

STLC

Presented by SecureTest Partners

Waterfall VS Scrum

- Sequential phases: Requirements, Design, Implementation, Testing, Deployment, Maintenance
- Extensive upfront planning
- Fixed scope, timeline, and budget
- Minimal flexibility for changes
- Well-suited for stable, well-understood projects

- Iterative sprints: Plan, Develop, Test, Review
- Adaptive planning based on prioritized requirements
- Emphasis on collaboration and cross-functional teams
- Continuous improvement through regular retrospectives
- Ideal for complex projects with evolving requirements

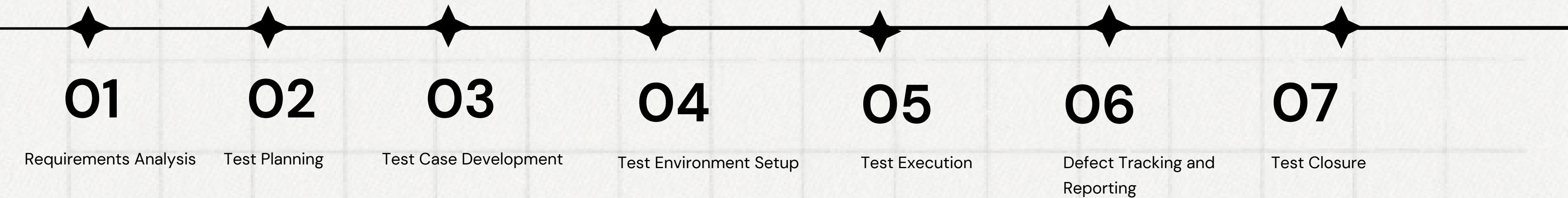
Introduction to STLC


STLC stands for Software Testing Life Cycle. It refers to a set of activities and processes performed during the testing phase of software development. STLC encompasses various stages, each with its own objectives and deliverables, aimed at ensuring the quality and reliability of the software product.

Importance of STLC

- Quality Assurance
- Risk Mitigation
- **Cost Reduction**
- **Customer Satisfaction**
- **Compliance and Standards**
- **Continuous Improvement**
- Risk Management

Key Stages of STLC





Phase 1 – Requirements Analysis

01.

Understanding the software requirements

02.

Identifying testable requirements





Phase 2 – Test Planning

01.

Creating a test plan

02.

Identifying test strategies and techniques

03.

Resource allocation and scheduling





Phase 3 – Test Case Development

01.

Writing test cases based on requirements

02.

Reviewing and validating test cases





Phase 4 – Test Environment Setup

01.

Establishing test environments

02.

Configuring hardware and software
components

03.

Data preparation for testing





Phase 5 – Test Execution

01.

Running test cases


02.

Logging test results

03.

Reporting defects





Phase 6 – Defect Tracking and Reporting

01.

Logging defects


02.

Prioritizing and assigning defects

03.

Tracking defect resolution progress





Phase 7 – Test Closure

01.

Evaluating test completion criteria

02.

Analyzing test results and metrics

03.

Preparing test closure reports



Importance of Documentation in STLC

Documentation in Each Phase

- Requirement Analysis: Documenting requirements, testing scope, and objectives.
- Test Planning: Creating comprehensive test plans outlining test strategy, approach, and resource allocation.
- Test Design: Developing detailed test cases and test scenarios based on requirements.
- Test Execution: Documenting test execution results, defects, and test coverage.
- Test Closure: Generating test closure reports summarizing testing activities and outcomes.

Comprehensive Documentation

- Facilitates communication and collaboration among project stakeholders.
- Provides a reference for future testing cycles, maintenance, and enhancements.
- Ensures traceability between requirements, test cases, and defects.
- Aids in compliance with regulatory standards and audits.
- Supports knowledge transfer and onboarding of new team members.

Examples of Key Documents

- Test Plan: Outlines testing objectives, scope, resources, and schedule.
- Test Cases: Detailed descriptions of test scenarios, inputs, expected outcomes, and steps to execute tests.
- Defect Reports: Documents defects found during testing, including descriptions, severity, and steps to reproduce.



Collaboration and Communication Among Teams

- Encourage open communication and collaboration between development, testing, and other project stakeholders.
- Foster a culture of teamwork to ensure alignment of goals, expectations, and deliverables across teams.

Automation of Repetitive Tasks

- Implement test automation to streamline repetitive testing tasks, such as regression testing, functional testing, and performance testing.
- Leverage automation tools and frameworks to increase efficiency, reduce manual effort, and improve test coverage.

Best Practices in STLC

Early Involvement of Testing in the SDLC

- Integrate testing activities early in the software development lifecycle (SDLC) to identify defects and issues sooner.
- Involve testing teams in requirements gathering and design phases to ensure testability and quality from the outset.


Continuous Improvement and Learning

- Foster a culture of continuous improvement and learning within the testing team.
- Encourage regular retrospectives to reflect on testing processes, identify areas for improvement, and implement corrective actions.
- Invest in training and skill development to keep testing teams updated with emerging technologies, methodologies, and best practices.

Challenges

Faced During STLC

- Lack of Clear Requirements: Incomplete, ambiguous, or changing requirements can lead to difficulty in defining test scope and criteria.
- Time and Resource Constraints: Limited timeframes and resources may impact the thoroughness and effectiveness of testing activities.
- Maintaining Test Environments: Ensuring availability and consistency of test environments with required configurations can be challenging.

The background is a light gray grid. It is decorated with various hand-drawn blue doodles, including circles, loops, a starburst, a zigzag line, and several checkmarks.

Thank you very much!

www.securetestpartners.com