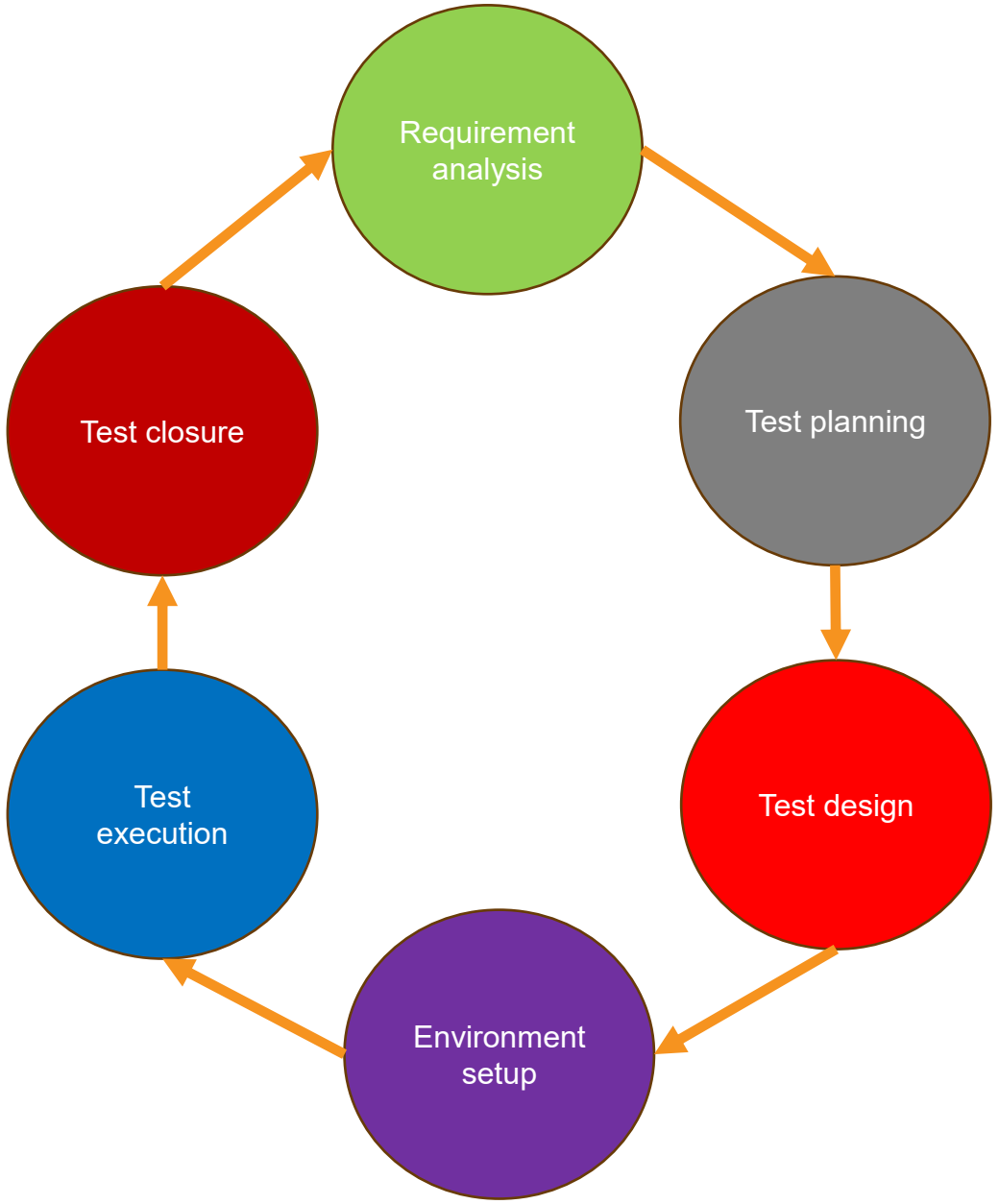
Software testing

Software Testing Life Cycle (STLC)

Jouni Juntunen, 10.12.2025

Overview

- The STLC is a sequence of specific activities conducted during the testing process to ensure software quality
- STLC is systematic, repeatable, and improves the effectiveness of testing
- Phases
  - Requirement analysis
  - Test planning
  - Test design
  - Test environment setup
  - Test execution
    - (Defect reporting and tracking)
  - Test closure



OAMK

Requirement analysis

- **Purpose**
  - o Understand what needs to be tested
- **Activities**
  - o Analyze requirements
  - o Identify testable items
  - o Clarify doubts with stakeholders
- **Deliverables**:,
  - o Requirement traceability matrix
  - o List of test requirements

Test planning

- **Purpose**
  - o Define the scope, approach, resources, and schedule for testing
- **Activities**
  - o Estimate effort
  - o Assign roles
  - o Select methods and tools
  - o Identify risks
- **Deliverables**
  - o Test plan document
  - o Resource plan
  - o Risk assessment

Test design

- **Purpose**
  - o Design and document test cases and test data
- **Activities**
  - o Write test cases
  - o Prepare test data
  - o Review and get approval
- **Deliverables**
  - o Test cases
  - o Scripts
  - o Data sets

Test environment setup

- **Purpose**
  - o Prepare the hardware and software environment for testing
- **Activities**
  - o Set up test servers
  - o Install applications
  - o Configure tools
- **Deliverables**
  - o Test environment ready for testing (and report)
  - o Environment configuration document

