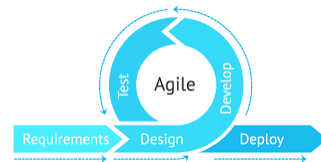


Software Engineering And Testing

Agile Testing



Outline

- Agile Testing – Principles, Methods & Advantages
- Agile Testing Quadrants: the Purpose of Testing
- Overview of Agile Testing Methods:
 - Acceptance Test Driven
 - Behaviour Driven Development.

What is Agile Testing?

- **Agile Testing** is a testing practice that follows the rules and principles of agile software development.
- Unlike the Waterfall method, Agile Testing can begin at the start of the project with continuous integration between development and testing.
- Agile Testing methodology is not sequential (in the sense it's executed only after coding phase) but continuous.

3

Principles of Agile Testing

1. **Provide continuous feedback.** Agile testers do not simply test constantly. They are also responsible for distributing the results of those tests and facilitating the provision of feedback from customers to developers in order to create a more robust product.
2. **Deliver value to the customer.** While this is the second principle, it is the paramount principle. The end goal of every action taken by an Agile tester is to create the best product possible for the customer.
3. **Enable face-to-face communication.** The Agile tester's role is to reduce confusion and errors by communicating directly with developers, as well as enabling customers to communicate directly with developers.

4

Principles of Agile Testing

4. **Have courage.** Developers can be defensive about their work. To meet their goal of providing value to the customer, Agile testers must fight for the changes and fixes that need to be made.
5. **Keep it simple.** Agile testers act on the aphorism that simplicity is the ultimate sophistication. For testing, that means performing *only* those tests that are necessary, and *all* tests that are necessary. For the product, that means delivering the simplest possible product that delivers the most possible value.
6. **Practice continuous improvement.** Agile testers are keen learners; they are never done learning how to do their job better.
7. **Respond to change.** Agile testers are adaptable and flexible, keeping up feedback from user stories, as well as changes in the product and the marketplace.

5

Principles of Agile Testing

8. **Self-organise.** Instead of waiting at an assigned seat on the assembly line, Agile testers spring into action at every point in the process. They actively seek out problems and bring people together to solve them.
9. **Focus on people.** Agile testers are collaborative, preferring human interaction to technology. Their focus on people enables them to deliver a product that prioritises usability and utility.
10. **Enjoy.** No one is as successful at meeting their goal as when they enjoy what they are doing. Agile testers who enjoy the work are able to deliver the greatest possible value to the customer.

6

Advantages of Agile Testing?

- **A higher-quality product** ☞ - Agile enables testers to detect *more* defects earlier in the development process.
 - One of the Agile principles is '**continuous feedback**.' The doctrine of starting testing concurrently with development means that bugs can be eliminated soon after they are created.
 - Each iteration of the product is thoroughly tested and debugged as it's created, rather than waiting until it's finished.
 - Testing also involves *every* member of the development team, so the skills of both developers and testers are leveraged in the pursuit of a perfect product.
 - Another outcome of continuous feedback combined with **early** and **frequent** testing is testers developing an intricate knowledge of the product.

7

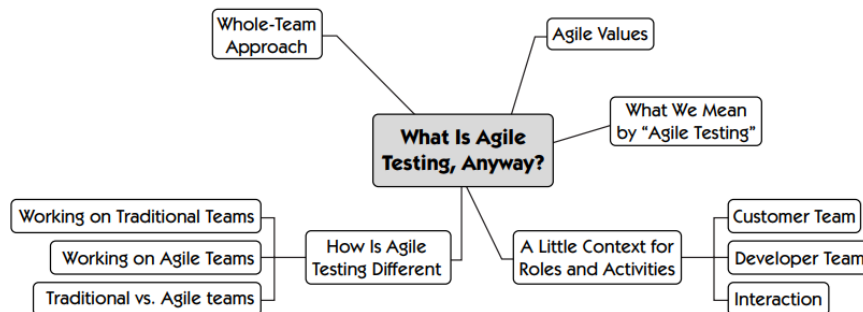
Advantages of Agile Testing?

