CSC 501:

SOFTWARE ENGINEERING

SOFTWARE TESTING

By

IKENNA CAESAR NWANDU PhD

Federal University of Technology Owerri, Nigeria.

What is Software Testing?

- Is a confirmation that software is doing what it is intended to do.
- Is an investigation to find out the degree in which software conforms to requirements and specifications.
- the process of executing a system in order to identify any gaps, or missing requirements in contrary to the actual requirements.
- Is a software quality assurance (SQA) issue because it scrutinizes software competence about how much they conform with the specified requirements.

Software Testing Targets:

1. **Error** – actual coding mistakes made by developers. E.g., **If there is a difference in output of software and desired output, it is considered as an error**
2. **Fault** – also known as a bug, is the manifestation of error, which can cause system to fail
3. **Failure** – the result of executing a fault. Hence, the system is unable to perform the desired task.
4. **Test Case** – Description of data that will test a system and the expected output
5. **Test Suite** – a set of related test cases

Definitions

- **General Definition**
  - **Software testing** is the execution of software program/code with the aim of discovering errors**.**

- **ANSI/IEEE 1059 Standard**
  - **Software testing** is a process of analyzing a software item to detect the differences between existing and required conditions (that is, defects/errors/bugs) and to evaluate the features of the software item.

Software Testing Approaches

1. Functionality testing
2. Implementation testing

- *Functionality Testing* – testing of software functionality without considering the actual implementation. This is referred to as **black-box testing.**
  - It is carried out to test functionality of the program
  - Here, a program is certified good if the output matches with the desired results over a set of inputs

Software Testing Approaches

- *Implementation testing* - testing of software functionality as well as the analysis of the way it is implemented. This is called **white-box testing.**
  - It is conducted to test program and its implementation.
  - It aims at improving the efficiency or structure of the program code.

Verification & Validation

Verification

- ensures that the software system meets all the functionality
- takes place first and is done by developers
- includes the checking for documentation, code, and other components
- is an objective process
- contains static activities such as collecting reviews, walkthroughs, or inspections
- Are you building the software right?

Validation

- ensures that the functionalities meet the intended behavior of the system
- done after verification by testers
- It checks the overall product to ensure that it satisfies user needs.
- is a subjective process
- involves dynamic activities such as subjecting the system to execution against the requirements
- Are you building the right software?

Levels of Software Testing

1. **Unit Testing:** Confirming that individual program code is free of error.
2. **Integration Testing:** Confirming that individual units would work (connect and interact) properly with one another without error.
3. **System Testing:** Confirming that entire compiled software works well when all components are fully integrated.
4. **Acceptance Testing:** Is the last phase of testing. End-user confirms whether software meets specified requirements

Types of Acceptance Testing

Alpha Testing:

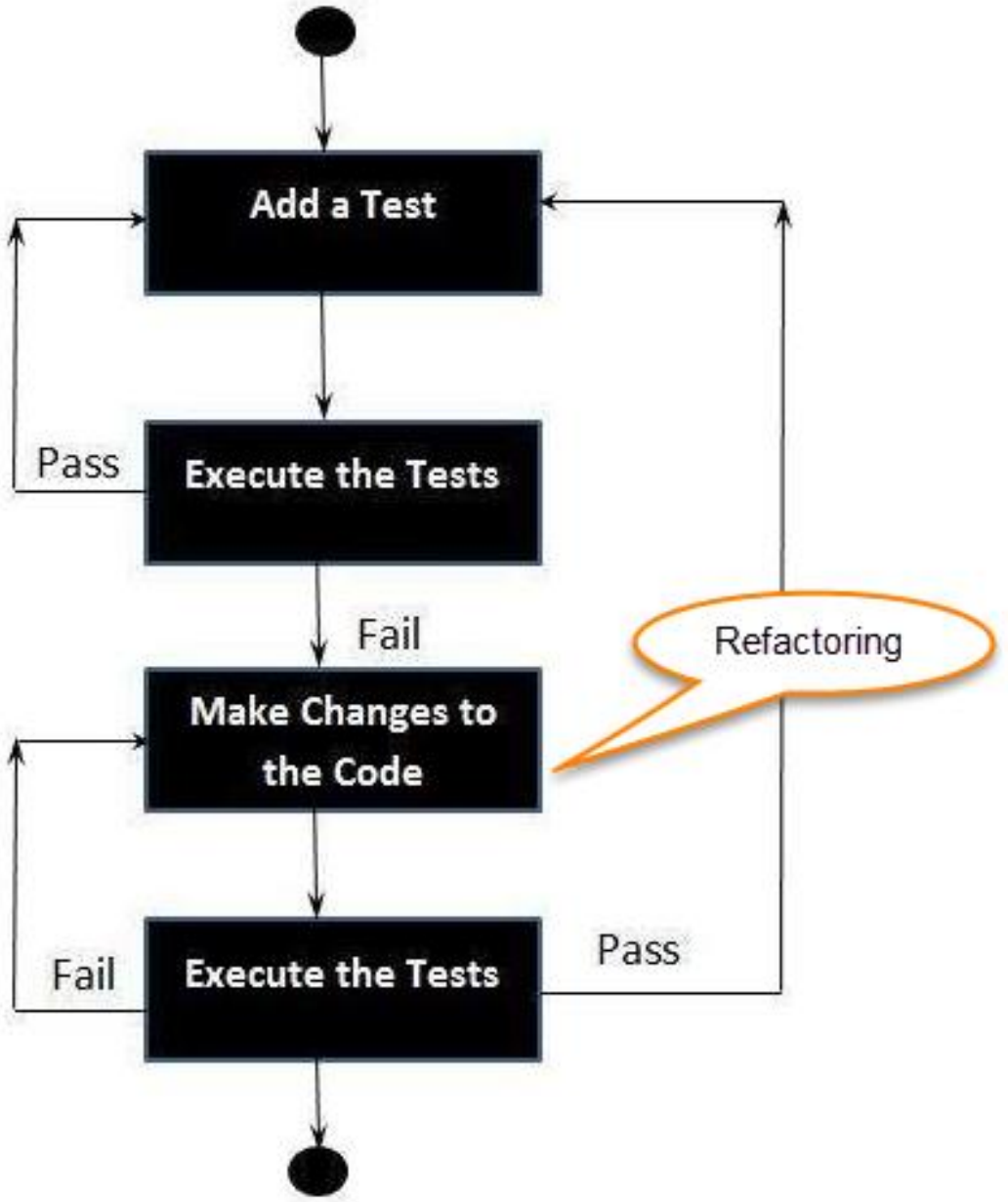
- This is an acceptance testing performed by developers using the system as if it has been deployed to the user environment. Alpha testing is meant to discover how the system would react to inputs and the potential reactions the user may have to some actions.
- **Beta Testing:**
- This is an acceptance testing conducted by users in their production environment. Beta testing expects to point out little errors that were skipped by developers with hope of having them corrected before final delivery.

Test-Driven Development (TDD)

- Is a software development method that takes testing very serious.
- It emphasizes that every code must be tested to ascertain that it runs correctly before a new line of code is written.
- Its major principle is that if a code fails a test, it must be rewritten to correct the error before new lines of code are written.
- Test cases for each functionality are created and tested first. If the functionality fails the test, the code is modified to pass the test.
- Hence, TDD is also called **Test-First Development.**

How does TDD Work?

- TDD has 5 steps:
  1. Add a test.
  2. Execute all tests and see if any one fails.
  3. Make changes and refactor code.
  4. Run tests.
  5. Repeat.



Pass. Development Stops

Review Questions