Software Testing

What is the need of Testing?
Testing is important because software bugs could be expensive or even dangerous. Software bugs can potentially cause monetary and human loss, and history is full of such examples.

- In April 2015, Bloomberg terminal in London crashed due to software glitch affected more than 300,000 traders on financial markets. It forced the government to postpone a 3bn pound debt sale.
- Nissan cars recalled over 1 million cars from the market due to software failure in the airbag sensory detectors. There has been reported two accident due to this software failure.
- Starbucks was forced to close about 60 percent of stores in the U.S and Canada due to software failure in its POS system. At one point, the store served coffee for free as they were unable to process the transaction.
- Some of Amazon's third-party retailers saw their product price is reduced to 1p due to a software glitch. They were left with heavy losses.
- Vulnerability in Windows 10. This bug enables users to escape from security sandboxes through a flaw in the win32k system.
- In 2015 fighter plane F-35 fell victim to a software bug, making it unable to detect targets correctly.
- China Airlines Airbus A300 crashed due to a software bug on April 26, 1994, killing 264 innocents live
- In 1985, Canada's Therac-25 radiation therapy machine malfunctioned due to software bug and delivered lethal radiation doses to patients, leaving 3 people dead and critically injuring 3 others.
- In April of 1999, a software bug caused the failure of a $1.2 billion military satellite launch, the costliest accident in history
- In May of 1996, a software bug caused the bank accounts of 823 customers of a major U.S. bank to be credited with 920 million US dollars.

Date:

Software Testing

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free.

It involves: execution of system components using
manual or
Automated tools
to evaluate one or more properties of interest.

Purpose:
To identify errors
gaps
missing requirements in contrast to actual req.

Why software testing is important?

If there are any error should be identified and removed before delivery of software product.
It ensures
- reliability
- Security
- High Performance
  - → that saves time.
  - → cost effectiveness
  - → customer satisfaction.

Date:

Types

Here are the software testing types.

- Functional testing
- Non functional testing / Performance Testing
- Maintenance

- Unit T
- Integration T
- Smoke T
- UAT (User Acceptance testing)
- Localization
- Globalization
- Interoperability
- So on

- Performance
- Endurance
- Load
- Volume
- Scalability
- Usability
- So on

- Regression
- Maintenance

Date: M T W T F S S

Testing Strategies in SE

Unit Testing

→ Basic approach followed by programmes to test the unit of program.

→ Helps developess to know the individual unit of code is working properly or not.

Integration Testing

→ Focuses on construction and design of software.

→ You need to see that integrated units are working without errors or not.

System Testing

Software is compiled as whole and tested as whole.

This strategy checks functionality, security, portability amongst others.

Error terms

Error: Human made mistakes. Incorrect results produce errors.

Fault: State of software caused by errors.

Bug: Presence of error is bug.

Failure: Deviation of software from expected results.

Test Case

A test case is a specific procedure of testing a particular requirement.

It includes:

→ identification of specific requirement tested
→ Test case success/failure criteria
→ Specific steps to execute test
→ Test Data

Details of Strategies of SE

Unit Testing (White box)

→ Individual components are tested.
→ It is a path test.
→ Focus on relatively small segment of code.
→ Aim to exercise a high percentage of the internal path.

White box testing

When testes knows the code behind functionality and uses that knowledge for testing purposes. Done by developes, code is visible for developes that's why white box testing. (Before giving to Test Engineer) called

