pennsylvania
OFFICE OF ADMINISTRATION
INFORMATION TECHNOLOGY

Information Technology Policy

Software Development Life Cycle (SDLC) Policy

ITP Number ITP-SFT000	**Effective Date** February 17, 2017
Category Software	**Supersedes** None
Contact RA-ITCentral@pa.gov	**Scheduled Review** August 2019

1. Purpose

Establishes policy for a Software Development Life Cycle (SDLC) framework, and related software application development methodologies and tools that are essential components in the management, development, and delivery of software applications to support agency business needs and services.

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.

An 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, and/or tools used to automate, manage, execute the development and/or delivery the information systems solutions.

It is imperative to have an SDLC framework established with procedures and processes aligned with their respective software application development methodology. Integrating software development tools (e.g., CAD, Application Life Cycle Management, Modeling, Testing, Compliance) can aid in the management, automation, and consistency of solution development as well as the overall quality of the product. These tools must also be properly aligned and integrated into the SDLC framework and respective SADM approach.

Managing the application portfolio is a key component of life cycle management. Understanding the type, composition, status, and risks associated with agency applications that enable business and IT services is critical for IT strategic planning and making informed decisions regarding modernization, enhancements, divestiture, or replacement based on the changing needs of the business and IT ecosystems.

4. Objective

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.
- Ensure proper alignment with Business and IT Service Portfolio and integrated ITIL processes
- 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.
- Outline some tools and specifications that can be used/referenced by agency application development teams for facilitating the management, automation, consistency, quality assurance, and compliance of solutions.
- Provide SDLC strategy concepts
- Posture the Commonwealth application portfolio towards a COTS or SaaS-first priority

5. 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 (SADM) 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 an 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.

Cloud Services Request

Refer to ITP-BUS011 *Commonwealth Cloud Services Requirements* for guidance on cloud solution implementation into the enterprise.

Agencies that have determined a Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), or Infrastructure-as-a-Service (IaaS) cloud-based solution meets the business requirements are required to engage OA/OIT Enterprise through a Service Request process prior to consumption of the cloud-based solution. This process allows the agency and OA/OIT Enterprise to perform a robust vetting analysis that will:

- Determine the impact and capacity of bandwidth on the Commonwealth backbone
- Ensure and maintain agency and enterprise information security
- Help establish consistent rules of engagement for implementation of the solution

- Help establish flexible cloud procurement vehicles
- Allow for a centralized repository of lessons learned, use cases, and other cloud-based artifacts to enhance the Commonwealth's cloud solutions posture
- Determine the impacts to existing to existing agency and/or enterprise service offerings, capabilities, and resources

Additional details on the Service Request process is in Section 8 - Related ITPs/Other References.

Requirements Management - requirements definition, analysis, refinement, categorization, prioritization, changes, traceability, and documentation procedures and processes based on SADM. Service Design Coordinator shall ensure alignment with Service Design Package (SDP) and affiliated application, infrastructure, data/information, security requirements defined and managed through service design and integrated SDLC frameworks.

Principles – To reduce the commonwealth's legacy and customized application portfolio, agencies tasked with new or modernizing applications to support business needs are to emphasize reuse engineering of existing solutions, Commercial-off-the-Shelf (COTS) and Software-as-a-Service (SaaS) solutions over commonwealth-customized applications. Agencies are to also consider leveraging multiple COTS or SaaS solutions that can be integrated to formulate a holistic solution to the business needs. Evidence of such must be included with required project initiative documentation.

If no third-party solution (i.e. COTS, SaaS, or combination with integration), meets business requirements, next consideration is to be given to commonwealth-custom application actively maintained in the Commonwealth (utilize the Enterprise Application Inventory for analysis of available commonwealth-custom applications). If a commonwealth-custom application is not available or does not meet business requirements, agencies may then leverage internal and external personnel to develop a *commonwealth-custom application*. NOTE: This policy requires agencies to enter and maintain all custom applications into the Enterprise Application Inventory. Failure to maintain current continuity plans and an updated application entry in the Enterprise Application Inventory may result in delays in agency project approvals.

