

Assignment No - 3

Government Polytechnic Tumakur.

Subject :- Software Testing

Course code :- CO16410

Roll No :- SD18CO008

Name :- Parvush Bhimrao Khobragade.

### SECTION - I

Q.1) a) - Quality Assurance :- Quality Assurance is popular known as QA testing is defined as an activity to ensure that an organization is providing the best possible product or service.

b) - Quality Control : Quality Control is popular known as QC. It is defined as process to used to ensure quality in product or a service.

b) Following list of bug:-

- 1) functionality Error
- 2) communication Error
- 3) Syntactic Error
- 4) Error handling Error

c) i) Black Box Testing :-

1) knowledge of internal working structure is not required.

2) It also known as fun<sup>n</sup>t, data driven T. & closed Testing

white Box T.

1) knowledge of internal working structure is required

2) it also known as structural T. :  
code base T. & transparent Testing

## Types of Functional Testing

3) Includes Error Guessing Method

it proceeds by verifying the system  
boundaries & data domain.

4) Not considered for algorithm testing

- HOTT it is well suitable & recommended  
for algorithm testing.

d) Dynamic Testing

Dynamic Testing is a type of SW Testing which is performed to analyze the dynamic behavior of a code.

e) Integration Testing

It is defined as a type of SW Testing where SW modules are integrated logically and tested as group.

TYPE

- Top-down

- Bottom up

- Sandwich

f) Compatibility Testing :-

Compatibility testing is nothing but the compatibility of existing or living together: in normal life; oil is not compatible with water, but milk can easily combine with water.

It checks whether your SW is capable of running on different hardware, OS, appl', network or mobile device.

Q.2

a) i) STATIC TESTING TOOL	ii) DYNAMIC TESTING TOOL.
ii) Testing was done without exiting the program	i) Testing is done by exiting the program
iii) This testing done through verification process	iii) This testing does not validation process
iv) Required tools of meeting of unit testing and integration testing	iii) comparatively requires lesser meetings
v) More required comment are highly recommended for good quality	iv) more defects are highly recommended for good quality
vi) cost of finding defects & statement coverage testing is less	v) cost of finding & fixing

### b) Explain Equivalent class partitioning

Equivalent class partitioning is a black box testing technique which can be applied to all levels of testing like Unit, integration, system etc. In this technique, you divide the set of test conditions into a partition that can be considered the same.

- it divides input data or set into different equivalence classes.
- you can apply this technique where there is a range in the input field.

### Example

- Let's consider the behavior of order pizza test box below.
- Pizza value 1 to 10 is considered valid. A user may type 70. While value 11 to 99 are considered invalid for order & an error message will appear, "only 10 pizza can be ordered".

Order Pizza:	Submit
--------------	--------

Hence the test condition is

- 1) ANY NO. greater to enter in the order pizza field is considered invalid
- 2)  $N < 0$  that is 0 or below it is considered invalid
- 3) Number 1 to 10 are considered valid
- 4) ANY 3 digit NO. say -100 is invalid.
- d) Compatibility Testing

Compatibility testing is type of software testing to check whether your software is capable of running on different hardware, operating system, application, network environment or mobile device.

There are two types of version checking

Compatibility  
Testing

backward

forward

- Backward compatibility testing (BCT)

BCT is to verify the behavior of the developed hardware / software with the older version of the hardware / software.

- Forward compatibility testing (FCT)

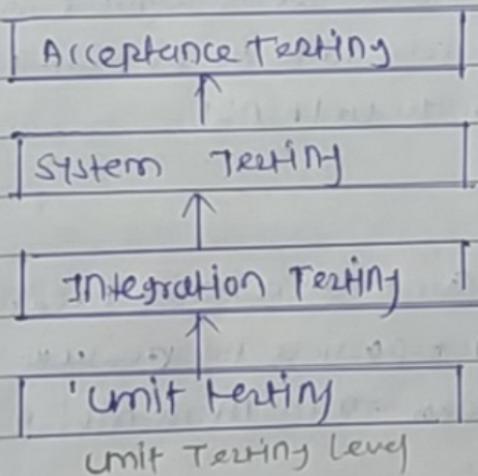
FCT is to verify the behavior of the developed hardware / software with newer version of the hardware / software.

### c) Unit Testing

unit testing type of software testing where individual unit or components of a software are tested.

unit testing is done during the development of a an application by the developer.

#### \* Why unit testing



1. unit testing help to find bugs early in development cycle

& save cost

2. it help to the developer to understand the code logic & enables them to make changes quickly.