Date: M T W T F S S

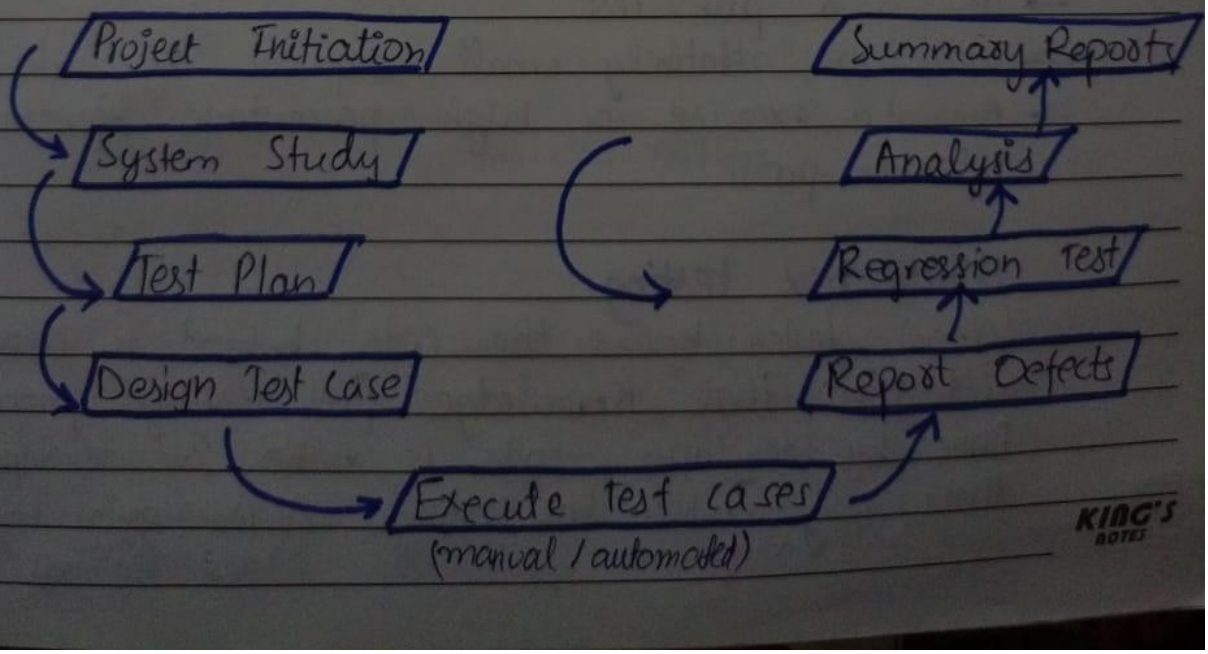
A person makes an **Error**
That creates a **fault** in software
That can cause a **failure** in operation

Testing In SDLC

→ Requirements
→ Analysis
→ Design
→ Coding
→ Testing
→ Implementation
→ Maintenance

(Testing starts from requirement phase)

Testing Life cycle



KING'S NOTES

Date:

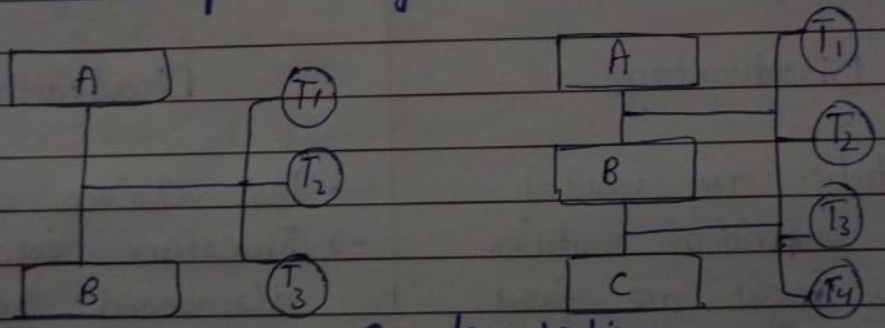
Disadvantage :

cost, rapidly changing code, missed cases (If explanation is required, see the text under heading disadvantages of white case box testing).

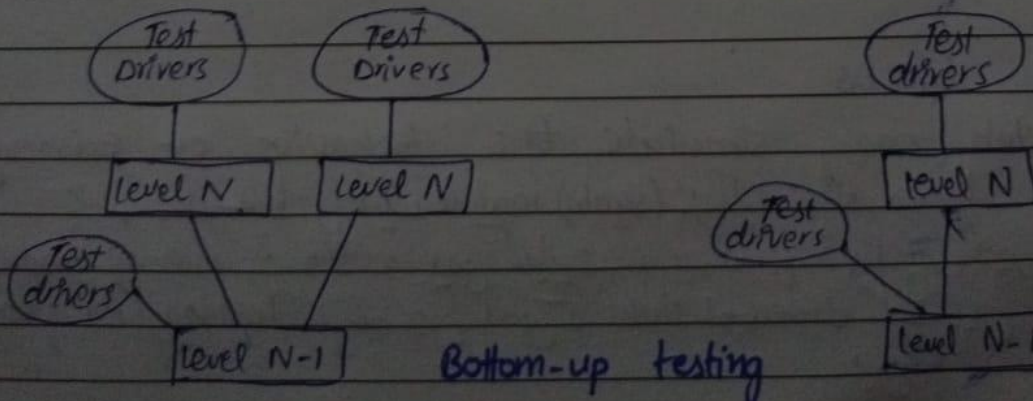
Integration:

After completing unit testing and dependent modules development, programmers connect the modules for Integration testing.

