

Non-Functional Testing

01

- **Load Testing:** A load test is a type of performance test that checks how systems function under a heavy number of concurrent virtual users performing transactions over a certain period of time.

02

- **Performance Testing:** Performance testing is the general name for tests that check how the system behaves and performs.

03

- **Volume Testing:** Volume testing is a software testing performed to test the system under huge data load.

04

- **Stress Testing:** A stress test is a type of performance test that checks the upper limits of your system by testing it under extreme loads.

05

- **Reliability Testing:** Reliability Testing is an important software testing technique that is performed by the team to ensure that the software is performing and functioning consistently in each environmental condition as well as in a specified period.



It is a number given to Installable software that is handed over to the customer by the tester or developer.

Build



It is a number given to Installable software that is given to the testing team by the development team.

Release



Bug release is when software or an application is handed over to the testing team knowing that the defect is present in a release. During this the priority and severity of bug is low, as bug can be removed before the final handover.

Bug Leakage



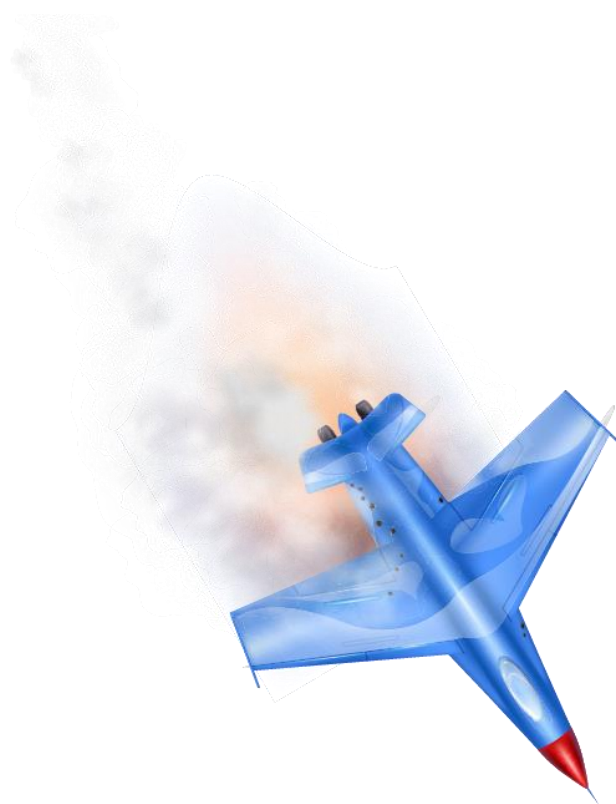
Bug leakage is something, when the bug is discovered by the end users or customer, and not detected by the testing team while testing the software.

Bug Release

Coding Analysis

```
void funtionWeTest(Parameter){  
    Int p = price(para1);  
}
```

```
void Price(para1){  
    return 10;  
}
```



STUBS

Top-Down

Is called by the Module under Test

- A **stub** is a replica of a module that collects the data and develops many possible data. However, it executes like an actual module and is mainly used to test modules.
- Generally, stubs are created by software developers in order to use them instead of modules if the particular modules are miss or not yet developed.
- By using these test stubs, the test engineers can simulate the performance of the lower-level modules, which are not yet joined with the software. Furthermore, it helps us to accelerate the activity of the missing modules.

Analytical Analysis

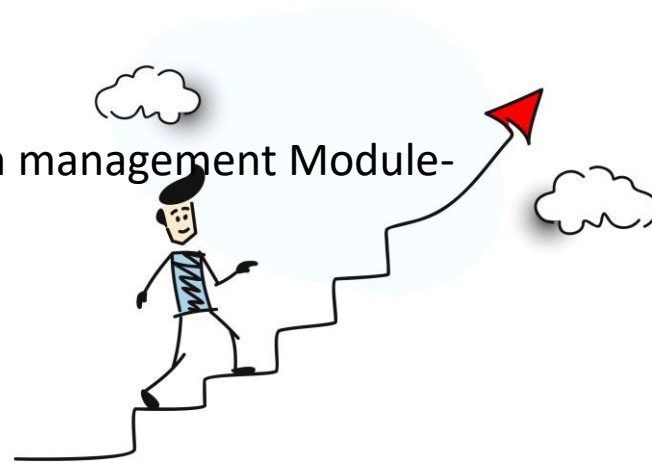
Let's consider a student information management Module-