3) Good unit test serve as project document-

## How to do unit testing (unit testing basic concepts)

- Manual
- Automatic

### Unit testing technique

- Statement coverage
- Decision coverage
- Branch coverage
- Condition coverage

### Tools

- 1) JUnit
- 2) Nunit
- 3) JMockit
- 4) EMMA

### Advantage

- Developers looking to learn what things is provided by a unit & how to use it can look at the unit test to gain a better understanding of the unit API.

### Disadvantage

- Unit testing can't be expected to catch every error in program, it is not possible to evaluate every executing path even in the most trivial program.

(3)

- 1) Software testing is an essential part of the software development life cycle (SDLC), playing a significant role in defining the success rate of a particular product, owing to the same reason the SW testing team plays a crucial role even after the product's development is completed.

### Role of good tester

#### 1) QA Leader

It is most important member of testing team, while it is extremely understanding of the testing process or methodology, it deciding the best test target & schedule, planning the entire testing plan, etc.

#### 2) Test Lead

Technical expertise related to the test program & approach.

Arranging walk-through test design & products.

#### 3) Knowledge

It complete knowledge or skills about that product it handle over all tool (analysis), what requirement to build fast our project if know.

## 4) Time :-

TO work correctly until the time is over.

Overdue - failure could happen till the end of time.

Time constraint is also known as deadline of project.

5) Tools

The tester know about different tools used to test it can be understood.

Tool used to test it can be understood.

6) Good Judgment :- S.T. need to make good decision

about what they will test, how long it will take, only if

the problem they see looking at easily acceptable.

most pain is to come to decisions about what

test to be performed and when

## 2) Black Box Testing

It is defined as a testing technique in which

functionality of the application under Test (AUT) is tested

without looking at

- internal code structure

- implementation details

- knowledge of internal parts of the system

- etc

- Based on requirement & specification

Input Black Box Output

: Technique of Black Box T.

## 1) Equivalence Class Testing

It is used to minimize coverage to no. of possible test cases on optimum level while maintaining test coverage.

## 11 - BOUNDARY TEST

Ex.

Age	_____	(accepts 18 to 50)	18 to 50
-----	-------	--------------------	----------

Invalid | Valid | Invalid | Valid

$$L = 17 \quad 18 + 50 = 68 \quad D = 51 \quad 50 + 1 = 51$$

## 2) Boundary Value Testing

It's focused on the value at boundaries. This technique determine whether a certain range or value are accepted by system or not.

Ex.

Age	_____	(accept 18 to 50)	18 to 50
		Invalid   Valid (min, min, min, min + 1)	
	17	18(14, 42, 51)	51

## 3) Positive testing

It's type of test that can be performed on the system  
by providing the valid data

Ex.

positive Testing	mobile No : [1234567890]	Mobile No
	only num	

## 4) Negative testing

It type of test that can be performed on the system  
by providing invalid data as input

Ex.

Negative testing	mobile No : [!@#%\$^&*]	Mobile No
	only num	

## SECTION - II

### Q.4 Q3 Accessibility Testing

Accessibility testing is defined as a type of software testing performed to ensure that the application being tested is accessible by people with disabilities like hearing, color blindness, etc. or other disadvantaged groups.

### 6) Grey Box Testing

Grey Box testing is a software testing method, which is a combination of both White Box testing & Black Box testing method.

- In white Box - internal structure (code) is known
- In Black Box - internal structure (code) is unknown
- Grey Box - internal structure (code) is partially known

Black Box T,	+	White Box T	=	Grey Box T
--------------	---	-------------	---	------------

### c), Manual Testing

(i) process of manually testing  
SW for defects

Automated testing

process of testing a specific SW  
to control the execution of test

### (ii) Requires more time

Requires less time

(iii) suitable for small to medium scale projects  
suitable for complex & large projects

### d) Automated testing tools

- ACCECO
- Test project
- Ranorex
- TestCraft
- ZEUS

### e) Automation testing benefit

- ① Accelerated Results
- ② Testing Efficiency Improvement
- ③ Throughput In Testing
- ④ Earlier Detection of defects

### f) goal of testing planning

A Test Plan is a detailed document that describe test strategy, objective, schedule, estimation & deliverables & resource required for testing.