Agencies must perform a comprehensive multidimensional examination of COTS and/or SaaS solution alternatives in comparison to custom application development. A comparative analysis matrix should be created using predefined evaluation criteria with weighted scoring and ranking method to evaluate solution alternatives in making informed decisions as to the solution that will provide the best value to the organization.

Agencies must be able to provide sound justification for the why a COTS or SaaS solution alternative is or is not the viable alternative to custom application development when investing in a new, modernizing, or replacing application platform used to support the agency mission.

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. Service Design Coordinator shall ensure alignment with Service Design Package (SDP) and

affiliated application, infrastructure, data/information, security design specifications managed through service design, change management and integrated SDLC frameworks.

Build – processes and procedures utilized to construct and/or configure the solution based on SADM. All Commonwealth-custom application source code and/or software must reside on Commonwealth IT Resources or approved commonwealth-contracted resources. Builds and associated packages, configurations, databases, and accounts are to be designated as development versions with naming conventions identifying as such. This source code and/or software is not being shared in public domains. A COPPAR waiver is required if an agency needs to share Commonwealth-custom application source code and/or software in a public domain. Service Design Coordinator shall ensure alignment with Service Design Package (SDP) and service transition activities affiliated with application, infrastructure, data/information, security design specifications managed through service design, transition, change management and integrated SDLC frameworks.

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.

- Application quality is fundamental to delivering expected business outcomes and agreed upon service level. The quality of testing is the overall contributor to the quality of the application. The effectiveness of the testing effort can be maximized by selection of a testing strategy which includes thorough unit, integration, system, regression, performance, stress testing, good management of the testing process, and the appropriate use of tools. Code packages, configurations, databases, and accounts are to be designated as beta/staging/test versions with naming conventions identifying as such.
- Testing tools are to be used to verify that changes in functionality were successfully implemented and that changes were implemented without degradation to other application components or performance. The use of testing tools is to be integrated with the change management strategy and the standards defined in section 7.

The selection and use of test tools (open source or purchased) should be properly evaluated relative to interoperability, extensibility, maintainability, and overall test coverage and effectiveness under the specified test conditions/parameters and targeted systems environment(s).

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. Builds and associated packages, configurations, databases, and accounts are to be designated as production versions with naming conventions identifying as such.

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 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 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.

Agencies shall ensure proper alignment and integration of application lifecycle management (ALM) and other application development tools with established SDLC frameworks and corresponding SADM approach used in the solution development. When utilizing tools, agencies should reference Section 7 and affiliated product listings.

Service Design Coordinator shall ensure alignment of Service Design Package (SDP) test plans, execution, validation, acceptance activities affiliated with application, infrastructure, data/information, security design specifications managed through service design, transition, change management, and integrated SDLC frameworks.

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

Release Management – The objective of release management is to ensure that standardized methods and procedures are used for defining executable solution deployment strategies and implementation playbooks to ensure efficient and successful delivery of all software releases with minimal impact the integrity of existing services and/or business operations. Release management practices are to be applied to all software development lifecycles as well as hardware, documentation, processes, and other components of a service. Release management focuses on strategic planning, scheduling, and controlling the movement of releases between development, staging, and production environments. Release management should include a release package, a set of configuration items to be built, tested, and deployed as a single release.

At a minimum, release management processes shall include:

- Use of proper change management (discussed below) with appropriate deployment, communications, and backout procedures documented as part of the process.

- Non-workday maintenance period - unless required by a break-fix situation, deployments shall take place during non-working hours, preferably during established maintenance windows. This is to be gauged by peak and valley usage of the application.

- Proper Production Configuration Files:

    - Normal Operating State – In a normal operating state, production configuration files must not contain usable or unusable keys or values that reference lower environments.
    - Normal/Scheduled Deployments – In normal production deployments configuration files that need to be included in the deployment package must not contain usable or unusable keys or values that reference lower environments.
    - Break/Fix Deployments – In break/fix situations that require connectivity with lower environments for troubleshooting or testing, the application is to be placed into a break/fix state that ensures outside users cannot access the system. Once the issue has been resolved, the configuration file will be modified back to its normal operating state version and any lower environment information must be removed from the production configuration file. At that point, the application can be placed back into its normal operating state.

