SOFTWARE QUALITY ASSURANCE AND TESTING

Assoc. Prof. Dr. Mehmet Akif Çifçi

What is Software Quality Assurance?

- Software quality assurance can be defined as "a systematic and planned management approach throughout the development process to evaluate the conformity of software work products and software development activities with the relevant standards in order to ensure software quality".

Quality Assurance Activities

- Planning
- Requirements management
- Independent verification/validation activities (fault finding methods by testing)
- Product/process review activities (use of checklists or standards to review requirements, design, code, test definitions, test results, process implementation)

Quality Assurance Activities

- Traceability/coverage analysis
- Problem solving process activities (including corrective/preventive actions)
- Monitoring compliance with defined processes and plans
- Documentation studies
- Product/process/project audits
- Risk Management
- Generation of product/process/software quality (assurance) metrics, use for improvement

What is Software Testing?

- Software testing is the actions that provide a business with information about the risks of its software before it goes to market. Software testing is the checking of whether the requirements of a piece of software or system are met during the design and development stages, whether it produces the required output against all kinds of inputs, whether it performs operations within a valid time, whether it is sufficiently useful, whether it is installed and operated smoothly on predefined hardware, whether it fulfills the customer's requests.

What is Software Testing?

- Testing can theoretically be done indefinitely, but the main goal is to ensure that testing is done in the most efficient way within the project's deadline and that the final product is as trouble-free as possible. Software testing provides independent insight into the software and creates scenarios about the risks of a failure.

- There are two main classifications of software testing; functional and non-functional.

Types of Functional Testing

- Functional testing is about testing how a product works. Performing a functional test requires checking and testing every single function of the software to make sure you get the expected results.

- Functional tests can be performed manually or by automation.

Types of Non-Functional Tests

- Non-functional testing is concerned with testing the non-functional aspects of the software, such as usability, performance, security, reliability and more. This testing is performed after functional testing. Non-functional testing does not focus on whether the software works or not, it focuses on how well it works.

- Non-functional testing helps you significantly improve the quality of your software. Non-functional testing is mostly done with automated tools because it is difficult to execute manually.

Software Quality Assurance Testing Strategies

- Unit Testing: Unit testing focuses on testing individual components or units of software in isolation. It helps detect defects early in the development process and ensures that each component works as intended. Unit testing is typically automated and forms the basis for subsequent levels of testing.
- Integration Testing: Integration testing verifies the interactions and interfaces between different modules or components of the software. It ensures that the integrated system runs smoothly and that data flows correctly between modules. Integration testing uncovers issues related to data dependencies, communication protocols and system integration.

Software Quality Assurance Testing Strategies

- Functional Testing: Functional testing verifies that the software meets the specified functional requirements. It involves testing individual features and functions to ensure that they work as expected. Testers simulate various user scenarios, input different data sets and validate the system's responses.
- Performance Testing: Performance testing evaluates the responsiveness, scalability and stability of software under different workloads. It measures factors such as response times, resource utilization and system throughput. Performance testing helps identify potential bottlenecks, optimize system performance and ensure a smooth user experience even during peak usage periods.

Software Quality Assurance Testing Strategies

- Security Testing: Security testing focuses on identifying gaps and weaknesses in the software's security mechanisms. It includes testing for potential breaches, data leaks, authorization issues and other security risks. Security testing helps protect sensitive data and protects the software from malicious attacks.
- Regression Testing: Regression testing ensures that changes or updates to software do not introduce new defects or break existing functionality. It retests previously tested components to verify that they work as intended. Regression testing helps maintain software stability and prevents regression issues.

Software Quality Assurance Testing Strategies

- Usability Testing: Usability testing evaluates the user-friendliness and ease of use of the software. Testers evaluate how intuitive the user interface is, whether users can perform tasks efficiently and whether the software meets user expectations. Usability testing provides insights to improve the user experience and interface design.

What is Continuous Integration (CI)?

- Continuous Integration (CI) is the practice where developers continuously integrate the code they write into a shared code repository to get quick feedback on the code's viability. This process supports automated builds and testing, so teams can quickly collaborate on a single project. CI also provides software companies with a more frequent and shorter release cycle, which facilitates fast and reliable release of the developed application in the live environment.