↳ Top-down Integration Test
↳ bottom-up Integration Test



Top-down testing



Bottom-up testing

KING NOTES

Date: M T W T F S S

Top - down

- Control Program is tested first.
- Modules are integrated one at a time.
- Emphasize on interface testing.

Advantages

- No test drivers needed
- Interface errors are discovered easly. Modular features aid debugging.

Disadvantage

→ Test stubs are needed
→ Errors in critical modules at low levels are found late.

Bottom - up

- Allow easly testing aimed at proving feasibility
- Emphasize on module functionality and performance.

Advantages

- No test stubs are needed
- Errors in critical modules are found easly.

Disadvantage

→ Test drivers are needed.
→ Interface errors are discovered late.

Stubs

A stub may simulate the behavior of existing code. جب ہمارے پاس ایک (signin) module ہے اور دوسرا نیا module (signup) ہے جو ابھی مکمل نہیں ہوا تو ہم ڈمی dummy drivers اور stubs کو Testing کرتے ہیں اس کو code کہتے ہیں۔ یہ ہوتے ہیں۔

KING'S NOTES

Date: M T W T F S S

§ Stubs are basically known as "called programs" and are used in Top-down integration testing.
§ Drivers are "calling program" and are used in bottom-up integration testing.

Black box Testing

→ Black B.t is done by Test Engineer.
→ Check functionality of application or the software according to customer's / client's needs.
→ Code is not visible while performing test that's why black box T.
(called)

Gray Box Testing

- Combination of white and black box testing.
- Performed by the person who knew coding and testing.

What are the benefits of Software Testing?

Here are the benefits of using software testing:

- **Cost-Effective:** It is one of the important advantages of software testing. Testing any IT project on time helps you to save your money for the long term. In case if the bugs caught in the earlier stage of software testing, it costs less to fix.
- **Security:** It is the most vulnerable and sensitive benefit of software testing. People are looking for trusted products. It helps in removing risks and problems earlier.
- **Product quality:** It is an essential requirement of any software product. Testing ensures a quality product is delivered to customers.
- **Customer Satisfaction:** The main aim of any product is to give satisfaction to their customers. UI/UX Testing ensures the best user experience.

SUMMARY OF SOFTWARE TESTING BASICS:

- Define Software Testing: Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is Defect free.
- Testing is important because software bugs could be expensive or even dangerous.
- The important reasons for using software testing are: cost-effective, security, product quality, and customer satisfaction.
- Typically Testing is classified into three categories functional testing, non-functional testing or performance testing, and maintenance.
- The important strategies in software engineering are: unit testing, integration testing, validation testing, and system testing.

Disadvantages of white-box testing

The disadvantages to white-box testing include its cost, rapidly changing code, and missed cases.

Expensive

Because white-box testing is more thorough it becomes very expensive in time and cost to conduct. Although unit tests alleviate this somewhat,

there is an initial investment that must be done to write the unit tests. Also, this type of testing can scale badly with large applications. It becomes virtually impossible to test every branch of code.

Compared to black-box testing, white-box testing requires skilled testers with programming knowledge. This increases the cost and could mean that developers are pulled off of developing new features. These costs all must be considered when conducting white-box testing.

Rapidly Changing Code Base

Automated test cases become a waste if the code base is rapidly changing. Often times, redesigns or reworks will cause most written test cases to be useless and in need of a rewrite.

Missed Cases

White-box testing only validates and tests features that are currently there. If a feature is only partially implemented or something is missing, white-box testing will not pick up on this. This is where requirements driven black-box testing is superior.

White-box testing has several clear advantages and disadvantages. Whether the cost is worth the advantages must be carefully considered, especially since mileage may vary from project to project.

How to perform Manual Testing

o First, tester observes all documents related to software, to select testing areas.

o Tester analyses requirement documents to cover all requirements stated by the customer.

o Tester develops the test cases according to the requirement document.

o All test cases are executed manually by using Black box testing and white box testing.

o If bugs occurred then the testing team informs the development team.

o The Development team fixes bugs and handed software to the testing team for a retest.

Advantages of Manual Testing

o It does not require programming knowledge while using the Black box method.

o It is used to test dynamically changing GUI designs.

o Tester interacts with software as a real user so that they are able to discover usability and

user interface issues.