#### [TOD]

- Help to people outside the test team such as developer, business managers.
- Test Plan guides our thinking, it like rule book, which need to be followed.
- ISO

(Q5)

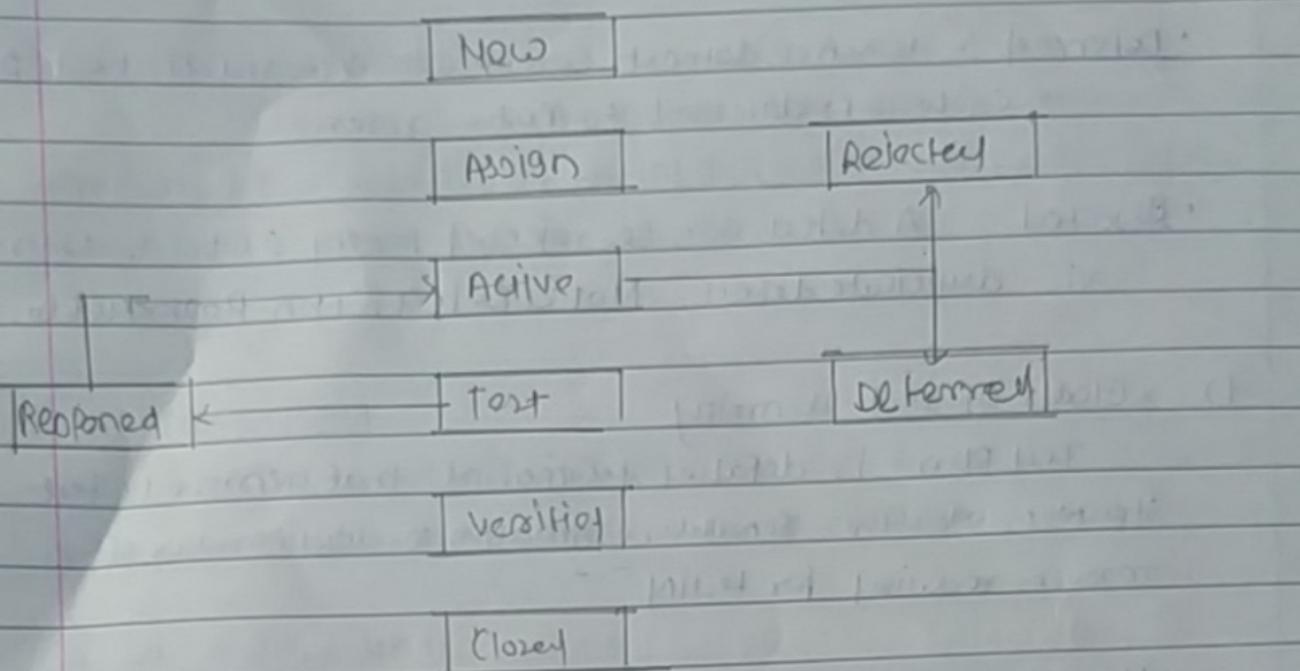
Alpha Testing	Beta Testing
① Alpha testing performed by tester who are usually internet employees of the organization	② Beta testing is performed by client or Eng. user who are not employees of the organization
③ Alpha testing performed at developer site	Beta testing performed at a client location or on a user's product
④ Reliability & security testing are not performed in Alpha testing	Reliability, security, robustness checked during Beta testing
⑤ Alpha testing requires a lab environment for testing environment	⑥ Beta testing doesn't require any lab environment or testing environment

### Buy or Detect life cycle

Buy life cycle is the journey of defect cycle, which a defect goes through during its lifetime. It varies from organization to organization & also from project to project as it is governed by the software testing process and also depends upon the tool used.

#### \* Buy life cycle stages:-

- New :- Potential defect that is raised & yet to be validated.



Assigned - Assigned against a development team to address it but yet resolved.

Active :- The defect is being addressed by the developer & investigation is under process. At this state there are two possible actions viz. Determined or Rejected.

Test - The defect is fixed and ready for testing.

Verified - The defect that is tested & the test has been verified.

G1 QA 1

Closed - The final state of the defect that can be closed after QA testing or can be closed if the defect is duplicate or considered as Not a defect.

Reopened - When the defect is not fixed, QA 'rejects' the defect.

- Deferred :- When a defect cannot be addressed in that portion cycle it is deferred to future release
- Rejected : A defect can be rejected for any of the reasons viz. Duplicate defect, Not a defect, Non Reproducible.

### t) Goals of test planning

Test Plan - is detailed document that describe the test strategy, objectives, schedule, estimation & deliverables to resources required for testing

### • How to write plan

#### ① Analyze the Product

How can you test a product without any information about it?  
The answer is Impossible.

- Suppose one website
- who will use the website?
- what is it used for?
- how will it work?
- what are SW/HW req.?