- Depending on the methodology of testing used, they can combine that knowledge with customer input to help developers create a superior product.
- **Fast delivery** ☞ - With waterfall testing, the initial stages of development and eventual release onto the market are separated by months, if not years. As a result, features or the entire product can be completely irrelevant by the time it reaches customers.
 - Agile testing methodology both compresses the development cycle and constantly provides customer feedback, ensuring the product adapts to the market during development and reaches customers as soon as possible.

8

Advantages of Agile Testing?

- **A happier team** □ - The last principle on the Agile testing list is no mistake: enjoyment. Agile testing necessitates close interaction between all members of a team, creating a happier, more enjoyable, and more productive workplace. Developers, testers and customers work side by side to create the best product and the most value possible.

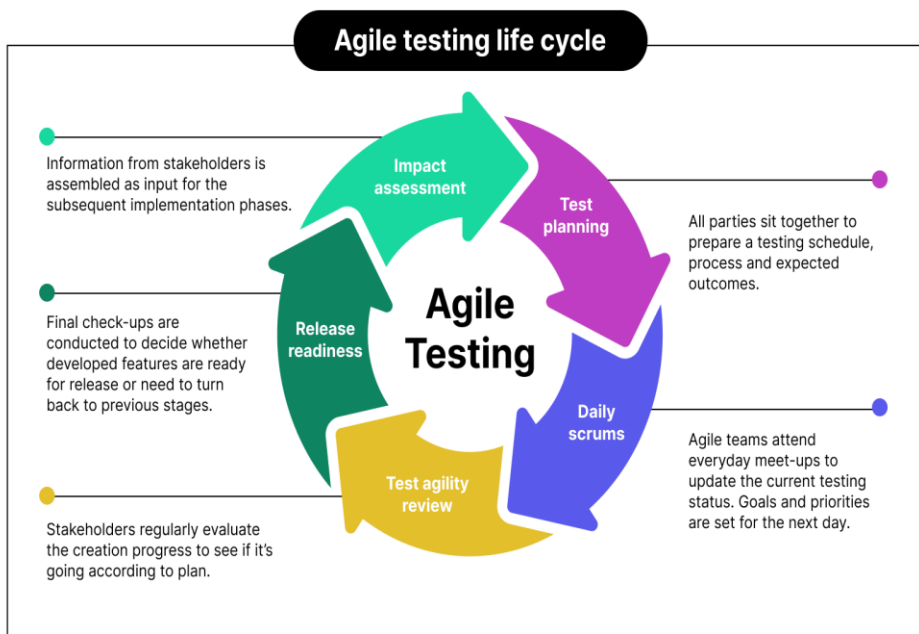


9

Interaction between Customer and Developer Teams

- Ideally, they're just one team with a common goal to deliver value to the organization.
- Agile projects progress in iterations, which are small development cycles that typically last from one to four weeks.
- The customer team, with input from the developers, will prioritize stories to be developed, and the developer team will determine how much work they can take on. They'll work together to define requirements with tests and examples, and write the code that makes the tests pass.
- Testers have a foot in each world, understanding the customer viewpoint as well as the complexities of the technical implementation
- Some agile teams may not have any members who define themselves as "testers."

