**Agile Testing**

1. Difference between sprint review and sprint retrospective

Sprint Review focuses on the product, while Sprint Retrospective focuses on the process. Sprint Review is concerned primarily with optimizing and maximizing product value, whereas Sprint Retrospective is involved with people, processes, and tools.

1. Product backlog, Release backlog, Sprint backlog:

Product backlog: Features you want to implement but have not yet prioritized for release.

Release backlog: Features that need to be implemented for a particular release.

Sprint backlog: User stories that need to be completed during a specific period of time.

1. Burn down and burn up chart:

Burn-down and burn-up charts are two types of charts that project managers use to track and communicate the progress of their projects.

A burn down chart shows how much work is remaining to be done in the project,

whereas a burn-up shows how much work has been completed, and the total amount of work.

1. Agile Manifesto A statement on the values that underpin Agile software development.

The values are: individuals and interactions over processes and tools, working software

over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan.

1. what is a test closure report in software testing

Test closure is a vital step in software testing that marks the official end of testing activities for a software release. These reports summarize the entire testing process and its outcomes for the projects. It provides valuable insights into how well the software performed during the tests.

1. The Agile work structure.

**Themes**

A theme is a wide area of focus that helps an Agile team to keep track of their organizational goals — think of it as a label that can be used to group similar activities. A theme helps to define the common characteristics between different areas and unite them under one heading.

**Epics**

An epic is a substantial collection of smaller stories that combine to make one large story. An epic cannot be completed in a single [Agile iteration](https://www.wrike.com/agile-guide/faq/what-is-agile-iteration/) (or [sprint](https://www.wrike.com/project-management-guide/faq/what-is-a-sprint-in-agile/)). The key element to an epic is that it takes a lot of time. For more information on Agile epics, check out our [detailed guide](https://www.wrike.com/agile-guide/agile-epics-guide/).

**User Stories**

A story, also referred to as a user story, is a short-form request that can be delivered in one sprint. It is written in simple language from the perspective of the user. [Story points](https://www.wrike.com/agile-guide/story-points-estimation/) are used to measure the complexity of a story. The overall goal of a story is to provide value to its user within a set timeframe. Learn how to create a user story [here](https://www.wrike.com/agile-guide/user-stories-guide/).

**Tasks**

A task is a subsection of a story. It helps to break the story down and outline how it will be completed. Tasks tend to be more technical as they are used by members of the development team (e.g., a quality assurance tester) rather than a front-end user.

**Manual Testing**

1. **Test Cases:**

Test case gives detailed information about the testing strategy, testing process, preconditions, and expected output.

1. **What is testing?**

Verify the application functionality to deliver quality products to the customer without having any issues/bugs.

1. **How you will confirm application meets all requirements**

If all the test cases are passed and no open bugs are there then we can confirm application meets all requirements

1. **When you stop testing?**

As per traceability, if all test cases are passed and there are no open issues are there then we will stop our testing.

1. **Difference between a project and a product?**

The project is for single customer requirements whereas the product is for market industry requirements.

1. **What is Smoke Testing when we will do it?**

Before accepting the sprint, verify the basic functionalities are working or not to accept it for major testing. It should meet the acceptance criteria 70 to 80% of functionalities should work then we will accept it for major testing or else the test lead will reject the sprint.

**OR**

Smoke Testing is a software testing process that determines whether the deployed software build is stable or not. Smoke testing is a confirmation for QA team to proceed with further software testing. It consists of a minimal set of tests run on each build to test software functionalities. Smoke testing is also known as “Build Verification Testing” or “Confidence Testing.”

In simple terms, smoke tests mean verifying the important features are working and there are no showstoppers in the build that is under testing. It is a mini and rapid regression test of major functionality. It is a simple test that shows the product is ready for testing.

1. Difference between GUI and Functional testing

GUI testing will verify all the elements on the page whether properly displaying or not.

Functional testing is to verify the application behavior and whether it is working according to the customer's requirements.

1. What is regression testing?
2. How will you identify related functionalities in regression testing?

By using decision table, we will identify what are the functionalities are interrelated. When bug is fixed, we will refer decision table and execute the test cases accordingly in the part of regression testing.

1. What will verify in GUI testing?

As per customer requirement, all the elements are displaying properly or not will be verify. Elements like buttons, textbox etc.

1. What is Adhoc testing? (No plan Testing)

When there is not enough time, we will test important functionalities by selecting randomly and verify the application is working or not according to the customer's requirements. The testing team will give conditional signoff for the production release.

When we have more time, the tester will do random testing on functionality and intend to break the application functionality.

1. What is the use of the test plan and the different sections in it?

A test plan is used for scheduling the testing activities from beginning to end with clear dates with the scope of testing. And also having resource requirements, software, hardware, defect management, traceability matrix, risks, and contingency plan.

1. What are the risk and contingency plans?

Risks are unexpected problems getting into the project and a contingency plan is the approach/plan to resolve the issue.

1. What are different types of applications we can test?

Windows, client-server, web, distributed mobile application.

1. What is bug life cycle?

Different phases of bug right from starting to ending. New, Reject, Differed, Open, Fixed, Reopen, Closed.

1. What is the testing life cycle?

Following is the process right from beginning to ending,

1. Test Plan
2. Test Development
3. Test Execution
4. Result Analysis
5. Bug tracking
6. Reporting
7. What is a bug and defect, remark?

The bug is raised by the testing team and when it is accepted by the dev team then calls it is defect.

Remark is a bug rejected by the dev team.

### ****What is a Bug?****

[A bug in software testing](https://qacraft.com/what-is-bug-in-software-testing/) is a divergence from the customer's requirements, or, to put it another way, a discrepancy between the expected result and the actual result in an application or a module that is discovered by the testing team throughout the testing phase.

