

What is Agile Testing? Process, Methodology and Strategies

testbytes.net/blog/agile-testing

December 2,
2019

testbytes
Making Quality a Habit



Process

Monday December 2, 2019

The agile Testing methodology has gained a lot of importance and popularity nowadays when it comes to software testing modern-day software. It is an intelligent way of testing complex software that bears more efficient end results than traditional testing methods.

This is a complete guide to Agile Testing, explaining all the important aspects related to it.

1. What is Agile Testing?

2. Agile Testing Methodology

3. Agile Testing Life-Cycle

4. Agile Test Plan

5. Agile Testing Strategies

6. Agile Testing Quadrants

7. Difference Between Agile Testing Vs Waterfall Testing

8. Advantages of Agile Testing

What is Agile Testing?

Agile testing is an iterative and incremental testing process in which software requirements are gathered gradually throughout the testing process.

Agile Testing is a part of the Agile Software development process which is carried out with the help of customer's involvement. The agile testing process starts at the beginning of the software development process and there is no integration required between the development and testing processes.

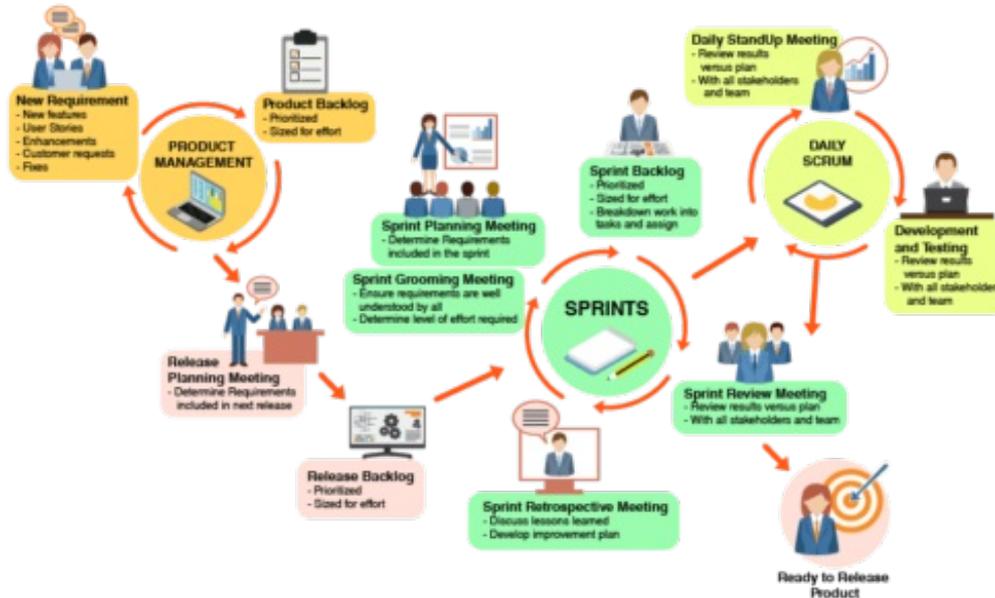


As the development and testing are integrated into this process, it allows early error detection and removal which results in less development time as well as costs. This also ensures high product quality as errors get removed in the initial stage itself.

The basic principles on which Agile Testing is conducted are:

- Agile testing is a continuous process and it ensures smooth project progress.
- Product and business requirements are realized through continuous feedback. Feedback's are provided on an on-going basis.
- In Agile Testing, the development team, testing team, and the customers, all are involved in the testing process.
- There is very little documentation required in Agile Testing as all the testers use a reusable checklist instead of lengthy documents. The main focus is on the test in this process.
- The errors are resolved by the testing team as and when they occur, in the same iteration. This results in simpler and clean code.
- Traditional testing is performed after the software is developed. However, agile testing is performed along with the development process which helps reducing development time.

Agile Testing Methodology



TDD (Test Driven Development)

As the name suggests, the test-driven development approach starts with the test itself. It depends upon the repetition of the development cycle.

The first step in this development cycle is to create a unit test case. The next step involves creating the code which fits the test case so that the test case can be run. Thus, the entire code is created until the unit test passes.

This test is usually performed with the help of automated testing tools and run on units and components of the code.

ATDD (Acceptance Test Driven Development)

In this testing methodology, the developers, testers and the customers together build an acceptance test from the point of view of the customers in order to understand the requirements of the customers.

The code is developed according to the acceptance test case created.

This testing method is very customer-centered, and the main aim is to get to develop a program according to the user's perception.

BDD (Behaviour Driven Development)

BDD is built on the same principles as TDD and ATDD. Thus, the code is developed according to the test case created in this testing methodology too.