- Usage of a code management / team collaboration tool. Ensure processes are administered via a code management tool, ensuring proper branch and merge, and build. Such usage can aid in a code review of what's been updated for that given release. Ensure staff are properly trained on the tool and the procedures it is administering.

Additionally, the various environments which are part of the application lifetime (e.g. Development, User Acceptance Testing (UAT), Staging, Production) are to be segregated with different access management – separate and distinct service accounts, passwords, test accounts, etc. – so that actions taken with a privileged account in a lower environment cannot be applied to another environment with the same privileged account.

The Service Design Coordinator shall evaluate systems and operational readiness assessments and monitor service transition activities and integrated SDLC frameworks in collaboration with business and service operations stakeholders.

Change Management – The objective of change management is to ensure that standardized methods and procedures are used for efficient handling of all changes to minimize the impact of change-related incidents and to improve day-to-day operations. Change management is the process of documenting change requests, analyzing feasibility, planning, implementing, and verifying changes to a system. Change requests can be initiated by the requirement for a new feature, by the requirement to fix a problem, or by the requirement to change the way a business function is performed. A request is accompanied by all the relevant information about the proposed change and the change initiator. Analyzing the change request involves assessing the urgency, assigning priority, performing technical and economic feasibility, and risk and impact analysis. Designated approvers use this information to approve or reject the

change request. Planning refers to developing specific technical requirements, scheduling, identifying the implementation and back-out strategy, and receiving approval. Implementation includes deployment and propagation of the change, as well as testing and documentation updates. Before closing the change request in the change management system, the change is verified by the stakeholders.

Application Inventory – The integration of an application inventory also provides a valuable tool for risk assessments and business continuity planning. The inventory identifies risks associated with technology maturity (software and hardware), compliance, sustainability, audits, supportability, recoverability, etc. It contains vital information regarding business criticality and physical location of assets which will provide valuable insights to agency and enterprise stakeholders to prepare for continuity plans, assessing impacts on business and/or IT operations, and assist in planning and making key decisions regarding modernization strategies, IT project and investments priorities.

The application inventory also provides agency CIOs and IT Managers with a resource to prepare their agency IT strategic plans and to ensure alignment with agency and enterprise business and IT strategic initiatives. In addition, the application inventory provides an additional mechanism for agency portfolio/project managers to understand potential risks/impacts, align business and IT strategies with IT projects initiatives to meet agency business goals and objectives.

Systems development life cycles need to have a long-term strategy to maintain value and alignment with ever-changing business function requirements. Key components that a mature SDLC strategy will capture include:

- Detailed demonstration of continued value to the enterprise
- A defined plan of communication and understanding between application development teams, key stakeholders, and business users using:
    - Service design packages
    - Documented workflow and decision-tree models
    - Implementation of a knowledge management process
- Integration of application security programs and processes in all SDLC processes
- A proper balance between delivery of systems based on business requirements and cost-effectiveness processes
- New and modernized application/system software designs that consider the hardware and software roadmaps that support the systems

6. Standards

Current Standards for Application Testing Tools

(These technologies meet the requirements of the current architecture and are recommended for use.)

Technology	Platforms	Technology Classification
Microsoft Visual Studio Team Services (VSTS)	Windows	Current
IBM Rational Team Concert (RTC)	Windows	Current
IBM Rational Performance Tester	Windows	Current

IBM Rational Robot	Windows	Current
IBM Rational Functional Tester	Windows	Current
IBM Rational PurifyPlus	Windows	Current
IBM Rational Test Manager	Windows	Current
HP LoadRunner	All	Current
HP/Quick Test Professional Version	Windows	Current
HP Quality Center	All	Current
Microsoft Team Foundation Server (TFS)	Windows	Current

Contain Standards for Application Testing Tools

(These technologies no longer meet the requirements of the current architecture and are not recommended for use. They are to be phased out over time. No date has been set for their discontinuance.)

