SOFTWARE DEVELOPMENT LIFECYCLE POLICY

NextGen CGI, LLC

Document Owner: NextGen CGI, LLC
Effective Date: January 1, 2021
Updated:

NextGen CGI, LLC Software Development Lifecycle Policy							
Effective Date:		01/01/2021		**Document Owner:**		NextGen CGI, LLC	
Revision History							
Revision	Rev. Date	Description	Prepared By	Reviewed By	Date	Approved By	Date
1.0	01/01/2021	Policy	S. Hall	S. Hall	01/01/2021	S. Hall	01/01/2021

1. Overview

In agreement with approved organizational security requirements set forth and approved by management, NextGen CGI, LLC has established a Software Development Lifecycle Policy and supporting procedures. The policy is to be implemented as soon as possible with relevant and applicable procedures. In addition, the policy is to be evaluated on a(n) [annual, semi-annual, quarterly] basis ensuring its sufficiency and relevancy towards NextGen CGI's needs and goals.

2. Purpose

This policy and supporting procedures are designed to provide NextGen CGI with a documented Software Development Lifecycle Policy that is to be utilized throughout the organization at all times. Compliance with the following policy and supporting procedures helps ensure safety and security of NextGen CGI systems.

3. Software Development Lifecycle

3.1 Goals and Objectives

- Policy goals consist of the following:
  - Deliver secure quality systems which meet customer expectations when promised and within cost estimates.
  - Provide a framework for developing quality systems using an identifiable, measurable, and repeatable process.
  - Assign roles and responsibilities of all involved parties, including functional and technical managers, throughout the system development lifecycle.
  - Make sure system development requirements are well defined and later satisfied.
- Policy objectives consist of the following:
  - Appropriate levels of management authority to provide timely direction, coordination, control, review and approval of the system development project should be established.
  - Document functional and security requirements and maintain traceability of those requirements throughout the development and implementation process.
  - Make sure projects are developed within current and planned information technology infrastructure.

3.2 Scope

- This policy applies to all parties operating within the company's network environment or utilizing information resources connection to the environment. It covers all employees, consultants and contractors, including 3rd parties, involved in the development or modification of critical applications that support NextGen CGI.

3.3 SDLC Phases

- At a minimum, the Software Development Lifecycle should include the following:
  - Planning Phase (Requirement gathering & Analysis)
    - Relevant information is collected from the customer to develop a product as per their expectations.
    - Business analyst and Project Manager set up meeting with customer to gather information like what the customer wants to build, who will be the end-user, and what is the purpose of the product.
  - System and Technical Design Phase
    - Requirements gathered are used for the software architecture and design which is used for implementing system development
    - Security requirements are also determined at this time

- Development Phase
    - Once design document is created the software can be translated into source code
    - All components of the software are implemented
- Testing Phase
    - Test once coding is complete and modules are released for testing
    - Test software thoroughly and assign project team any defects found to be fixed
    - Test software in test environment
    - Test security controls to ensure proper functionality
- Deployment Phase
    - Once testing is complete and approved, deploy software to production environment and schedule a Go-Live date
    - Make edits to software configuration as needed for client
    - Train client on functionality of software
- Maintenance/Project Conclusion Phase
    - Following project completion, begin regular software maintenance and make any final changes to software

Refer to Section 5, Roles and Responsibilities, for more details on operational procedures specific to roles.

4. Policy

NextGen CGI is to ensure that the Software Development Lifecycle Policy adheres to the following conditions for purposes of complying with the mandated organizational security requirements set forth and approved by management.

4.1 Document Software Development

- All phases of the software development cycle shall be logged, whether approved or rejected, in a standardized, central system. The approval and results of the developed software request shall be documented.
- A documented audit trail, maintained at a Business Unit level, containing relevant information shall always be maintained. This should include change request documentation, change authorization, and the outcome of the change. No single person should be able to effect changes to production information systems without the approval of other authorized personnel.
- Information resources documentation is used for reference purposes in various scenarios (e.g. further development of existing information resources as well as ensuring adequate knowledge transfer in the event of the original developer and/or development house being unavailable). It is therefore imperative that information resources documentation is complete, accurate and kept up to date with the latest changes. Policies and procedures, affected by software changes, shall be updated on completion of each change.

4.2 Risk Management

- A risk assessment shall be performed for new software and dependent on the outcome, an impact assessment should be performed. The risk assessment should take into account the following:

    - The value of the information involved
    - The classification of the information according to the scheme used within the organization
    - The environment in which the information will be accessed or processed
    - The criticality of the new system and the information it holds to ensure proper availability
    - The legal, regulatory and contractual environment the system must operate within
- The impact assessment should include the potential effect on other information resources and potential cost implications. The impact assessment should, where applicable, consider compliance with legislative requirements and standards.