The main aim of this development is to focus on the identification of business needs and outcomes. The development should be related to a business outcome.

The steps that are followed in BDD are:

- First, describe the behavior
- Creating the test case
- Writing code according to the test case defined
- Continuing the process until the code passes the test case.

Exploratory Testing

In this software testing methodologies, the testers have complete freedom to explore with the code and develop the most efficient software.

The testers create multiple test cases in order to explore each aspect of the software functionality. Testers perform different tests and record the process to study it and figure out the exact flow of the software.

This type of testing is a very important part of the agile test as it helps uncover hidden risks from the software which could not have been detected by simple testing methods.

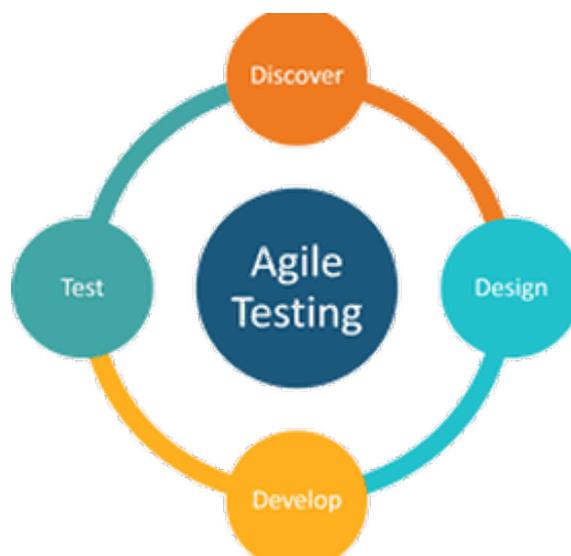
Session-Based Testing

Session-based testing is built on the principles of exploratory testing. However, unlike exploratory testing which is conducted spontaneously without any planning, there is some structure in session-based testing.

This structure is provided by conducting tests during uninterrupted sessions where testers are required to report the tests that took place during the process.

The session-based testing methodology also helps in finding hidden bugs and defects in the software.

Agile Testing Life-cycle



There are 5 phases in the Agile Testing Life-cycle:

Agile Test Planning

The first phase is creating Agile Test Plans. In this phase, all the stakeholders: developers, testers, customers and users, come together and plan testing schedules.

Daily Scrums

The goals and targets of the day are set at the start of each day. It helps testers realize the status of the testing.

Test Agility Review

Agility reviews are performed frequently in the development process. These are done to analyze the progress of the development.

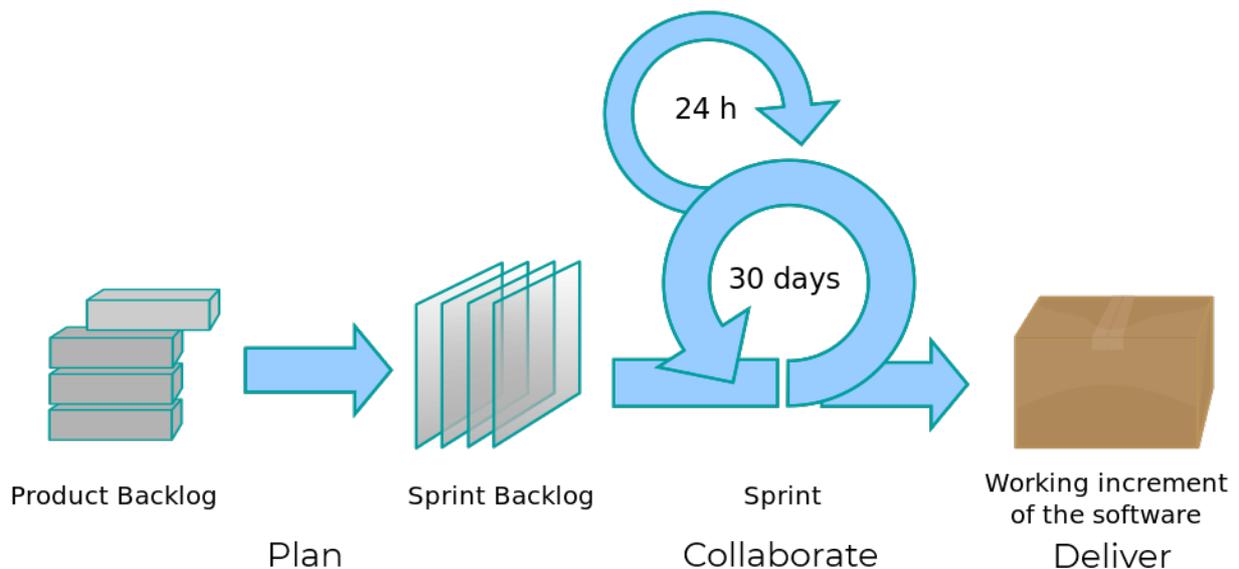
Release Readiness

In this phase, testers decide which assess which features are completely developed and ready to go live and which ones need to go back to the previous development phase.

