CIO
<b>Operations Manual or Systems Administration Manual</b> – The Operations Manual focuses on mainframe systems; the Systems Administration Manual is oriented toward distributed (client/server) applications. These manuals are not applicable for SaaS-only projects.	<ul style="list-style-type: none"> <li>Provide detailed instruction for system operations</li> </ul>	Development Team	Agency CIO
<b>Release Notes</b> – provides summary information regarding the current release of the system being built; typically includes major new features and changes and identifies known problems and workarounds.	<ul style="list-style-type: none"> <li>Document critical information regarding the system release</li> </ul>	Development Team	Agency CIO

Deliverable	Goals	Developed By	Approved By
<b>Maintenance Manual</b> – details effective system maintenance. Appendices might document maintenance procedures, standards, or other essential information. This manual is not applicable for SaaS-only projects.	<ul style="list-style-type: none"> <li>• Provide maintenance personnel with the information necessary to maintain the system effectively</li> </ul>	Development Team	Agency CIO
<b>Training Plan</b> – outlines technical and user training needs on the new or enhanced information system.	<ul style="list-style-type: none"> <li>• Ensure the schedule accounts for all necessary training needs to implement, operate, and maintain the system successfully</li> </ul>	Development Team	Project Sponsor Agency CIO
<b>User Manual</b> – describes to end users how to make full use of the information system, including system functions and capabilities, contingencies and alternate modes of operation, and step-by-step procedures for system access and use.	<ul style="list-style-type: none"> <li>• Provide users with detailed information to fully utilize the system</li> </ul>	Development Team	Project Sponsor 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. 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

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

- Agency CIO
- Project Sponsor
- Executive Sponsor
- 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.

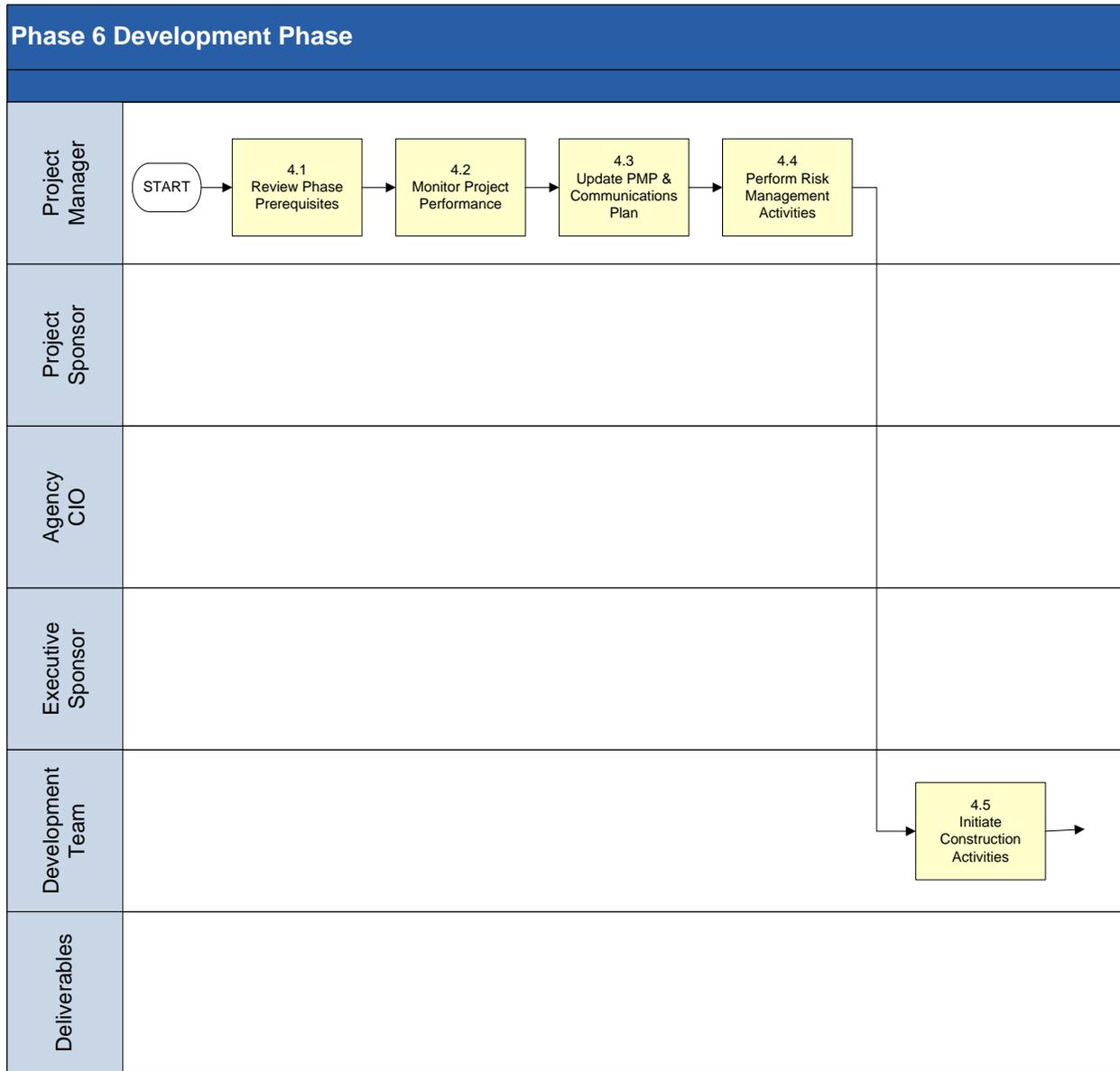
**Informed** – Describes roles that receive information about the task.

