SOFTWARE APPLICATION DEVELOPMENT TOOLS & TECHNIQUES

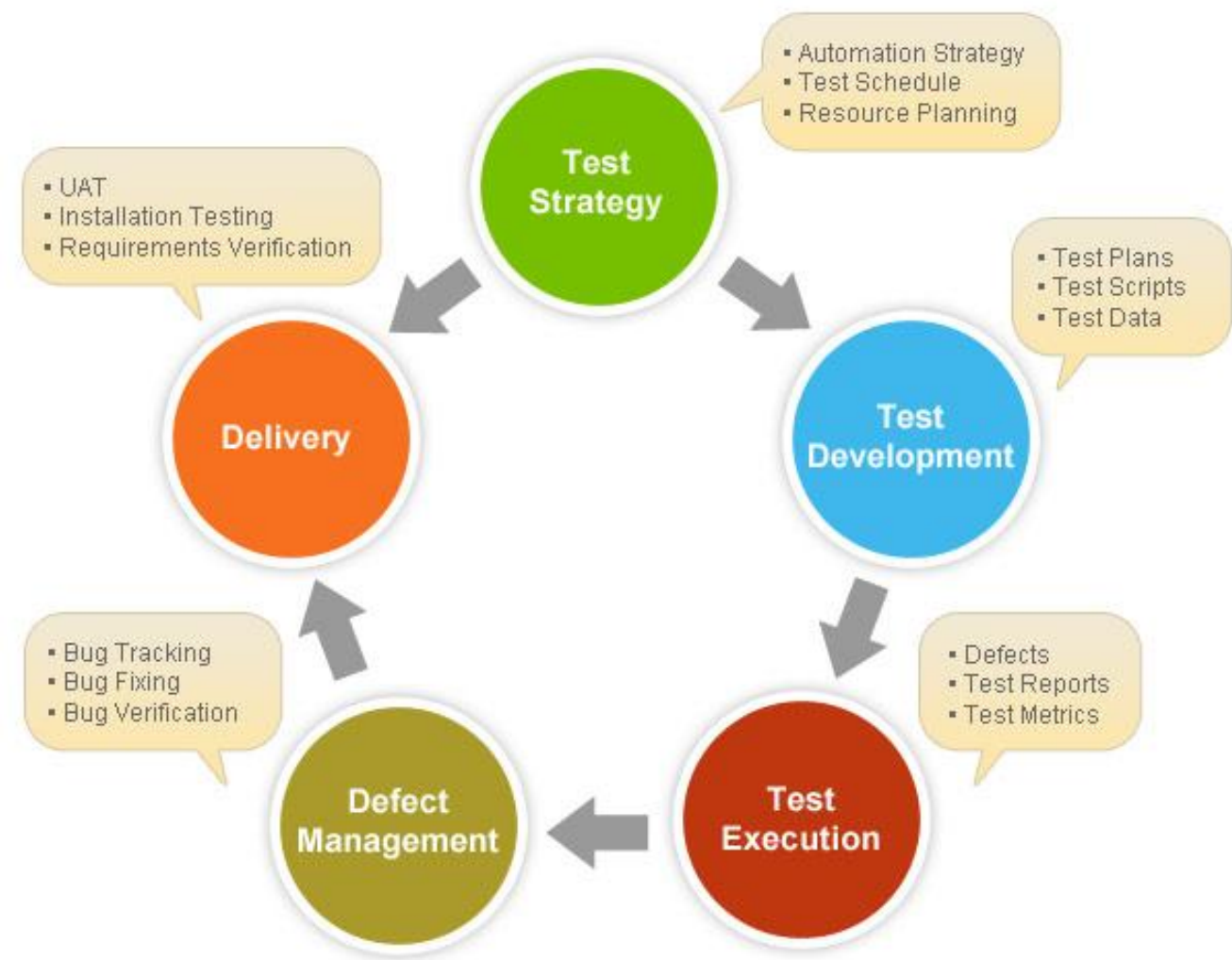
24-11-2020

Software Testing Strategy & Environment

Quotes

- Kent Beck said “Test-first code tends to be more cohesive and less coupled than code in which testing isn’t a part of the intimate coding cycle”

- “If you can’t write a test for what you are about to code, then you shouldn’t even be thinking about coding”


Test Strategy
▪ Automation Strategy
▪ Test Schedule
▪ Resource Planning
Test Development
▪ Test Plans
▪ Test Scripts
▪ Test Data
Test Execution
▪ Defects
▪ Test Reports
▪ Test Metrics
Defect Management
▪ Bug Tracking
▪ Bug Fixing
▪ Bug Verification
Delivery
▪ UAT
▪ Installation Testing
▪ Requirements Verification

TDD Overview

- Made popular by Extreme Programming
- Method of developing software not just testing software
- Software is Developed in short iterations
- Unit Tests are developed FIRST before the code

Cont..