10



Agile process testing steps

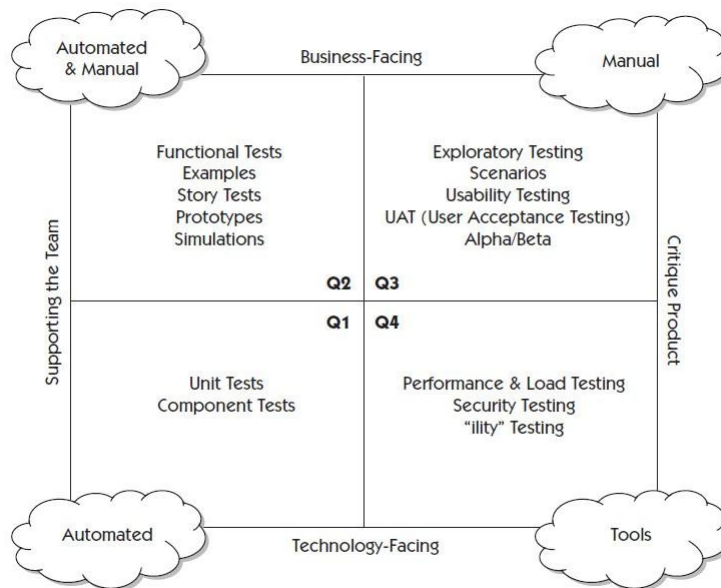
- **Phase1: Impact Assessment:** In this initial phase, we gather inputs from stakeholders and users. This phase is also called the feedback phase, as it assists the test engineers in setting the objectives for the next life cycle.
- **Phase 2: Agile Testing Planning:** It is the second phase of the Agile testing life cycle, where all stakeholders come together to plan the schedule of the testing process and deliverables.
- **Phase 3: Release Readiness:** At this stage, we review the features that have been developed/ Implemented are ready to go live or not. In this stage, it is also decided which one needs to go back to the previous development phase.
- **Phase 4: Daily Scrums:** This stage includes every standup morning meeting to catch up on the status of testing and set the goal for the entire day.

Agile process testing steps

- **Phase 5: Test Agility Review:** The last phase of the Agile life cycle is the Agility Review Meeting. It involves weekly meetings with stakeholders to regularly evaluate and assess progress against goals.

13

Agile testing Quadrants



14

Agile testing Quadrants

○ Agile Quadrant I

- The internal code quality is the main focus in this quadrant, and it consists of test cases which are technology driven and are implemented to support the team, it includes
 - Unit Tests
 - Component Tests

○ Agile Quadrant II

- It contains test cases that are business driven and are implemented to support the team. This Quadrant focuses on the requirements. The kind of test performed in this phase is
 - Testing of examples of possible scenarios and workflows
 - Testing of User experience such as prototypes
 - Pair testing

15

Agile testing Quadrants

○ Agile Quadrant III

- This quadrant provides feedback to quadrants one and two. The test cases can be used as the basis to perform automation testing. In this quadrant, many rounds of iteration reviews are carried out which builds confidence in the product. The kind of testing done in this quadrant is
 - Usability Testing
 - Exploratory Testing
 - Pair testing with customers
 - Collaborative testing
 - User acceptance testing

16

Agile testing Quadrants

- **Agile Quadrant IV**
 - This quadrant concentrates on the non-functional requirements such as performance, security, stability, etc. With the help of this quadrant, the application is made to deliver the non-functional qualities and expected value.
 - Non-functional tests such as stress and performance testing
 - Security testing with respect to **authentication** and hacking
 - Infrastructure testing
 - Data migration testing
 - Scalability testing
 - Load testing

17

Agile testing Methodologies

- Four basic agile methods are being used by developers:
 - Behaviour Driven Development (BDD)
 - Acceptance Test-Driven Development (ATDD)
 - Exploratory Testing
 - Session-Based Testing

18

Test Driven Development (TDD)

- In the Test Driven Development (TDD) method, the code is developed based on the Testfirst approach directed by Automated Test Cases.
- A test case is written first to fail, code is developed based on that to ensure that the test passes.
- Method is repeated, refactoring is done through the development of code.
- TDD can be understood with the help of the following steps
 1. Write a Test case to reflect the expected behavior of the functionality of the code that needs to be written.
 2. Run the test. The test fails as the code is still not developed.
 3. Develop code based on the test case.
 4. Run the test again. This time, the test has to pass as the functionality is coded. Repeat Step (3) and Step (4) till the test passes.

