SOFTWARE DEVELOPMENT LIFECYCLE POLICY

CHANGE HISTORY

Date	Version	Created by	Approved By	Description of change
08/10/2024	1.0	Bharath S	Ashwin Khosla	Basic document outline and submitted for review

Purpose

The purpose of the Software development lifecycle policy is to outline the procedures for establishing and maintaining baseline protection standards for company software development procedures.

Scope

This policy applies to following

- Information Systems
- All in-house built information systems
- Source code and source code controls systems
- Inhouse and 3rd party developer and engineering teams

Background

The intent of this policy is to ensure a well-defined, secure and consistent process for managing the entire lifecycle of software and information systems, from initial requirements analysis until system decommissioning. The policy defines the procedure, roles, and responsibilities, for each stage of the software development lifecycle. The software development lifecycle consists of requirements analysis, architecture and design, development, testing, deployment/implementation, operations/maintenance, and decommissioning. These processes may be followed in any form such as a waterfall model, agile development model, or any other iterative methodology.

Policy

To ensure high quality and secure development of applications and services, **Veranda Learning** has established the following.

Additional training for all engineering and development staff

- In addition to the mandatory corporate awareness training, all developers, product managers, and testers are required to take compulsory training in principles such as OWASP 10 and other Secure Coding practices.
- Additional training for system administration for information systems that developers will have access to as part of their day-to-day job.

Standards and practices

- Each development teams follow its specific or corporate secure coding guidelines and standards based on the language and platform.
- The guidelines are updated every six months.
- New team members are required to read and understand the guidelines.

Development and Release approach

- All in-house software follows agile development practices and aligns with principles of continuous integration and continuous deployment.
- The complete agile lifecycle is managed in an agile lifecycle management tool.
- As a part of the agile practices, each requirement is broken down into smaller units of work called STORIES. Each Story is reviewed by the team and tagged for potential Security impacts such as Confidential Data Access, Potential vulnerability exposure, etc.

- Potential security controls are tagged, and a requirement and design review are performed with the Infosec team.
- Each release is tagged with possibly impacted security controls.
- The security team is involved and performs a security review for each major release.

Source code management and version control

- All code is managed in source control such as Github or Bitbucket
- All code follows a strict check-in, checkout, merge and pull request process to manage changes and track versions
- All main branches of source code are protected from deletion.

Other secure practices

- Segregation of environments - Dec, Test, and Production environments are on entirely separate and disconnected networks.
- Role-based access - Engineers are provided access to source code, data repositories, and data demanded by their job and role.
- Review and evaluation of new tools before new tools and products are introduced.
- Environment and system hardening based on recommendations and practices of tools and platforms used

Testing & validation

- Automate testing wherever possible using tools like
  - POSTMAN
- The QA team performs manual and functional testing and all bugs and resolutions are documented in the following tools.
  - JIRA
- A vulnerability scan of the application is performed every quarter or after every major release.
- Penetration testing is performed every six months.
- Data retention is per the Data Retention Policy.

Mapping with Industry Standards

This policy addresses the following risks related Software Development Lifecycle and Information Security standards, frameworks:

Risk	ISO 27001:2022
Unauthorized access	8.26 Application security requirements 8.32 Change management 8.31 Separation of development, test and production environments 8.30 Outsourced development 8.33 Test information 8.18 Use of privileged utility programs

Risk	ISO 27001:2022
Information Leakage	8.33 Test information 8.32 Change management
Loss of CIA (Confidentiality, Integrity, Availability)	8.26 Application security requirements 8.30 Outsourced development 8.31 Separation of development, test and production environments 8.32 Change management 5.8 Information security in project management. 8.27 secure system architecture and engineering principles
Malware	8.32 Change management 8.25 Secure development life cycle 8.28 Secure coding 8.29 Security testing in development and acceptance 8.27 secure system architecture and engineering principles

Enforcement

This policy is maintained and enforced by the Infosec team.

Non- Compliance

Violations of the policy may subject employees to disciplinary action, including removal of privilege to the systems, up to and including termination of employment. Sanctions for non-compliance may include, but are not limited to, one or more of the following:

- Disciplinary action according to applicable Veranda Learning policies.
- Termination of employment.
- Legal action according to applicable laws and contractual agreements.

Validity and Document Management

The owner of this document is the management, who must check and, if necessary, update the document at least once every 12 months.