A: Login Page

Module-B: Home Page

Module-C: Print Setup Page

Module-D: Logout Page



All these module are interdependent on each other:

- For example, to reach home page(Module-B), you need to first validate the login page(Module-A).
- Similarly, to test Print Setup Page(Module-C), you need to execute or access and validate Home Page(Module-B)
- Further, in order to test logout page(Module-D) you first need to execute Login page(Module-A), thereafter which you can test the logout page.

DRIVERS

Down-Top

Calls the Module to be tested

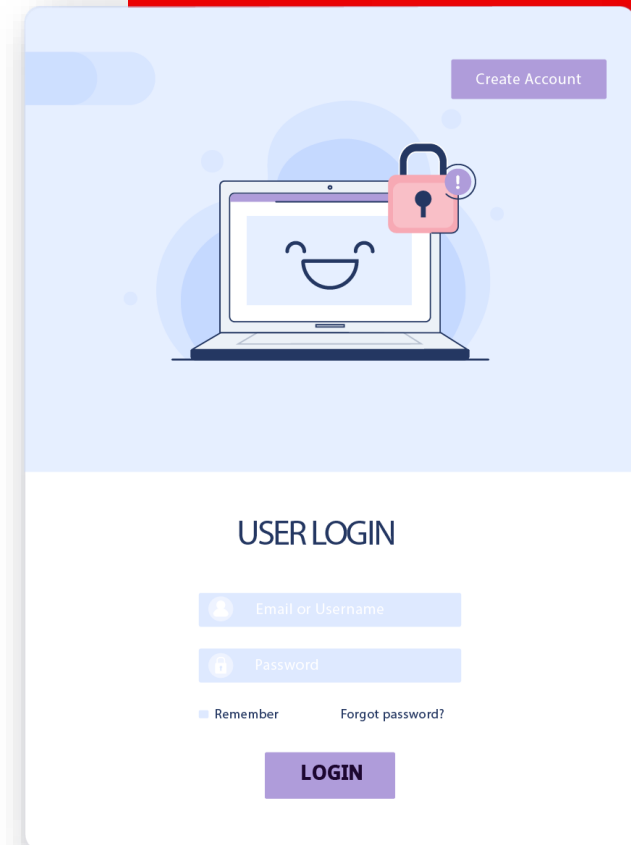
- The **Drivers** establish the test environments and takes care of the communication, estimates results, and also sends the reports.
- These are just like stubs and used by software test engineers in order to accomplish the missing or incomplete modules/ components requirements.
- The drivers are mainly developed in the **Bottom-up approach of incremental integration testing**.

Decision Table Testing

Decision Table Testing is a software testing methodology used to test system behavior for various **input combinations**.

Conditions	Rule 1	Rule 2	Rule 3	Rule 4
Username (T/F)	F	T	F	T
Password (T/F)	F	F	T	T
Output (E/H)	E	E	E	H

Fig: Decision Table



The image shows a user login interface. At the top right is a 'Create Account' button. Below it is an illustration of a laptop with a smiley face on the screen and a red padlock icon with an exclamation mark. The main section is titled 'USER LOGIN'. It contains two input fields: 'Email or Username' and 'Password'. Below these fields are two checkboxes: 'Remember' and 'Forgot password?'. At the bottom is a 'LOGIN' button.

Parameter

- Requirement ID
- Requirement Type & Description
- Test Cases with Status

Req No	Req Desc	Testcase ID	Status
123	Login to the application	TC01,TC02,TC03	TC01-Pass TC02-Pass
345	Ticket Creation	TC04,TC05,TC06, TC07,TC08,TC09 TC010	TC04-Pass TC05-Pass TC06-Pass TC06-Fail TC07-No Run
456	Search Ticket	TC011,TC012, TC013,TC014	TC011-Pass TC012-Fail TC013-Pass TC014-No Run

Requirement Traceability Matrix

Requirement Traceability Matrix (RTM) is a document that maps and traces user requirement with test cases.