### ****What is a Defect?****

Defects, which are discovered during the development phase of an application's unit testing, occur when the functionality of the program does not function as the client requires. Like A flaw could be caused by entering incorrect data or by any programming error.

### ****What is an Error?****

A software engineer's slip-up, misunderstanding, or mistake is referred to as an error in software testing. Software engineers, analysts, programmers, and testers are all considered developers. For instance, a programmer can type a variable name improperly or a developer might misunderstand a design notation, both of which could result in an error. It is produced as a result of a bad login, loop, or syntax.

### ****What is a Failure?****

If the customer discovers any problems with the software after it has been completed and delivered to them, that indicates that the software has failed.

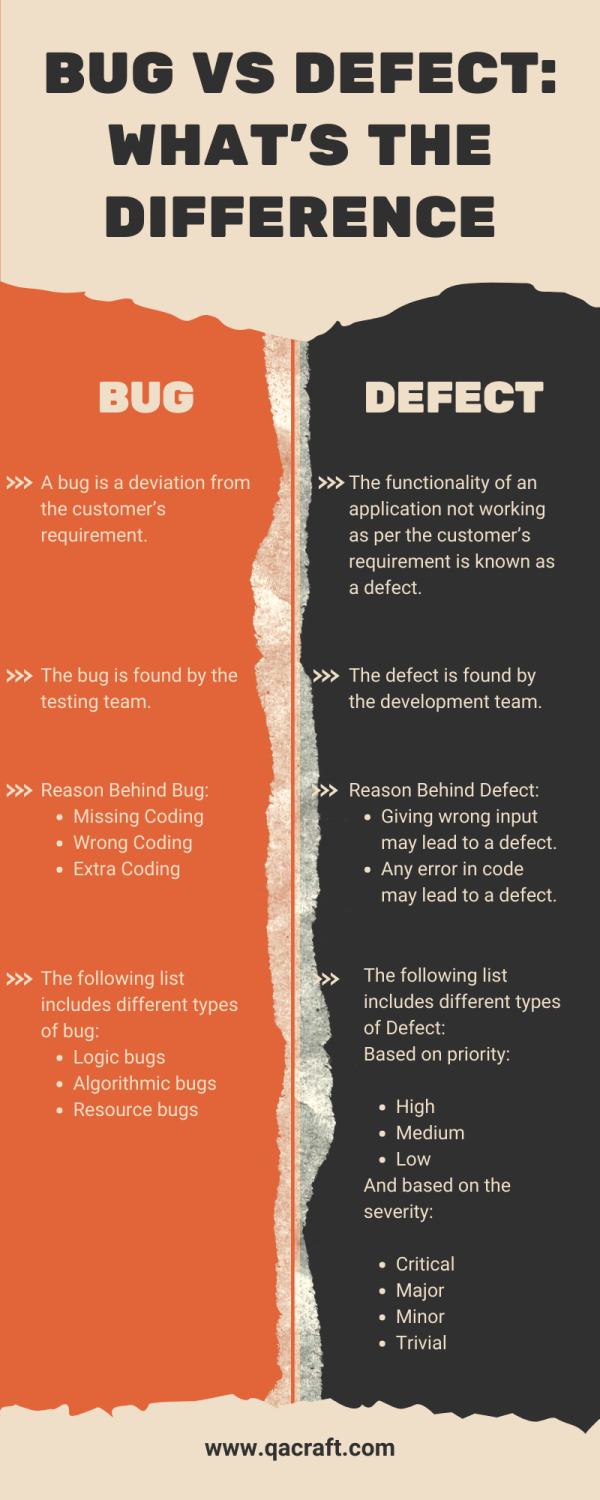
To put it another way, a failure occurs when a user discovers a problem with the software.

**Sum up:**

A simple diagram makes you understand.



We can state that a mistake committed by a programmer during coding is considered an error, a defect discovered during unit testing during the development phase is called a bug, and when an error is discovered at the end user's end is called a failure.



1. **What is deferred bug?**

Due to lack of time, any defect is not fixing in current release and will be fixing in future then dev team will give the status as deferred.

1. **When we need to go for agile model?**

For unstable requirements projects, other models are not suitable because when there is any requirement change then again need to start from the beginning. Whereas in Agile, always we can accept the changes implement them in less time, and deliver them to the client.

1. **What are the drawbacks of the waterfall model?**
2. No verification technique.
3. When there is any change in requirements, again need to start from the beginning.
4. There is no bottom to up.
5. If requirements are unstable, time and budget will be very high.
6. **What are the different section we will fill while raising the bug?**
7. Summary
8. Description
9. Steps to reproduce.
10. Project
11. Module
12. Build No
13. Severity
14. Priority
15. Founded by
16. Assign to
17. Comments
18. Status
19. **What are the different types of reports send from testing team?**

Daily, weekly, monthly, test, summary report, test closure report, bug report

1. **What is severity and priority?**

Severity is talking about the seriousness of the issue and given by the testing team.

Priority is talking about how urgently to fix the issue and given by the developer.

1. **What is the difference between validation and verification?**

Quality assurance techniques verification and here we will verify all our documentation is according to the requirement or not.

Quality Control technique is validation and here we will perform test executions and verify whether working according to the requirement or not.

1. **What is high priority and low severity bug?**

Page title is spell mistake as it is low severity to the tester but high priority to the developer.

1. **What is low priority and high severity bug?**

Some functionality is not working as tester will give high severity and if it is not impacting any other functionalities then it is low priority

Eg Trash in gmail

1. **How many types of test cases you write.**
2. GUI
3. Functional +ve
4. Functional -ve
5. Field level validation