We can review website & Product documentation

#### ② Develop Test Strategy

Test strategy is a middle step in making a test plan.  
A test strategy document, is a high-level document, which is usually developed by Test Manager. This document defines:

- The project's testing objectives & the means to achieve them
- Determining testing effort and cost

### (3) Define test objective

Test objective is the overall goal to achievement of the test execution. The objective of testing is finding as many bugs as possible; ensure that the software under test (SUT) is bug-free before release.

### (4) Define criteria

Specify the critical suspension criteria for a test procedure or test judgement in case the SUT fails or test succeeds.

#### ① Suspension criteria

#### ② Exit criteria

### ⑤ Usability testing

Usability testing measure how easy to use, to use-friendliness a SW system is. Here, a small set of target end-user "use" the SW system to expose usability defects.

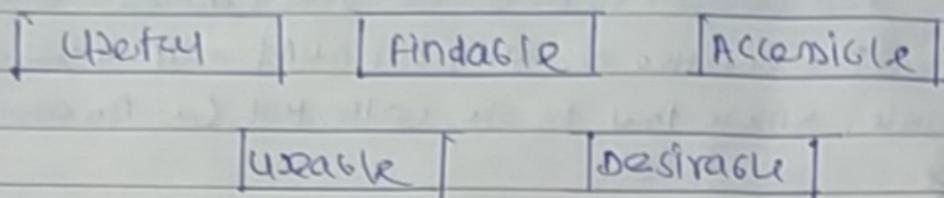
#### • Why do Usability testing

Aesthetics & design are important. How does a product look usually determine how well it works.

There are many SW applications / websites, which miserably fails, once launched, due to following reason:

- Where do I click next?
- Which page need to be navigated?
- Which icon or jargon represent what.
- Links may or can not consistent or effectively displayed.

## Ex. Usability testing



## How to test

- ① Planning :- What our goal of testing are determining.
- ② Recruiting :- Desired Number of testers as per your usability test plan.
- ③ Usability testing :- During this phase, usability tests are actually executed.
- ④ Data Analysis :- Data from usability test is thoroughly analyzed to derive meaningful inference to give actions.
- ⑤ Reporting :- Finding ty usability test is shared with all concerned stakeholders which can include design, developer, client & CEO.

Q.  
6.

a) Software testing.

Software testing is defined as an activity to check whether the actual result matches the expected result so as to ensure that the software system is defect free.

To make good software tester.

Here is list of traits that most software tester should have.

i) They are explorers :-

S.T. aren't afraid to move into unknown situations. They love to get new pieces of software installed on their PCs, and see what happens.

ii) They are troubleshooters

S.T. are good at figuring out why something doesn't work. They love puzzles.

iii) They are unshakenly persistent :-

S.T. keep trying, they may see a bug that quickly vanishes or it difficult to recreate. Rather than dismiss it as a chance chance they will try every way possible find out.

#### ④ They are creative

Testing to obvious isn't sufficient for SQA factors.  
Their job is to think up creative and over off  
to cool approaches to find bugs.

#### ⑤ They are mature

They aim for perfection, but know when it's  
unattainable and they're okay with getting as  
close as they can.

#### ⑥ They practice good judgement

SQA need to make decisions about what they  
will test, how long it will take, and fit to problem they're  
looking at exactly a bug.

### Test tool

There are tons of s/w testing tools available in the  
market, & with the plethora of choice, it becomes  
difficult to zero in on the best tool for your  
project

### Test tool list

- ① Xray
- ② selenium
- ③ Junit
- ④ Testpat
- ⑤ TestComplete
- ⑥ QTP.

## 1) Selenium

Selenium is the most popular automated testing tool. It specially designed to support automation testing of functional aspects of web based application, wide range of platform and browser.

### \* features

- it offers to support for parallel test execution that reduce the time taken in executing parallel test.
- Selenium need very less resources when compared to other testing tools.
- ~~Fast~~ it support to many known programming languages like Java, Python, C#, Perl, PHP & JavaScript.

## ② Tottpy

Tottpy is a similar and more accessible manual test tool that prioritises pragmatism over process. Instead of managing cases one at a time, it uses checklist-inspired test plan that can be adapted to a wide range of styling including Exploratory testing, the manual side of Agile, syntax-highlighted BDD, and even traditional test case management.

### Key features

- Guest tester invited by email, who don't need account
- simple enough to use by non-testers, get most everyone to help at release time.
- Keycode-driven editor with javascript-powered UI.
- Drag-and-drop organisation of test plans
- Add new tests during testing, as you think of new ideas.