It captures all requirements proposed by the client and requirement traceability in a single document, delivered at the conclusion of the Software development life cycle.

USE CASE

In software testing, a use case is a graphic representation of business needs explaining how the end-user will cooperate with software or an application.

A GOOD CASE

A good test case is one that finds a defect. But all test case will not find defects, so a good test case can also be one which has all the prescribed details and coverage.

BUG Life CYCLE

The steps for Bug Cycle:

- Once the bug is identified by the tester, it is assigned to the development manager in open status.
- If the bug is a valid defect the development team will fix it.
- If it is not a valid defect, the defect will be ignored and marked as rejected.
- The next step will be to check whether it is in scope. If the bug is not the part of the current release then the defects are postponed.
- If the defect or bug is raised earlier then the tester will assign a DUPLICATE status.
- When bug is assigned to developer to fix, it will be given a IN-PROGRESS status.
- Once the defect is repaired, the status will change to FIXED at the end the tester will give CLOSED status if it passes the final test.

BUG ADVOCACY

Bug Advocacy is a **presentation of the facts and data around a bug**; it is an assessment of a bug's impact and the consequences of not fixing it

BUG REPORTING

An important asset in Bug Advocacy is the bug report. The report should be written using clear language, so that stakeholders can understand the issue and clearly assess its severity. Other supporting material such as screenshots, recordings, stack traces and log snippets can be very helpful in communicating the issue.



