Software Testing Roles

สาขาระบบสารสนเทศทางธุรกิจ มหาวิทยาลัยราชภัฏจันทรเกษม

www.chandra.ac.th

AGENDA

- Seven Principles of Testing
- Types of Testing

ผู้ช่วยศาสตราจารย์ ดร.สุรชาติ บัวชุม

Introduction to the Seven Principles of Testing

The seven principles of testing provide guidelines to help testers understand the fundamental aspects of effective software testing.

ผู้ช่วยศาสตราจารย์ ดร.สุรชาติ บัวชุม

Seven Principles of Testing

Principle 1 - Testing Shows Presence of Defects

- Explanation: Testing can show that defects are present, but it cannot prove that there are no defects.
- Implication: Testing reduces the probability of undiscovered defects remaining in the software but cannot guarantee the software is bug-free.

Seven Principles of Testing

Principle 2 - Exhaustive Testing is Impossible

- Explanation: Testing everything (all combinations of inputs and preconditions) is not feasible except for trivial cases.
- Implication: Instead of exhaustive testing, risk analysis and priorities are used to focus testing efforts.

Seven Principles of Testing

Principle 3 - Early Testing

- Explanation: Testing activities should start as early as possible in the software development lifecycle.
- Implication: Early testing (shift-left testing) helps identify defects early when they are easier and cheaper to fix.

Seven Principles of Testing

Principle 4 - Defect Clustering

- Explanation: A small number of modules usually contain the most defects.
- Implication: Testing efforts should focus on these high-risk areas, often identified by past defect data and project experience.

Seven Principles of Testing

Principle 5 - Pesticide Paradox

- Explanation: Repeating the same tests will eventually find no new defects.
- Implication: Test cases need to be regularly reviewed and revised, and new tests should be written to cover different parts of the software.

Seven Principles of Testing

Principle 6 - Testing is Context Dependent

- Explanation: Testing is done differently in different contexts, such as commercial software vs. safety-critical software.
- Implication: Test strategies and methods should be tailored to the specific requirements and constraints of the project.

Seven Principles of Testing

Principle 7 - Absence-of-Errors Fallacy

- Explanation: Finding and fixing defects does not help if the system built is unusable and does not fulfill the user's needs and expectations.
- Implication: It is essential to ensure that the software meets the business and user requirements, not just that it is defect-free.

Seven Principles of Testing

Conclusion

- Summary: The seven principles of testing guide effective testing practices, ensuring that testing is both efficient and effective.
- Final Note: Understanding and applying these principles can significantly improve the quality and success of software testing efforts.

Types of Testing

Main Categories of Testing

- Functional Testing
- Non-Functional Testing
- Maintenance Testing

Functional Testing

Definition:

- Verifies that each function of the software operates according to the requirement specification.

Types:

- Unit Testing: Tests individual components or modules of the software.
- Integration Testing: Tests the combination of modules to ensure they work together.
- System Testing: Tests the complete system as a whole.
- Acceptance Testing: Tests to determine if the system meets the acceptance criteria and is ready for delivery.

Non-Functional Testing

Definition:

- Evaluates aspects of the software that do not relate to specific functions or user actions.

Types:

- Performance Testing: Assesses the speed, responsiveness, and stability under various conditions.
- Load Testing: Determines how the software behaves under a specific expected load.
- Stress Testing: Evaluates the system's robustness and error handling under extreme conditions.
- Usability Testing: Checks how user-friendly and intuitive the software is.
- Security Testing: Identifies vulnerabilities and ensures the system protects data and maintains functionality.

Maintenance Testing

Definition:

- Performed after the software has been deployed to ensure it continues to function correctly and to make improvements.

Types:

- Regression Testing: Ensures that new code changes do not adversely affect existing functionality.
- Retesting: Verifies that previously identified defects have been fixed.
- Maintenance Release Testing: Tests changes such as bug fixes and enhancements.

Types of Testing: Specialized Testing Types

Specialized Testing Types

- Alpha Testing: Internal testing performed by the development team before beta testing.
- Beta Testing: External testing performed by a select group of real users in a real environment.
- Smoke Testing: Preliminary testing to check whether the major functionalities are working.
- Sanity Testing: Focused testing to check specific functionalities after making minor changes.
- Compatibility Testing: Ensures the software works across different devices, browsers, and operating systems.

Types of Testing: Conclusion

- Summary: Different types of testing serve different purposes and are crucial for ensuring comprehensive software quality.
- Final Note: Understanding these types of testing helps in planning and executing effective test strategies to deliver high-quality software.

https://forms.gle/78B6TeBSf5fqGFHY6

Q&A