

Information Technology Policy

Systems Development Life Cycle Policy

<i>ITP Number</i> ITP-APP012	<i>Effective Date</i> May 1, 2013
<i>Category</i> Recommended Policy	<i>Supersedes</i>
<i>Contact</i> RA-itcentral@pa.gov	<i>Scheduled Review</i> May, 2014

This Information Technology Policy (ITP) establishes policy for Application Development Methodologies

1. Purpose

Managing information systems to support business and technical operations requires an organization to have a well-defined Systems Development Life Cycle (SDLC) framework. This ITP establishes policy for a SDLC framework and related software application development methodologies that are essential components in the development and delivery of software applications to support agency business needs.

2. Scope

This Information Technology Policy (ITP) applies to all departments, boards, commissions and councils under the Governor's jurisdiction. Agencies not under the Governor's jurisdiction are strongly encouraged to follow this ITP.

3. Background

Software application development is a complex endeavor, susceptible to failure, unless undertaken with a deliberate and systematic methodology. Application development requires an SDLC framework that fully integrates Software Application Development Methodologies (SADM), Project Management, and Software Quality Control and Assurance components to create quality software applications with real business value in a timely cost effective manner.

In addition, a SDLC is the essential underlying foundation required in establishing a standard framework for the proper evaluation, development, installation, validation, integration, implementation, and life cycle management of information system solutions (i.e., hardware and software), regardless of the

systems engineering or software development methodologies used to execute the development and/or delivery the information systems solution.

4. Definitions

Systems Development Life Cycle: The SDLC is a conceptual model used in software and systems engineering as well as project management that describes the phases involved in an information system solution development and delivery. A SDLC framework consists of multiple phases to assure high quality systems are delivered, provide strong management controls over IT projects, and ensure that the information system can, and will, work as required and is effectively maintained to support agency's missions.

Software Application Development Methodology (SADM): A software application development methodology is a structured framework of procedures and processes used to develop custom software applications. Software application development methodologies are essentially derivatives from the system development life cycle model but are unique in their respective processes and execution. Some common SADMs are as follows:

Waterfall Model: A software development process model that involves distinct sequential phases (i.e., conception, requirements, design, build/construct, test, and implementation). Solution progress is seen as flowing steadily downwards (like a waterfall) through each of the phases. This means that any phase in the development process may begin only if the previous phase is complete. There can be some slight variations in the waterfall approach (i.e., modified water fall) that define the circumstances and processes to go back to the previous phase. Documentation in this process is also sequential. Documentation is typically created, delivered, and approved with each phase as a prerequisite for the next phase to begin. Each phase in this model is a phase gate or key milestone.

Spiral Model: An incremental software development process model that incorporates requirements, design, build/construct, test/simulations, and deploy prototype phases separated by planning and risk assessment. A prototype is created with each iteration and evaluated until a final production ready (i.e., fully functional and validated) prototype model has been created. This method can be used to create temporary prototype solutions that are later discarded or for large, expensive, and complicated projects using each iterative prototype build as a phase gate and/or milestone. Documentation in this process is dynamic and incrementally refined. Documentation is finalized with the implementation of the final production ready prototype.

Agile Model: A highly iterative software application development model that involves an interactive, cross-functional, and focused team approach to build software solutions in a time boxed (sprints) development methodology. The

Agile model uses feedback and checklists, tightly integrated cross functional teams, and multi-faceted iterations or sprints to quickly build custom software applications. The feedback is driven by regular tests and releases of the evolving software.

5. Objectives

This ITP is being established to:

- Provide a framework for the creation and delivery of high quality business information systems that:
 - Meet or exceed customer expectations when promised and within cost estimates;
 - Work effectively and efficiently within the current and planned information infrastructure; and
 - Are properly managed, maintained, and properly documented throughout their useful life.
- Facilitate the development of agency specific policies and associated standard operating procedures to establish sound SDLC frameworks, audit controls, and separation of duties.
- Ensure Commonwealth agencies are employing the best practices of SDLC and providing some assurance that systems are being developed efficiently and effectively.

6. Policy

All new application development and enhancement projects are required to utilize a well-documented systems development life cycle framework. This applies to projects performed by Commonwealth employees and by Commonwealth contractors.

Whether a software application development methodology is based on waterfall, spiral, agile processes or some other methodology they share fundamental systems development life cycle components and activities. Agencies are required to establish a SDLC framework that at a minimum include the following components:

Feasibility - processes and procedures to evaluate and define the best solution approach through research, feasibility studies, analysis of business needs and/or high level requirements, resources, capability, capacity, IT investment and risk strategies, alternatives analysis, SADM, etc.

Requirements Management - requirements definition, analysis, refinement, categorization, prioritization, changes, traceability, and documentation procedures and processes based on SADM.

Design – processes and procedures for the creation and evaluation of conceptual design models and high level diagrams to detailed design models and diagrams based on SADM.

Build – processes and procedures utilized to construct and/or configure the solution based on SADM.

Testing & Validation - processes and procedures associated with test planning, test design, test execution, validations, defect management, and approvals, based on SADM and in relation to unit, systems integration, user acceptance, and security vulnerability testing requirements. These processes and procedures should also include integrated quality control and assurance mechanisms to ensure solution meets all business, systems, security, policy, product quality, and/or other relevant compliance/certification requirements.

Implementation - processes and procedures regarding production ready solution adoption, delivery, and deployment; including business and technical operational readiness assessments with integrated go-live decision and roll-back mechanisms.

Operations & Maintenance - processes and procedures to ensure the system is monitored for expected performance in accordance with requirements in live production environments, needed modifications are incorporated and subsequent product releases are effectively managed to ensure the system continues to evolve in order to meet the changing needs of the business. All documentation is finalized and archived for future reference.

Agencies shall incorporate separation of duties to maintain continuity and integrity throughout the execution of the procedures and processes associated with the SDLC framework and affiliated software development projects. Careful consideration should be given to:

Establishing access controls granting permissions to Commonwealth employees and/or outside contractors performing multiple roles within the various environments (i.e., development, production, system integration, testing, staging, etc) to add, modify, delete, and migrate application code, data sets, and/or make configuration changes to systems in these environments.

Granting privileged access granting permissions to outside contractors to add, modify, and/or delete user accounts and IDs and/or information systems security configurations.

Establishing controls defining oversight, authority and responsibilities for end product verifications, validations, and final acceptance/approvals associated with operational readiness assessments, testing, systems and data conversions, and go-live decisions.

Agencies shall ensure proper alignment of SDLC frameworks with the desired project management approach based on the SADM chosen, i.e., integrated

project management elements associated with waterfall, spiral or agile approaches that are used to facilitate the initiating, planning, executing, monitoring/controlling, and closing of all systems development tasks and activities within the SDLC framework.

It is acceptable for agencies to maintain and utilize more than one SADM and project management approach within the SDLC framework.

7. Responsibilities

Agencies: Agencies that perform application development are required to utilize a documented SDLC framework for all new application development and enhancement projects.

8. Related ITPs/Other References

N/A

9. Authority

- Executive Order 2011-05, *Enterprise Information Technology Governance*

10. Publication Version Control

It is the user's responsibility to ensure they have the latest version of this publication. Questions regarding this publication are to be directed to RA-itcentral@pa.gov.

Version	Date	Purpose of Revision
Original	9/18/2008	Base Document
Revision	7/13/2012	Meet GAAP Findings
Revision	5/1/2013	Meet GAAP findings, required to establish SDLC; rescind STD-APP012A
Revision	12/10/2013	ITP Reformat
Revision	4/14/2014	Changed contact email, minor formatting