Agile Software Development (TCS 855)

Unit-IV Agile Testing

Agile Testing Life Cycle, Open Source Testing Tools and Exploratory Testing

Prof.(Dr.) Santosh Kumar

Department of Computer Science and Engineering

Graphic Era Deemed to be University, Dehradun

Agile Testing

- **Agile Testing** is a type of software testing that follows the principles of agile software development to test the software application.
- All members of the project team along with the special experts and testers are involved in agile testing.
- Agile testing is not a separate phase and it carried out with all the development phases i.e. requirements, design and coding and test case generation.
- Agile testing takes place simultaneously through the Development Life Cycle.
- Agile testers participate in the entire development life cycle along with development team members and the testers help in building the software according to the customer requirements and with better design and thus code becomes possible.
- Agile testing team works as a single team towards the single objective of achieving quality. Agile Testing has shorter time frames called iterations or loops.
- This methodology is also called delivery driven approach because it provides a better prediction on the workable products in less duration of time.

Agile Testing Principles

- **Shortening feedback iteration:**
  In Agile Testing, testing team get to know the product development and its quality for each and every iteration. Thus continuous feedback minimizes the feedback response time and the fixing cost is also reduced.

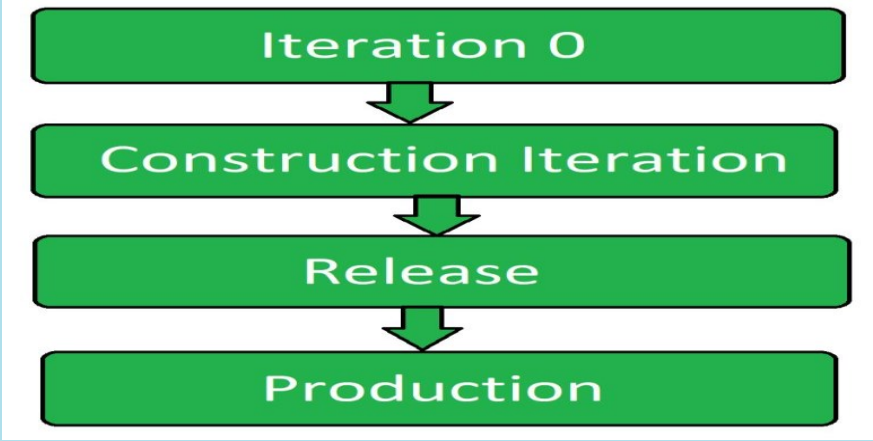
- **Testing is performed alongside:**
  Agile testing is not a different phase. It is performed alongside the development phase. It ensures that the features implemented during that iteration are actually done. Testing is not kept in pending for a later phase.

- **Involvement of all members:**
  Agile testing involves the each and every member of the development team and the testing team. It include various developers and experts.

- **Documentation is weightless:**
  In place of global test documentation, agile testers use reusable checklists to suggest tests and focus on the essence of the test rather than the incidental details. Lightweight documentation tools are used.

- **Clean code:**
  The defects that are detected are fixed within the same iteration. This ensures clean code at any stage of the development.

Agile Testing Life Cycle



1. **Iteration 0:**
   It is the first stage of the testing process and initial setup are performed in this stage. Testing environment is set in this iteration.

2. **Construction Iteration:**
   It is the second phase of the testing process. It is the major phase of the testing and most of the works are performed in this phase. It is a set of iterations to build an increment of the solution.

3. **Release:**
   This phase includes the full system testing and the acceptance testing. To finish the testing stage, the product is tested more relentless while it is in construction iterations. In this phase testers work on the defect stories.

4. **Production:**
   It is the last phase of the agile testing. Product id finalized in this stage after removal of all defects and issues raised.

Agile Testing Activities

- Agile testing includes the following activities:
- Participating in iteration planning
- Estimating tasks from the view of testing
- Writing test cases using the feature descriptions
- Unit Testing
- Integration Testing
- Feature Testing
- Defect Fixing
- Integration Testing
- Acceptance Testing
- Status Reporting on Progress of Testing
- Defect Tracking

Agile Test Plan

Agile test plan includes types of testing done in that iteration like test data requirements, infrastructure, test environments, and test results. Unlike the waterfall model, in an agile model, a test plan is written and updated for every release. Typical test plans in agile includes

- Testing Scope
- New functionalities which are being tested
- Level or Types of testing based on the features complexity
- Load and Performance Testing
- Infrastructure Consideration
- Mitigation or Risks Plan
- Resourcing
- Deliverables and Milestones

Open Source Testing Tools for Agile Testers

- A list of useful tools to help your testing activities in agile projects.