How the customer explained it



How the project leader understood it



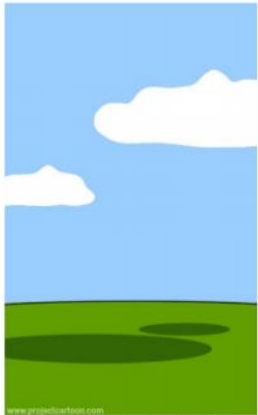
How the analyst designed it



How the programmer wrote it



How the business consultant described it



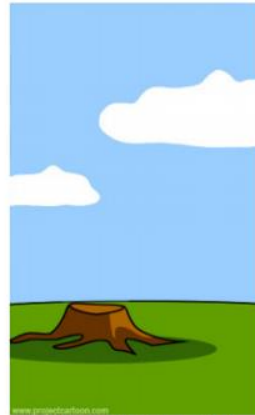
How the project was documented



What operations installed



How the customer was billed

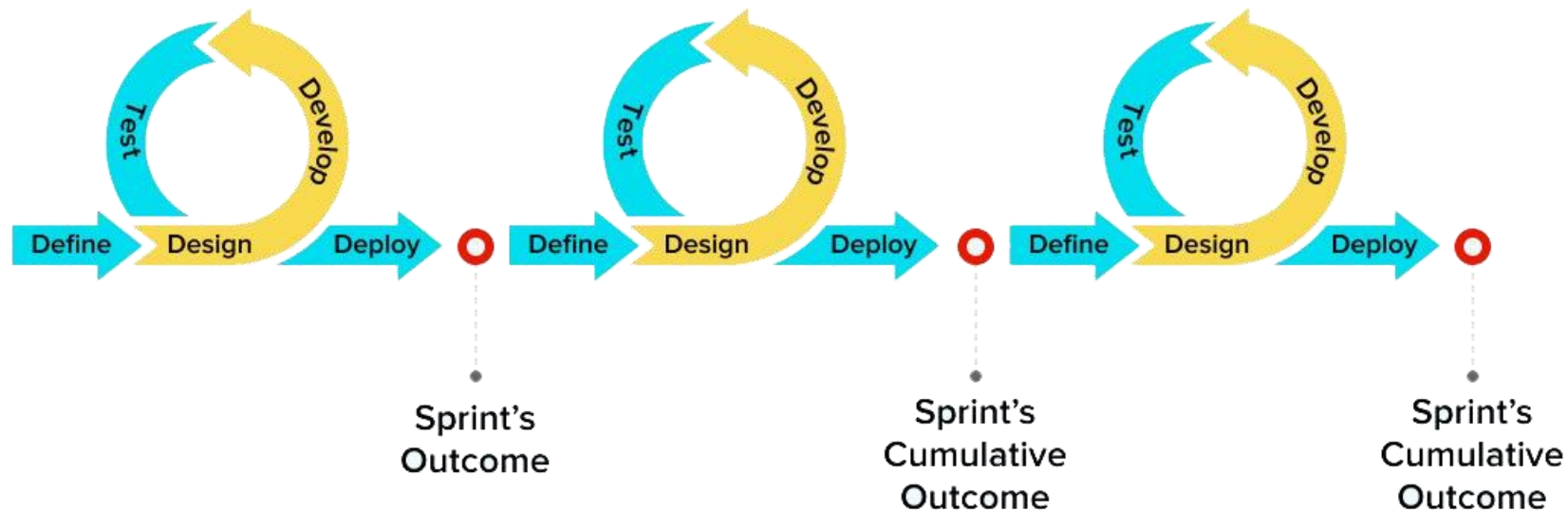


How it was supported



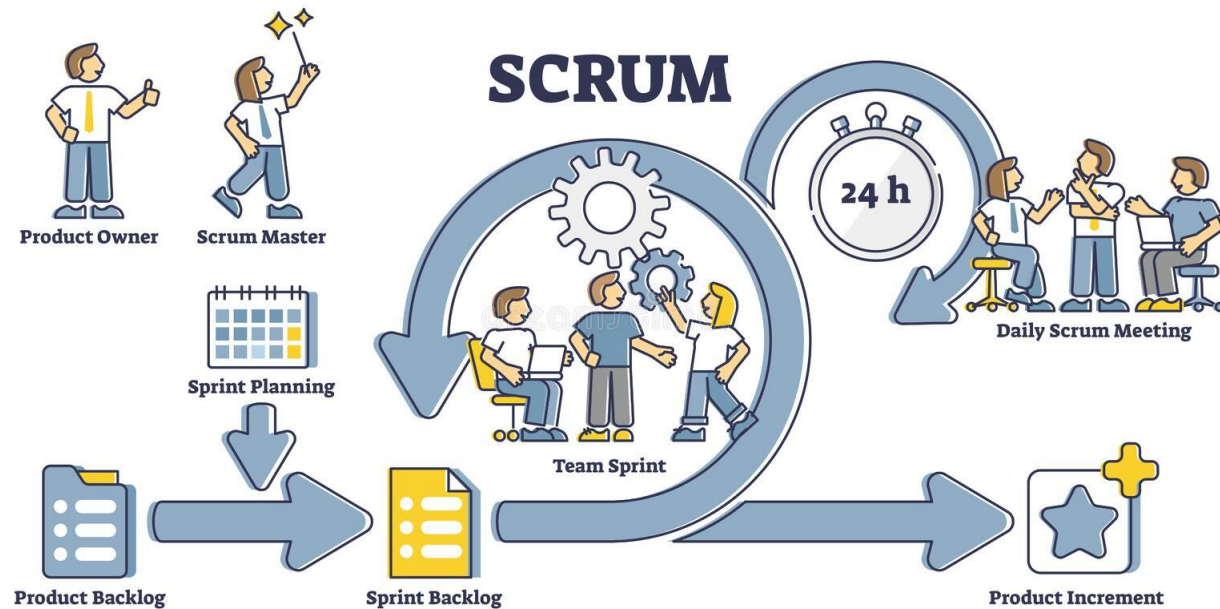
What the customer really needed

Imagine
a Scenario



AGILE

Agile is a project philosophy that takes an iterative approach towards the completion of a project.



SCRUM

SCRUM is an agile development method which concentrates specifically on how to manage tasks within a team-based development environment.