Impact Assessment

This phase is the feedback phase of the life cycle which helps testers set goals for the next life-cycle. The inputs and feedback are collected from users and stakeholders to perform an impact assessment.

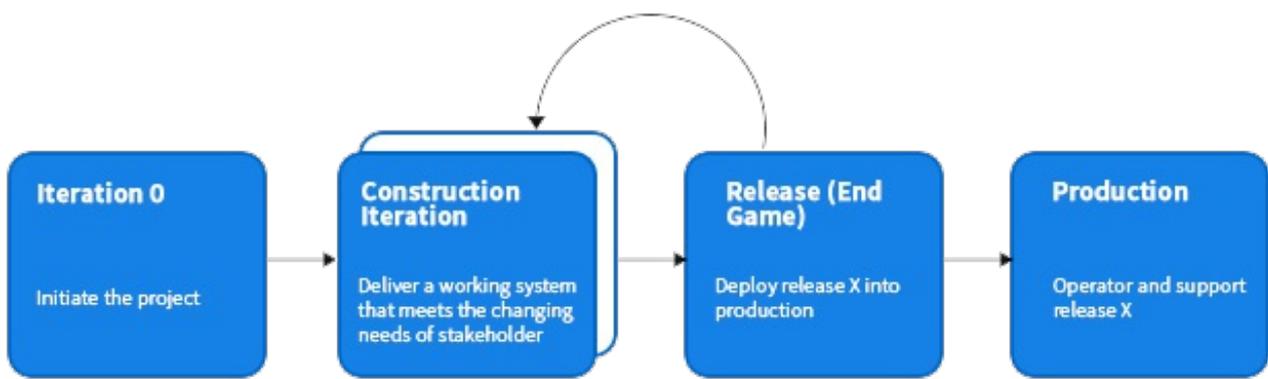
Agile Test Plan



The Agile Test plan includes all the types of testing performed in a particular iteration. It includes the following:

- It defines the test scope, sprint goals, test and extent to which the test is to be performed
- It specifies the testing tools to be used, data and configurations for the test and the environment in which the test will be performed.
- It schedules the test tasks and frequency of tests, i.e. how many times will they be performed.
- It defines the testing methods and techniques
- It also determines the expertise and training required to carry out tests.
- It sets the priority of the tests according to the customer's point of view.

Agile Testing Strategies



iteration 0

Process involved

- Initial stages of testing such as tester selection, selection of testing tools, resource allocation, testing method selection, etc.
- Risk identification
- Cost estimation
- Boundary condition establishment and scope defining

Construction iteration

This process can be divided into two confirmatory testing and investigative testing. confirmatory testing is performed to make sure that the product satisfies all the requirements of the stakeholders.

Investigative testing is performed to test deep and find all the issues that are ignored in confirmatory testing.

