

Software Development Lifecycle Policy

Revision: *Initial Release*

Effective: 2026-01-05



UNCONTROLLED COPY: ALWAYS VERIFY REVISION

<https://itservices.machspec.com/policies>

1 Purpose

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

Objectives

1. Deliver secure quality applications which meet expectations
2. Provide a framework for developing quality applications
3. Assign roles and responsibilities

2 Scope

This policy applies to all parties operating within the Machine Specialties' 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 which support Machine Specialties.

3 Definitions

SDLC

Software Development Lifecycle

4 Policy

At a minimum, the Software Development Lifecycle should include the following:

1. Planning Phase (Requirement gathering and 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.

2. 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

3. Development Phase

- Once design document is created the software can be translated into source code
- All components of the software are implemented
- Create comprehensive documentation for administrators and end users

4. 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

5. 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

6. Maintenance/Project Conclusion Phase

- Following project completion, begin regular software maintenance and make any final changes to software

4.1 Risk Assessment

A risk assessment must be performed for all new software and, if deemed necessary, an impact assessment may be required. The risk assessment should take into account the following:

- Regulatory and/or contractual requirements for data storage and processing
- Criticality of the new system to ensure proper availability
- The legal, regulatory, and/or contractual hosting environment requirements

The impact analysis should include the potential effect on other information resources and potential cost implications. The impact assessment should, where applicable, consider compliance with regulatory requirements and standards.

4.2 Application Design

Applications must provide the appropriate security measures based on the classification of information the system will store, transmit, and/or process. The design process must also consider regulatory and MSI specific policies. The OWASP Top 10 application vulnerabilities should be taken into consideration to ensure secure coding and configuration of the application. The latest version can be retrieved from <https://owasp.org/Top10/>

4.3 Access Control

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.

Software must include adequate access control measures to meet the requirements set in the Access Management policy.

4.4 Change Management

All production applications and application code must follow the Change Management policy. Before changes are released, relevant documentation must be updated.

4.5 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.

4.6 Version Control

Any software changes must be tracked using Git and uploaded to the Machine Specialties GitHub page. <https://github.com/machspec>

4.7 Deployment

Once software has passed all reviews and tests and comprehensive documentation has been created, it will be scheduled for deployment according to the Change Management policy.

5 Related Documents

- Change Management Policy
- Logging and Monitoring Policy
- Code Review Process

6 Enforcement

The software development lifecycle policy is enforced by the Director of IT or designated individual. Noncompliance and violations to this policy will be dealt with according to the sanctions policy.

7 History

Date of Change	Revision	Responsible	Summary of Change
2025-12-26	-	Owen Jacobs	Initial release