Deliverable	Executive Sponsor	Project Sponsor	Agency CIO	Project Manager	Development Team	Project Stakeholders
Software Development Document	I	I	A	R	I	I
System (Application) Software	I	A	A	R	I	C
Test Data	I	I	A	R	I	I
Test Analysis Report(s)	I	I	A	R	I	I
Conversion Plan	I	I	A	R	I	I
Implementation Plan	I	A	A	R	I	I
Operations Manual or Systems Administration Manual	I	I	A	R	I	I
Release Notes	I	I	A	R	I	I
Maintenance Manual	I	I	A	R	I	I
Training Plan	I	A	A	R	I	I
User Manual	I	I	A	R	I	I

Possible RACI Matrix

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

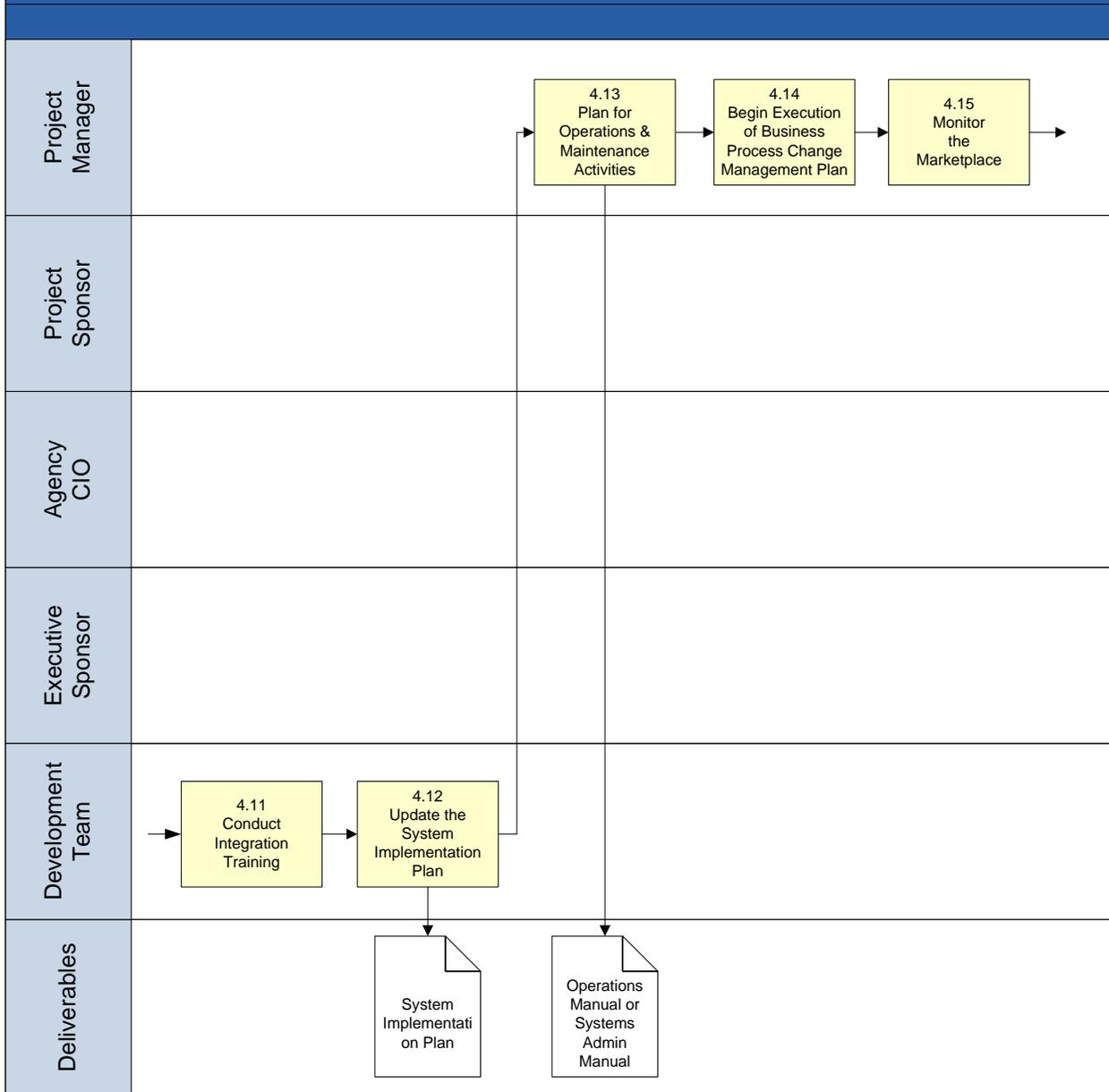
## 4.0 TASKS AND ACTIVITIES



## Phase 6 Development Phase

Project Manager	
Project Sponsor	
Agency CIO	
Executive Sponsor	
Development Team	<pre> graph LR     A[4.6 Install and Test Base Software] --&gt; B[4.7 Customize and Test Software]     B --&gt; C[4.8 Prepare Test Data]     C --&gt; D[4.9 Develop and Conduct Preliminary Data Migration Testing]     D --&gt; E[4.10 Integrate Software]     E --&gt; F[ ]             </pre>
Deliverables	

## Phase 6 Development Phase





- The Development Team has completed:
  - System Design Document
  - Test plans for unit and integration testing (draft)
  - Conversion Plan (draft)
  - Implementation Plan (draft)
  - Integration Document (draft)
  - Operations or System Administration Manual (draft)
  - Maintenance Manual (draft)
  - Training Plan (draft)
  - User Manual (draft)

#### 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
- 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* 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 account for potential changes in project stakeholders. The Project Manager distributes the updated PMP and risk management information according to the revised Communication Management Plan. *PMBOK*, Chapter 10 contains additional details on project communications and information distribution.

#### 4.4 Perform Risk Management Activities.

The Project Manager conducts risk management activities during the Development Phase; these activities include:

- Identification – determination of initial risks that might affect the project and emerging risks as well as each risk characteristic
- Risk Analysis – conducting quantitative and/or qualitative analysis of each identified risk. Usually, qualitative risk management techniques are the 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 *PMBOK*, section 11.
- Response Planning – planning of methods for developing mitigation, transfer, or avoidance strategies to reduce risk