Test execution

- **Purpose**
  - o Run test cases and log results
- **Activities**
  - o Execute tests
  - o Record outcomes
  - o Report defects
- **Deliverables**
  - o Test execution reports
  - o Defect logs

Defect reporting and tracking

- **Purpose**
  - o Identify, report, and track defects to closure
- **Activities**
  - o Log defects
  - o Retest after fixes
  - o Update status
- **Deliverables**
  - o Defect reports
  - o Updated test cases

Test closure

- **Purpose**
  - o Finalize and evaluate the testing process
- **Activities**
  - o Assess test completion
  - o Document lessons learned
  - o Archive artifacts
- **Deliverables**
  - o Test summary report
  - o Lessons learned document

SLDC in Agile projects

- Requirements are flexible, and therefore testing cannot be extensively planned beforehand
- Use of test automation is emphasized
- Lot of documentation (also mentioned here) is not implemented
- In a nutsheel, phases of SLCD are repeated on every iteration of agile software development process

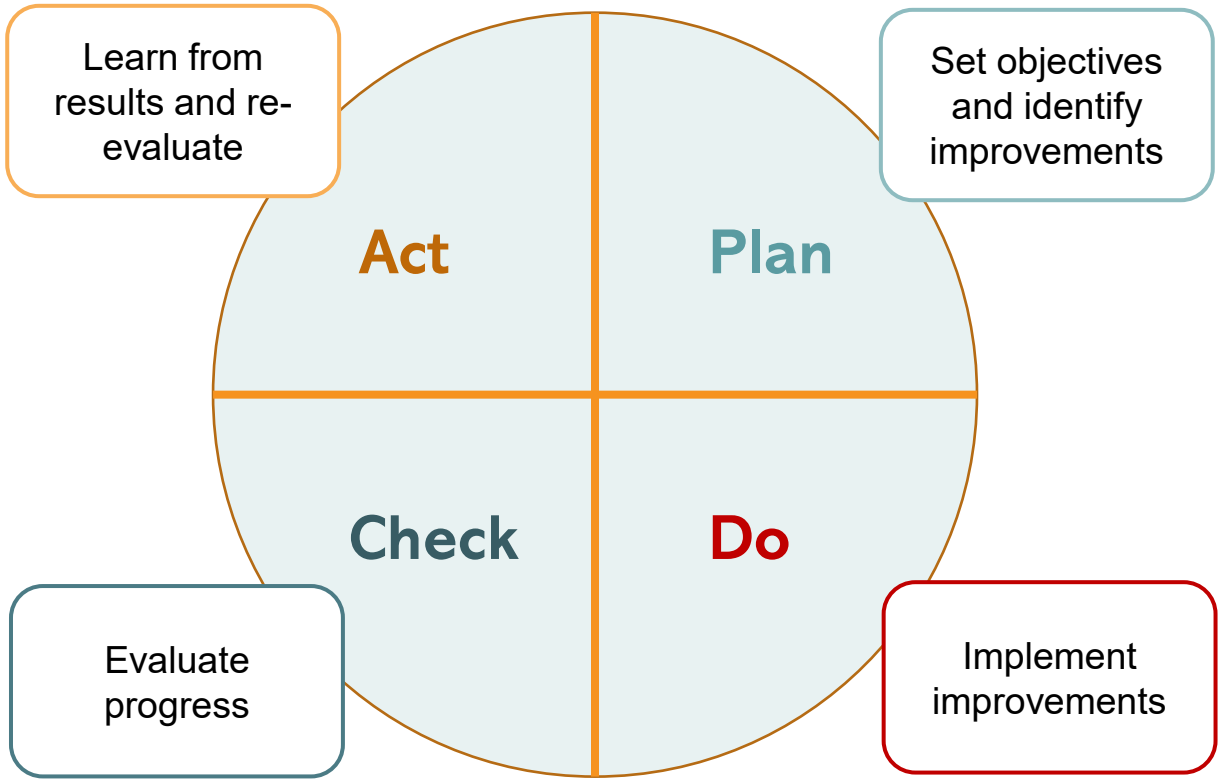
Test process improvement (TPI)

- There are always room for improvement
- Eventhough current process is satisfactory, future improvements will most likely to be required
- Aim for highest quality with lowest cost and shortest delivery time
- Review and improve testing process

Challenges in testing

- Usage of resources
  - Money
  - Time
  - Staff
  - Tools
- Quality of the test process is poor
- Testing is not well planned or managed
- Overall testing becomes troublesome and uncontrollable process
- Bugs, defects etc. caught too late on the process and fixing is expensive and time-consuming

Plan-Do-Check-Act cycle for continuous improvement in software testing



Guidelines for improving testing processes

- Analyze current status
- Plan testing and QA processes
- Shift left approach
- Take care of working environment, tools and skills
- Optimize automated testing
- Enable code quality measurement/metrics
- Review and apply improvement (that suits to your processes) continuously

Benefits of testing process improvement

- Reduce errors and downtime
- Improve software quality
- Increase efficiency and effectiveness of testing
- Reduce testing time
- Lower costs
- Better customer satisfaction
- …


OAMK
OULUN AMMATTIKORKEAKOULU