Technology	Platforms	Technology Classification
Mercury WinRunner (all versions)	All	Contain

Current Standards for Application Requirements, Data, and Object Modeling Tools

(These technologies meet the requirements of the current architecture and are recommended for use.)

Technology	Platforms	Technology Classification
Microsoft Visual Studio Team Services (VSTS)	Windows	Current
IBM Rational Team Concert (RTC)	Windows	Current
IBM Rational Software Architect	Windows	Current
IBM Rational Software Modeler	Windows	Current
Computer Associates ERwin Data Modeler	Windows	Current
Microsoft Visio (Standard/Professional/Enterprise Architect Editions)	Windows	Current
Sparx Systems Enterprise Architect	Windows	Current
Sparx Systems MDG Integration for Visual Studio	Windows	Current

Contain Standards for Application Requirements, Data, and Object Modeling Tools

(These technologies no longer meet the requirements of the current architecture and are not

recommended for use. They are to be phased out over time. No date has been set for their discontinuance.)

Technology	Platforms	Technology Classification
IBM Rational Rose Modeler (all versions)	Windows	Contain
IBM Rational Rose Developer for Java (all versions)	Windows	Contain
IBM Rational Rose Developer for Visual Studio (all versions)	Windows	Contain
IBM Rational Rose Enterprise (all versions)	Windows	Contain
Borland Together	Windows	Contain
Sybase PowerDesigner (all versions)	Windows	Contain
MagicDraw UML (all versions)	Windows	Contain
Computer Associates Groundworks (all versions)	Windows	Contain

Current Standards for Application Requirements Management Tools

(These technologies meet the requirements of the current architecture and are recommended for use.)

Technology	Platforms	Technology Classification
Microsoft Visual Studio Team Services (VSTS)	Windows	Current
IBM Rational Team Concert (RTC)	Windows	Current
IBM/Rational RequisitePro	Windows	Current
HP Quality Center	All	Current
Microsoft Team Foundation Server	Windows	Current
Sparx Systems Enterprise Architect	Windows	Current

Contain Standards for Application Requirements Management Tools

(These technologies no longer meet the requirements of the current architecture and are not recommended for use. They are to be phased out over time. No date has been set for their discontinuance.)

Technology	Platforms	Technology Classification
Borland Caliber-RM	All Platforms	Contain
Telelogic DOORS	All Platforms	Contain

Current Standards for Software Configuration Management Tools

(These technologies meet the requirements of the current architecture and are recommended for use.)

Technology	Platforms	Technology Classification
Microsoft Visual Studio Team Services (VSTS)	Windows	Current
IBM Rational Team Concert (RTC)	Windows	Current
Microsoft Team Foundation Server	Windows	Current
IBM Rational ClearCase MultiSite	All	Current

Contain Standards for Software Configuration Management Tools

(These technologies no longer meet the requirements of the current architecture and are not recommended for use. They are to be phased out over time. No date has been set for their discontinuance.)

Technology	Platforms	Technology Classification
IBM Rational ClearCase	All	Contain
Microsoft Visual SourceSafe 2005 Standard Edition	Windows	Contain

Current Standards for Software Change Management Tools

(These technologies meet the requirements of the current architecture and are recommended for use.)

Technology	Platforms	Technology Classification
IBM Rational ClearQuest	All	Current
IBM Rational ClearQuest MultiSite	All	Current
Microsoft Team Foundation Server	All	Current
Microsoft Visual Studio Team Services (VSTS)	Windows	Current

Contain Standards for Software Change Management Tools

(These technologies no longer meet the requirements of the current architecture and are not recommended for use. They are to be phased out over time. No date has been set for their discontinuance.)