- Monitoring and Control – definition of procedures to track risks, monitor residual risk, identify new risks, execute response plans, and evaluate risk management effectiveness

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

#### 4.5 Initiate Construction Activities.

The Development Team begins construction by:

- Verifying selection for standards, methods, tools, and computer programming languages
- Consulting with the Maryland EA Repository for available tools and components of the system (programming language identification is applicable for COTS implementation projects that require any amount of customization)
- Developing the Software Development Document
- Carrying out Development Phase activities according to the detailed project WBS prepared during the Planning Phase
- Performing configuration management and change control
- Documenting and resolving problems and non-conformances found in the software products and tasks with an issue-tracking process

Project stakeholders should be involved periodically throughout the Development Phase to ensure that the development team understands their expectations and that it develops the system in accordance with requirements.

#### 4.6 Install and Test Base Software.

The Development Team installs the base software and configures components to meet requirements. This frequently involves initializing and loading databases, registering components with the operating system, setting parameters, initializing security features, scripting screens and reports, developing other scripts, and verifying and testing the results of configuration. Requirements and design decisions should drive specific activities conducted as part of the installation and testing of the base software. This activity involves COTS installation and configuration, sometimes referred to as “tailoring”—not customization, modification, adaptations, or extensions that require access to source code.

#### 4.7 Customize and Test Software.

To meet business needs, many COTS implementation projects require some level of customization or modifications to the base software beyond mere configuration. COTS customization is inherently risky because COTS software modifications frequently negate much of the productivity increase gained from using COTS components and may jeopardize supplier maintenance, support, and upgrades. As such, decisions to customize COTS solutions must be made with care and after significant efforts to obtain stakeholder consensus to meet business needs through available COTS functionality. SaaS projects do not involve customization.