- How It Works –

1. Add a Test
   - Use Cases / User Stories are used to understand the requirement clearly
2. Run all tests and see the new one fail
   - Ensures test harness is working correctly
   - Ensures that test does not mistakenly pass
3. Write some code
   - Only code that is designed to pass the test
   - No additional functionality should be included because it will be untested

Cont..

4. Run the automated tests and see them succeed
   - If tests pass, programmer can be confident code meets all tested requirements
5. Refactor code
   - Cleanup the code
   - Rerun tests to ensure cleanup did not break anything Repeat

Test First vs. Test Last

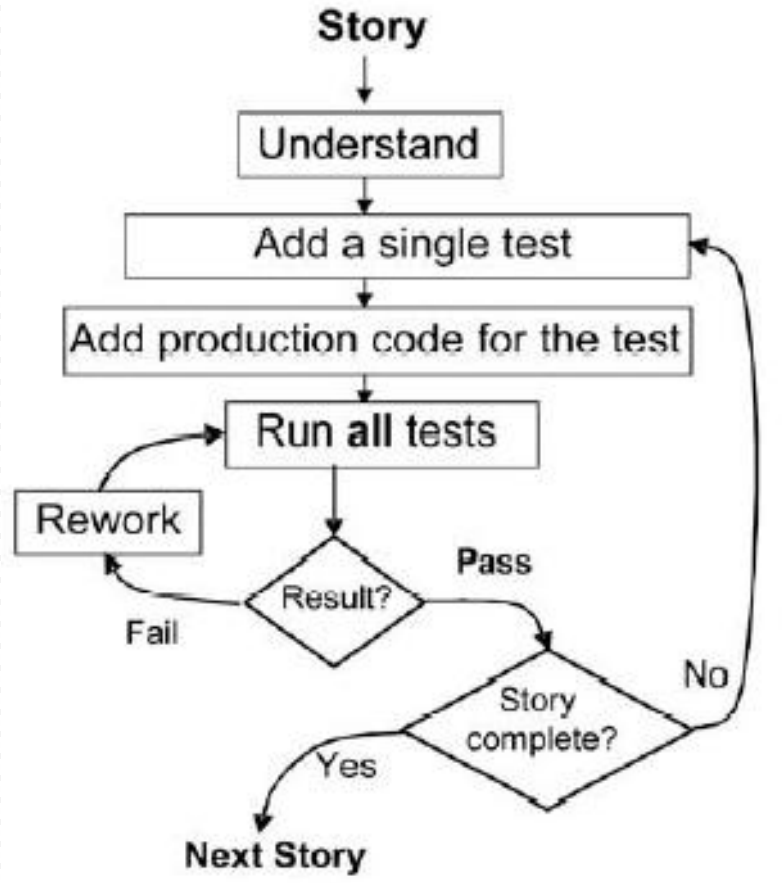
Test First

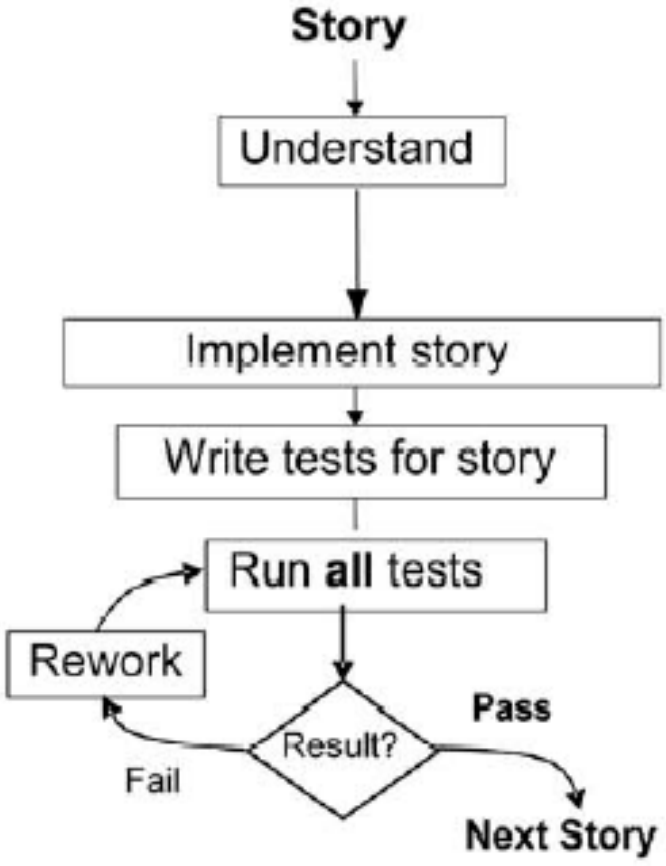
- Pick a piece of functionality
- Write a test that expresses a small task that fails
- Write production code until test passes
- Run all tests
- Rework code until all tests pass
- Repeat