Technology	Platforms	Technology Classification
N/A	N/A	N/A

7. Responsibilities

7.1 Commonwealth Agencies:

- When performing application development, are required to utilize a documented SDLC framework for all new application development and enhancement projects.
- All applications that directly or indirectly support vital business functions (e.g. financial, federally-mandated, etc.) and/or are linked to a Continuity of Planning (CoP) Primary Mission Essential Business Function (PMEF) must be reported into the Enterprise Application Inventory tool (see Section 9 for location of tool). The agency must update this inventory to reflect the status and composition of their application portfolio supporting their agency mission.
- Agencies that utilize an internal agency-based application inventory are required to report their applications into the Enterprise Application Inventory tool.
- Agencies are to maintain proper alignment of their application portfolios contained in the Enterprise Application Inventory tool with their respective agency IT Strategic Plans and IT Policy compliance status (existing policy waivers or other non-compliance conditions).
- Agency applications that are public-facing must comply with ITP-SEC005 *Commonwealth Application Certification and Accreditation* procedures and requirements.
- Align and integrate ITIL Service life cycle phases with SDLC frameworks as appropriate and level of maturity evolves within commonwealth IT organizations.
- Agencies are to adhere to the software decision Principles for determining software/application needs.
- Agencies are to submit a Service Request for all SaaS, PaaS, IaaS solutions prior to consumption of solutions.

7.2 Office of Administration, Office for Information Technology Enterprise

- Will maintain an intranet-available centralized Enterprise Application Inventory tool and provide guidance and assistance to agencies utilizing that tool.
- Will maintain a Service Request process for agencies on SaaS solutions and provide a timely analysis of all SaaS solution requests.

8. Related ITPs/Other References

Definitions of associated terms of this policy are published on the Office of Administration's public portal: http://www.oa.pa.gov/Policies/Pages/Glossary.aspx

Commonwealth policies, including Executive Orders, Management Directives, and IT Policies are published on the Office of Administration's public portal: http://www.oa.pa.gov/Policies/Pages/default.aspx

- Management Directive 205.34 *Commonwealth of Pennsylvania Information Technology Acceptable Use Policy*
- ITP-BUS011 *Commonwealth Cloud Services Requirements*
- ITP-SEC000 *Information Security Policy*
- ITP-SFT001 *Software Licensing*
- ITP-SEC005 *Commonwealth Application Certification and Accreditation*
- Enterprise Application Inventory: https://itcentral.pa.gov/apps/applicationinventory/Pages/ApplicationInventory.aspx (CWOPA limited access only)
- Enterprise Application Inventory Guidelines: https://itcentral.pa.gov/apps/applicationinventory/Documents/ApplicationInventoryGuideline.docx (CWOPA access only)
- Enterprise Service Request: https://itcentral.pa.gov/Pages/Enterprise-Services.aspx (CWOPA access only)

9. Authority

Executive Order 2016-06, Enterprise Information Technology Governance

10. Publication Version Control

It is the user's responsibility to ensure they have the latest version of this publication, which appears on https://itcentral.pa.gov for Commonwealth personnel and on the Office of Administration public portal: http://www.oa.pa.gov/Policies/Pages/default.aspx. Questions regarding this publication are to be directed to RA-ITCentral@pa.gov.

11. Exemption from This Policy

In the event an agency chooses to seek an exemption, for reasons such as the need to comply with requirements for a federally mandated system, a request for waiver may be submitted via the Commonwealth of PA Procurement and Architectural Review (COPPAR) process. Requests are to be entered into the COPPAR Tool located at http://coppar.oa.pa.gov/. Agency CIO approval is required.

This chart contains a history of this publication's revisions:

Version	Date	Purpose of Revision
Original	02/17/2017	Base Document Moved to Software domain from Application, including ITP number change Merged ITP-APP012 *Systems Development Life Cycle Policy*, ITP-APP014 *Application Testing Tools Policy*, ITP-APP016 *Requirements, Data and Object*

		Modeling Tools, ITP-APP017, ITP-APP018 *Software Configuration Management Tools*, ITP-APP019 *Software Change Management Tools* into ITP Added additional guidance to Policy and Responsibilities sections
Revision	09/13/2017	Inserted language on software decision Principles Added software decision Principles Agency Responsibility Added MS VSTS to Current Standards for Software Change Management Tools Added Release Management concepts Added Service Request requirement for SaaS solutions Added additional definitions Added additional responsibilities addressing Service Request requirement for SaaS
Revision	08/15/2018	Definitions moved to online Policy Glossary Added Release Management guidance Added production designation and naming convention guidance Added Access Management guidance Clarified cloud-based solutions and review process