What is Continuous Integration (CI)?

- It is the first stage of the software development process.
- It is a process where developers regularly submit their code to a central repository (usually Git with a version control system).
- Each time a code submission is made, a series of tests are automatically run. These tests usually include unit tests, integration tests and sometimes static code analysis.
- This phase ensures continuous testing of the software and early detection of bugs.

Stages of Continuous Integration:

- Code Submission
- Build and Automated Tests
- Feedback and Notifications
- Repeatability

What is Continuous Delivery (CD)?

- The goal of Continuous Delivery is to deliver a packaged artifact to the production environment. CD automates the entire delivery process, including the deployment process. This can include provisioning infrastructure, managing changes, deploying artifacts, validating and tracking those changes. CD enables automated delivery of applications to selected infrastructure environments.

CI/CD Advantages

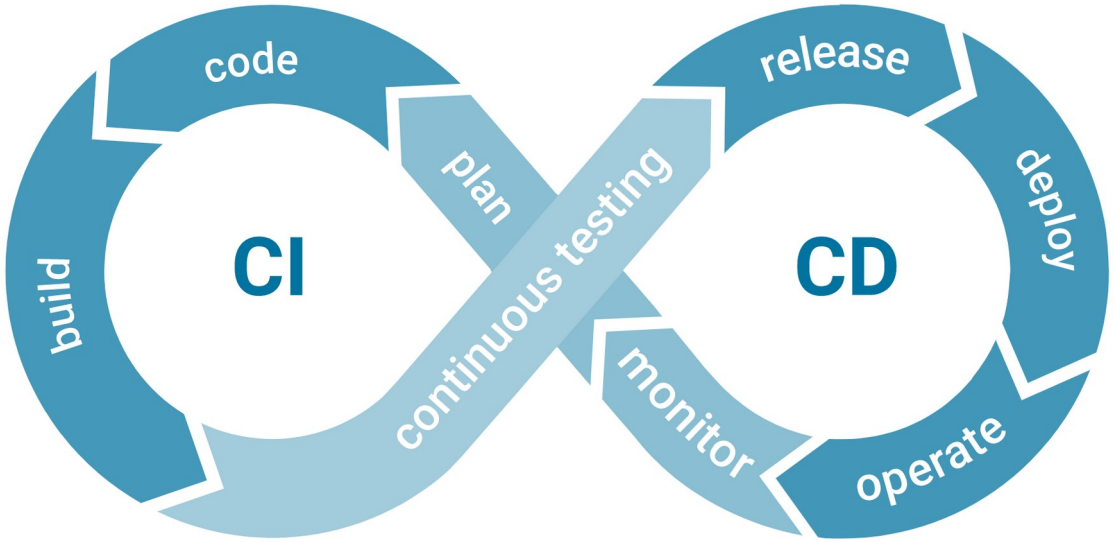
- Efficient Software Development
- Competitive Software Products
- Freedom to Fail
- Better Software Maintenance
- Better Operation Support

What are the Benefits of CI/CD?

- Early Detection of Bugs: Every code change is automatically tested, so bugs are detected early.
- Continuous Feedback: Provides developers with continuous feedback and the ability to fix bugs quickly.
- Continuous Deployment: Enables rapid deployment of software to the live environment so that new features and fixes can be rolled out quickly.
- Automation: Automates many aspects of the software development process, thus reducing human error and repetitive work.

What is the Difference Between CI and CD?

- CI aims to make software development processes easier and more traceable. CD, on the other hand, aims to deliver an artifact to the production environment safely and repeatedly. CD takes place after CI.



What are the Commonly Used CI/CD Tools?

- Jenkins Azure DevOps
- Travis CI
- CircleCI
- Jetbrains TeamCity
- GitLab
- Atlassian Bamboo
- Codeship
- Buddy
- Semaphore

Doc. Dr. Mehmet Akif Cifci

- Technical University of Vienna (Austria)
- University of Klaipeda (Lithuania)
- Bandirma Seventeen Eylul University

To Follow and Connect

https://github.com/themanoftalent
https://www.linkedin.com/in/themanoftalent/
https://www.researchgate.net/profile/Mehmet-Akif-Cifci