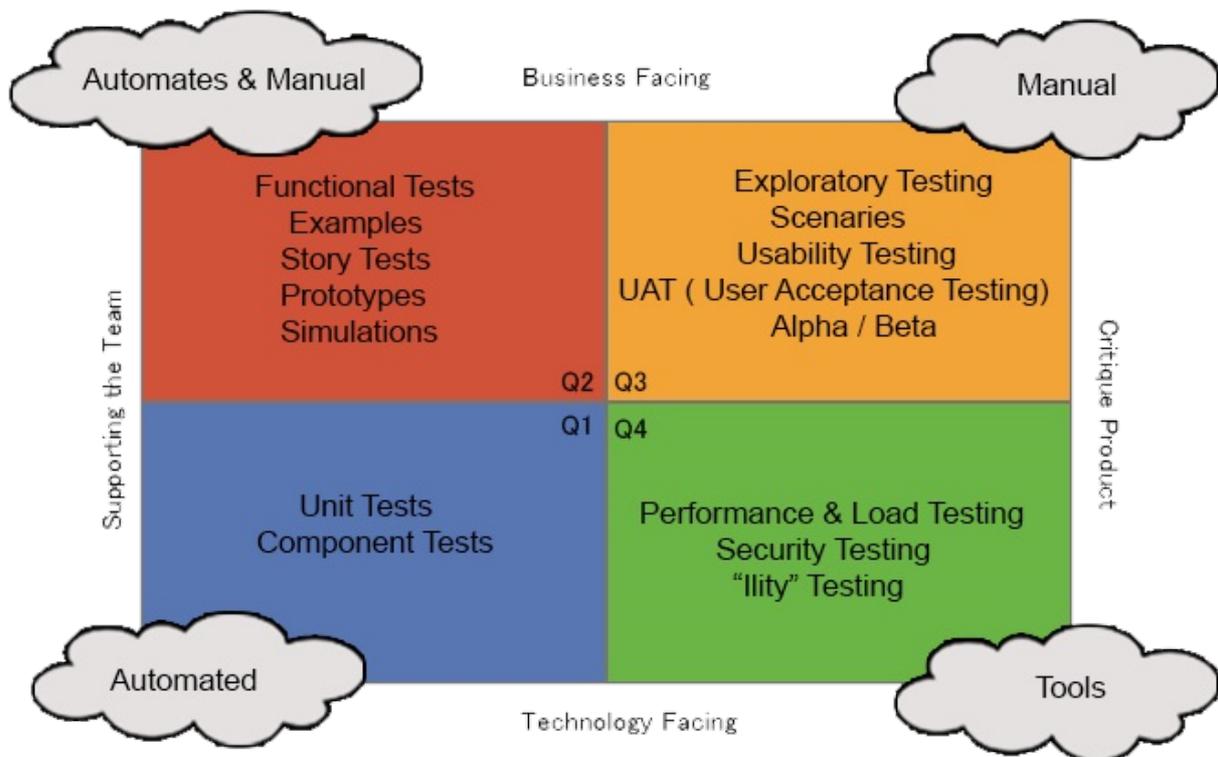
confirmatory testing can be again divided into agile acceptance testing and developer testing. Both the tests can be automated to make sure that continuous regression testing happens

Agile acceptance testing is a combination of functional testing as well as acceptance testing. on the other hand developer testing is a mix of unit testing and integration testing

Production

Once the release stage is completed the product will move on to the production stage

Agile Testing Quadrants



The Agile testing quadrant divides the entire testing process into 4 parts. This makes the Agile testing process easy to understand.

Among the 4 quadrants, the left 2 tell the testers which code to write and the right 2 quadrants help them understand the code better with the help of feedback to the left quadrants.

Quadrant 1

This quadrant focuses on the quality of the code. It includes test cases and test components that are implemented by the testers.

These test cases are for automation testing to help to improve the code.

Quadrant 2

This quadrant contains business-driven test cases which are also implemented by the testing team.

The main focus of this quadrant is on customer requirements. It improves the business outcomes of the software being created.

It contains the following tests:

1. Testing scenarios which may occur and workflow
2. Testing the user experience
3. Performing pair testing

Quadrant 3

This phase provides feedback for the previous two phases.

There are many iterations of reviews and feedbacks carried out in this quadrant which helps to strengthen the code.

Usability tests, exploratory tests, user acceptance tests, and collaborative tests are performed in this quadrant.

Quadrant 4

The non-functional requirements of the code, such as performance, security, scalability, etc. are taken care of in this quadrant.

Testing for stress and performance is carried out in this phase. Security and infrastructure test, data migration and load testing.

This quadrant makes sure that the code satisfies all the non-functional requirements.

Difference Between Agile Testing Vs Waterfall Testing

Agile Testing	Waterfall Testing
Incremental approach	Sequential process
Project will be divided into sprints	Project will be divided into sequential process
Meant for small projects	Meant for all sort of projects
Starts at the beginning of the product and changes in the middle is not an issue	Product can be only tested at the end of the project

unstructured	More secured
Less priority for documents	High priority
Testers and developers can work together	Both testers and developers has to work separately

Advantages of Agile Testing

- saves time and money. Because the testing is carried along with the development. The errors and bugs are fixed at the same time and no additional time s required to test the entire software, find errors and solve them.
- There is little to no documentation required in Agile testing as testers use reusable checklists.
- Feedback is also a part of the Agile testing life-cycle and users give feedback after each process iteration.

Conclusion

Agile Testing is the best testing methodology for the modern-day software which is highly complicated and requires in-depth testing.

This testing methodology is very customer-centric. It requires high customer involvement in order to deliver a better quality of software which meets customer and user expectations.



This testing demands a high level of communication between developers, testers as well as customers and users as they all work together in testing the software.

Agile testing allows testers to be flexible and incorporating any requirement changes is very easy in Agile development.

This testing is becoming increasingly popular among major software development companies. It is a smart way of testing and the end result is of better quality too.