Test Last

- Pick a piece of functionality
- Write production code that implements entire functionality
- Write tests to validate all functionality
- Run all tests
- Rework code until all tests pass

Cont..


Story
Understand
Add a single test
Add production code for the test
Run all tests
Rework
Result?
Pass
Fail
Story complete?
No
Yes
Next Story


Story
Understand
Implement story
Write tests for story
Run all tests
Rework
Result?
Pass
Fail
Next Story

TDD Benefits

- Instant Feedback
  - Developer knows instantly if new code works and if it interferes with existing code [1]
- Better Development Practices
  - Encourages the programmers to decompose the problem into manageable, formalized programming tasks [1]
  - Provides context in which low-level design decisions are made [1]
  - By focusing on writing only the code necessary to pass tests, designs can be cleaner and clearer than is often achieved by other methods [4]

Cont..

- Quality Assurance
  - Having up-to-date tests in place ensures a certain level of quality [1]
  - Enables continuous regression testing [2]
  - TDD practices drive programmers to write code that is automatically testable [2]
  - Whenever a software defect is found, unit test cases are added to the test suite prior to fixing the code [2]

Cont..

- Lower Rework Effort
  - Since the scope of a single test is limited, when the test fails, rework is easier
  - Eliminating defects early in the process usually avoids lengthy and tedious debugging later in the project [4]
  - "Cost of Change" is that the longer a defect remains the more difficult and costly to remove [3]

TDD Limitations

- Counterproductive and hard to learn [1]
- Difficult in Some Situations
  - GUIs, Relational Databases, Web Service
  - Requires mock objects
- TDD does not often include an upfront design [2]
  - Focus is on implementation and less on the logical structure

Cont..

- Difficult to write test cases for hard-to-test code
  - Requires a higher level of experience from programmers [2]
- TDD blurs distinct phases of software development
  - design, code and test [2]

Selenium

- Selenium Web-Driver
  - Collection of language specific bindings to drive a browser
  - https://www.seleniumhq.org/
- Selenium IDE
  - Do simple record-and-playback of interactions with the browser

Test Cases

- text box
- radio button selection
- check box selection
- drop down item selection
- keyboard actions
- mouse actions
- multi select

TestNG

- TestNG is a testing framework that is capable of making Selenium tests easier to understand and of generating reports that are easy to understand.

- TestNG over Junit
  - Annotations are easier to use and understand.
  - Test cases can be grouped more easily.
  - TestNG allows to create parallel tests.

Cont..

- Provides graphical output of the test result plus other meaningful details such as:
  - Runtimes of each method.
  - The chronological order by which methods were executed
- HTML-based reports.
- Annotations can use parameters just like the usual Java methods.

Annotations

- **@BeforeSuite:** The annotated method will be run before all tests in this suite have run.
- **@AfterSuite:** The annotated method will be run after all tests in this suite have run.
- **@BeforeTest:** The annotated method will be run before any test method belonging to the classes inside the tag is run.
- **@AfterTest:** The annotated method will be run after all the test methods belonging to the classes inside the tag have run.

Cont..

- **@BeforeGroups:** The list of groups that this configuration method will run before. This method is guaranteed to run shortly before the first test method that belongs to any of these groups is invoked.
- **@AfterGroups:** The list of groups that this configuration method will run after. This method is guaranteed to run shortly after the last test method that belongs to any of these groups is invoked.

Cont..

- **@BeforeClass:** The annotated method will be run before the first test method in the current class is invoked.
- **@AfterClass:** The annotated method will be run after all the test methods in the current class have been run.
- **@BeforeMethod:** The annotated method will be run before each test method.
- **@AfterMethod:** The annotated method will be run after each test method.