For projects that do require COTS customization, the team modifies COTS components and develops the custom components necessary to satisfy all requirements and incorporates custom components with pre-existing components. Care should be taken to ensure that modifications are modular in nature to help mitigate some of the inherent risk in customization. This activity is performed after the base software is installed, configured, and verified.

Customization may also require the development of “glue code” to make the COTS software components function as required. Glue code serves as a layer between the system and the COTS components, wrapping the data in such a way to ensure that upgrades are as seamless as possible. Again, the amount of custom code should be kept to a minimum, because the greater the amount of custom code, the greater the complexity and difficulty in addressing bug fixes.

While developing the custom code, the Development Team should consult the Maryland EA Repository for any usable components and strive to build reusable code. Reusing code is particularly valuable when the project involves new software or software components or replacing large legacy components. After building the software, the Development Team should enable future software reuse on State projects by submitting the code or software to the Maryland EA Repository.

The Development Team creates test procedures and data for testing each software unit and database.

The Development Team performs unit tests of each software unit and documents the results in the Test Analysis Report(s). The Development Team also updates the Requirements Traceability Matrix (RTM) to include pointers to test results and updates the test requirements.

The Development Team conducts routine code reviews to monitor progress and quality according to:

- Traceability to the requirements and design of the system
- Quality of coding documentation
- Test coverage of units
- Resolution of testing deficiencies
- Appropriateness of coding methods and standards used

#### 4.8 **Prepare Test Data.**

Test data should be prepared with real and varied data in realistic amounts to ensure that testing results mirror production results as much as possible. Including real data in the test environment allows stakeholders, during acceptance testing, to relate more to the system in a meaningful way and improves effectiveness of testing business rules and algorithms.

#### 4.9 **Develop and Conduct Preliminary Data Migration Testing.**

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.

The Development Team may need to develop migration utilities that will extract, transform, and load data from a legacy system to the new system. The migration involves manually entering or systematically loading the data into the new system and verifying that the migrated data is correct. Before any data is migrated, the Development Team should develop, test and validate initial portions of the Conversion Plan to determine if any plan modifications are necessary. The scope of the test will vary depending on the scope of the project. The Development Team should develop the data migration programs and tools according to the Conversion Plan and may need to repeat data migration testing for the release to production.

#### 4.10 **Integrate Software.**

The Development Team integrates the software units and software components and tests as the aggregates are developed. The Development Team must ensure that each aggregate satisfies the requirements of the software item and that the software item is integrated at the conclusion of the integration activity.

#### 4.11 **Conduct Integration Testing.**

The Development Team conducts integration testing of integrated software units. Integration testing involves the testing of aggregated software units, which were previously unit-tested. Through integration testing, Development Teams are able to verify requirements of major groupings of the software.

The Development Team summarizes the testing results in the Test Analysis Report(s). The Development Team also updates the RTM to include pointers to test results.

#### **4.12 Update the System Implementation Plan.**

The Development Team updates the implementation procedures as necessary for the system in the target environment.

#### **4.13 Plan for Operations & Maintenance Activities.**

The Project Manager and Development Team review Development Phase deliverables, and the Project Manager adds any necessary tasks to the project plan for the Operations & Maintenance Phase.

#### **4.14 Begin Execution of Business Process Change Management Plan.**

In the Development Phase, the Project Manager ensures that all planned business process change activities are conducted in accordance with the Business Process Change Management Plan approved in the Requirements Analysis Phase. End users begin to implement necessary business process changes and monitor and communicate the implications of these changes. This may involve development of required policies and procedures, restructuring the organization, and implementing mechanisms to encourage system adoption.

#### **4.15 Monitor the Marketplace.**

The Project Manager and Development Team must continue to maintain current market information to identify and evaluate new and changed COTS components and to determine or update their upgrade approach appropriately.

#### **4.16 Complete the System Documentation.**

The Development Team finalizes the following documents:

- Conversion Plan
- Implementation Plan
- Maintenance Manual (not applicable for SaaS)
- Operations or Systems Administration Manual (not applicable for SaaS)
- Training Plan
- Release Notes
- User Manual

#### **4.17 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 completion of all Development Phase tasks. This review addresses:

- Status of Development 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 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 Development 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 detect and understand 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 analysis. The Project Manager updates the Maryland EA Repository with any new or updated components before beginning the next phase, Test.

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

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

## **5.0 CONCLUSIONS**

At the end of the Development Phase, the Development Team has created a working information system. The completion of the System (Application) Software, approval of the Development Phase deliverables, the completion of the Development project status review, and the approval to proceed to the next phase, signify the end of the Development Phase.