19

Test Driven Development (TDD)

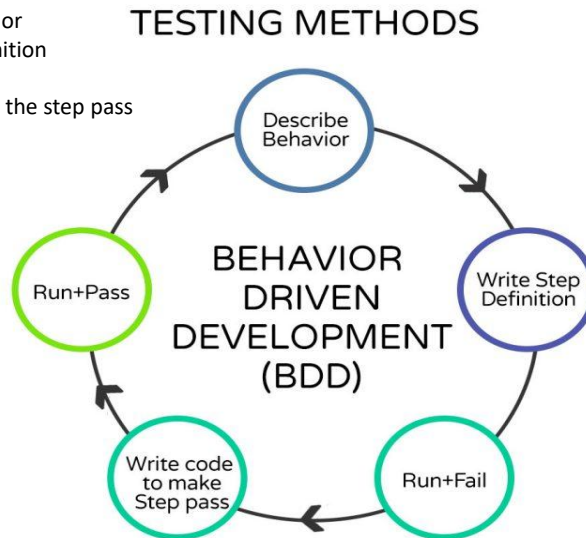
5. Refactor the code.
 6. Run the test again to ensure it passes.
- Repeat Step 1 – Step 6 adding test cases to add functionality.
 - The added tests and the earlier tests are run every time to ensure the code is running as expected. To make this process fast, tests are automated.
 - The tests can be at unit, integration or system level. Constant communication between testers and developers needs to be ensured.

20

Behaviour Driven Development (BDD)

BDD has [five steps](#):

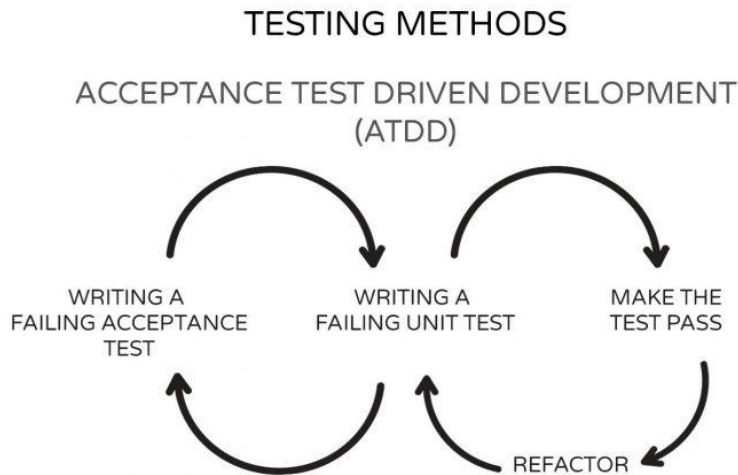
1. **Describe** the behavior
2. **Write** the step definition
3. **Run** and fail
4. **Write** code to make the step pass
5. **Run** and pass



Behaviour Driven Development (BDD)

- It encourages communication between the stakeholder involved in a project so every other member understands every feature of the system before the beginning of the development process.
- In BDD, tester developers and analysts create scenarios that promote example focused communication in detail.
- The main idea of BDD is that a specific team creates relevant scenarios and then builds a test series around those scenarios that are predicted to fail.
- Next, these testers build software functionality and program them to pass in the same scenarios.

Acceptance Test-Driven Development (ATDD)



Acceptance Test-Driven Development (ATDD)

- Acceptance Test-Driven Development or ATDD involves three **players, customer, developer, and tester.**
- All these three players come together in meetings to gather inputs from their roles and use them to define acceptance tests.
- The customer focuses on the problem, the developer pays attention to proposing a possible solution to the problem and testing analysis on possible situations where things could go wrong.
- It basically represents a user's perspective and represents the possible ways of how the system can function.
- It also ensures that the system functions as planned.
- These tests are mostly automated and their script is closely similar to the BDD approach.



25