4.3 Application Design

- Based on the risk assessment and the classification of the information that is to be held in and processed by the new application, the design must provide for appropriate security features to be available. This extends not only to the creation and maintenance of user accounts and permissions within the application but also the following areas:

    - Data input validation controls
    - Data flow
    - Data output
    - Interfaces with other systems
    - Reporting
    - Time stamps
    - Logging
    - Batch and transaction counters
    - Monitoring facilities
    - Use of cryptography

- The OWASP Top 10 application vulnerabilities should always be taken into consideration in the secure coding and configuration of an application. As of 2019 they are as follows:
    - Injection
    - Broken Authentication
    - Sensitive data exposure
    - XML External Entities (XXE)
    - Broken Access control
    - Security misconfigurations
    - Cross Site Scripting (XSS)
    - Insecure Deserialization

- Using Components with known vulnerabilities
- Insufficient logging and monitoring

4.4 Access Control

- Separation of Development, Testing, and Operational environments shall be required. This includes both logical and physical separation as well as separation of duties for those developing, testing, and deploying the system or application.
- Adequate controls must be put in place to protect test data. Whenever possible, sensitive data should be pseudonymized or tokenized in non-production environments. Any production data that is used outside of production must be deleted after it is no longer needed.
- Developers should have their own user accounts and should only have access to production environments temporarily and for troubleshooting purposes only. Care should be taken so that the creation of user accounts for developers does not involve the assignment of excessive privileges that will remain after go-live.

4.5 Change Management

- The change control process shall be formally defined and documented. A change control process shall be in place to control changes to all critical company information resources (e.g. hardware, software, system documentation, and operating procedures). Further details can be found within the *Change Management Policy.*
- All change requests should be prioritized in terms of benefits, urgency, effort required and potential impact of operations following conclusion of software development.
- The impact of changes on existing SLAs should be considered. Where applicable, changes to the SLA should be controlled through a formal change process which includes contractual amendments as necessary.
- All changes should be approved prior to implementation. Approval of changes should be based on formal acceptance criteria, such as ensuring that change requests were performed by an authorized user and impact assessments were performed and proposed changes were tested.
- All users that are or will be significantly affected by the change should be notified of the change. The user representative should sign-off on the change, and users should be required to make submissions and comments prior to acceptance of the change.

4.6 Testing

- Software should be tested in an isolated, controlled, and representative environment (where such an environment is feasible) prior to implementation to minimize the effect on the relevant business process, to assess its impact on

operations and security and to verify that only intended and approved changes were made.

- A separate exercise of security testing should be carried out against the security requirements that were established during the business requirements and design stages based on the risk assessment results.
- Additional security testing should occur before any major update and at least annually

4.7 Version Control and Backups

- Any software change and/or update should be controlled with version control. Older versions should be retained in accordance with corporate retention and storage management policies.
- Developed code and other types of program and configuration components should be stored in a secure repository to which access is restricted according to the access control policy in force for the project. Admin access to the source code should be limited to appropriate personnel approved by management.
- Adequate measures should be put in place to ensure that regular backups are taken of the development and other environments. Although the system may not be live, the cost in man days of the loss of development effort could be considerable.

4.8 Implementation

- Implementation will only be performed after appropriate testing and approval by stakeholders. Software Development should be treated as new system implementations and should be established as a project. Projects will be classified according to effort required to develop and implement the software.
- Initial implementation and any future changes will be monitored once they have been rolled-out to the production environment. Deviations from design specifications and test results will be documented and escalated to the solution owner for ratification.
- Deployment of the software packages to production should be performed by personnel who did not develop the code to ensure separation of duties. All changes and deployment of code shall follow the Change Management Process to ensure proper notification, authorization and that contingency procedures are in place.

5. Roles and Responsibilities

Role	Functional Responsibilities
Chief Information Officer	• Responsible for overseeing resources needed and technical review of all projects and resource allocations • Approval of project
Project Manager	• Responsible for the technical aspects of the project including identifying the necessary roles, plans, schedules, and identifying project risk factors

	• Responsible for directing the development of the project plan and assigning roles to staff members on the project team • Responsible for the accuracy and completeness of all deliverables, and for reporting on project progress • Responsible for communicating with client and ensuring documentation and training is provided
Development Team	• Evaluate technical risk factors • Developing the system and application • Responsible for smoke testing software to insure it can be passed to quality assurance
Business Analyst	• Plan and conduct the collection of data and formation of Requirements Document • Work with client to identify the appropriate means for data collection and vetting of the final document • Responsible for making sure the System Specifications accurately reflect the Requirements Document and get client approval of any changes • Responsible for product documentation • Develop and schedule demonstrations of software and training with client
System Architect	• Responsible for overall design and development process • Responsible for risk assessment and the technical properties of the system and acceptance tests • Responsible for making sure the hardware and software systems are available and properly configured • Responsible for properly completing unit tests • Develop a deployment plan and get approval • Oversee deployment process
Quality Assurance Team	• Responsible for testing the product according to the system test plan • Reports problems to development team and update the System Administration and Deployment plans • Once system testing has been passed, report to Project Manager • Develop plan for maintaining product and integrating with the defect management system
Security Lead	• Responsible for making sure that security and privacy concerns are identified and accounted for in the planning and design process • Ensure security testing is performed before implementation to ensure controls are working as designed

6. Compliance

Any person, subject to this policy, who fails to comply with the provisions as set out above or any amendment thereto, shall be subjected to appropriate disciplinary or legal action in accordance with the NextGen CGI Disciplinary Code and Procedures. Company Information Security policies, standards, procedures and guidelines shall comply with legal, regulatory and statutory requirements.

7. Related Standards, Policies, and Processes

- Change Management Policy
- Data Protection Standard
- Security Patches and Vulnerability Assessment Policy
- Asset Management Policy
- Access Control Policy

8. Definitions and Terms

The following definition are not all-inclusive and should be updated as new information is made available:

Term	Definition