1. Selenium Web Driver

- Selenium Web Driver is the most widely used test tool for browser test automation.
- By running automated tests on the UI, we can simulate what the user sees when they interact with the web application.
- One of the main advantages of using Selenium WebDriver over other UI automation tools is that you can write your automated tests in a variety of supported programming languages, such as Java, C#, Ruby, Python, and PHP.
- Selenium is often used in Agile projects because there is a heavy emphasis on automated testing and that the whole team can contribute to automated tests.

2. Jmeter

- Jmeter is an open source performance testing tool written entirely in Java. It can be used to load test websites (HTTP, HTTPS) as well as Web Services(SOAP and REST) and Databases.
- The HTTP(S) Test Script Recorder can be used to record and replay requests.
- Jmeter can be extended by plugins to support further functionalities and there is also a Jenkins plugin which means you can run performance tests as part of the delivery pipeline.

3. SoapUI

- SoapUI tool is primarily used for functional testing of Web Services.
- The tool is relatively easy to use and you can create tests for web services in just a few minutes.
- You can execute tests and analyze the reports all inside SoapUI GUI.
- A nice feature of SoapUI is its ability to create mock web services which are handy when you are creating tests for a web service which are not yet developed.

4.VirtualBox

- These days, most applications need to be tested against multiple browsers and operating systems.
- Instead of having physical servers with different operating systems and browsers, VirtualBox provides an easy solution to create virtual machines with different configurations.
- You can run VirtualBox on any operating system to create virtual machines and you can even have multiple virtual machines with different operating systems on the same box.

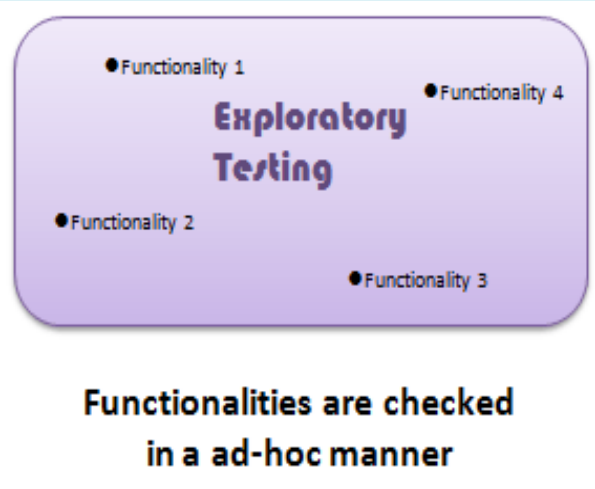
5.XMind

- XMind in the most popular and feature-rich mind mapping tool. You can use it to create stunning mind maps to highlight the features of a website, or to create a user journey flow through the application.
- Mind maps have become very popular in the world of software testing because they provide nice visuals and can be of great help when designing test cases.

Exploratory Testing in Agile

- Exploratory Testing is an important activity in an agile environment as it can help software testers to keep up with the rapid development pace of agile software projects.
- In agile methodology, software is released in small iterations.
- Each iteration goes through planning, estimation, development, integration, testing and release.
- Because of frequent releases, test automation becomes ever so important as developers need to get quick feedback on the status of the application.
- The automated checks serve as regression tests to ensure that with each release the software has not regressed.

- **Exploratory Testing** is defined as simultaneous learning, test design and test execution. It is an approach to testing that values the tester as an integral part of the test process and shares the same values as the Agile Manifesto:
  - ✓ *Individuals and interactions* over processes and tools
  - ✓ *Working software* over comprehensive documentation
  - ✓ *Customer collaboration* over contract negotiation
  - ✓ *Responding to change* over following a plan
- Exploratory Testing is also complementary to test automation; that is while automated checks are checking for regression issues, Exploratory Testing focuses on new features which have been developed.
- This is important because each sprint typically lasts only couple of weeks, which doesn't allow sufficient time for scripting tests cases and executing them later against the application.
- On the other hand, exploratory testing in agile environment allows testers to get familiar with the domain and the application and on each iteration, that understanding is enhanced and hence testers become more efficient.



Exploratory Testing

- Directed from requirements and exploring during testing
- Determination of test cases during testing
- Investigation of system or application
- Emphasizes adaptability and learning
- Involves Investigation
- Is about Improvement of test design
- Like making a conversation – it’s spontaneous

Exploratory testing

- Is not random testing but it is ad-hoc testing with a purpose of find bugs
- Is structured and rigorous
- Is cognitively (thinking) structured as compared to the procedural structure of scripted testing. This structure comes from Charter, time boxing etc.
- Is highly teachable and manageable
- It is not a technique but it is an approach. What actions you perform next is governed by what you are doing currently