

5/8

Q3]

c]

① Verification in Software Testing

- i) Verification is a process of checking documents, design, code and program in order to check if the software has been built according to the requirements or not. The main goal of the verification process involves activities like reviews, walk-throughs, and inspection.
- ii) Verification is the static testing.
- iii) It checks if the software conforms to specifications or not.

② Validation in Software Testing

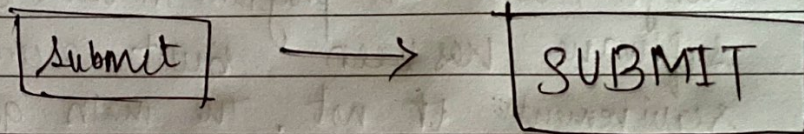
- i) It is a dynamic mechanism of testing and validating if the software product actually meets the exact needs of the customer or not. The process helps to ensure that the software fulfills the desire use in an appropriate environment. The validation process involves activities like unit testing, integration testing, system testing and user acceptance testing.
- ii) Validation involves code execution, dynamic testing.
- iii) The goal of validation is an actual product.
- iv) Validation is executed software code with the help of testing team.
- v) Consists of execution of programs and is performed by computer.

6/8

Example of verification and validation

Consider the following specification:

A clickable button with name submit



- Verification would check the design doc and correcting the spelling mistake
- So now the new button with name Submit
- Once the code is ready, validation is done.
A validation test found, button is not clickable
- Now, development team will make the submit button clickable in the validation process.

From the above example, it is clear that if verification is not done first then the validity is baseless and as verification checks the authenticity and hence if the document or button is errorless, it can be validated further.

Hence, verification comes first and then the validation is done

Q3)

B)

① COCOMO II is the revised version of the original COCOMO (Constructive Cost Model) is developed at University of Southern California. It is a model that allows one to estimate the cost, effort and schedule when planning a new software development activity.

② It contains 3 submodels.

i) End user programming

Application generators are used in this sub model. End user write the code by using application generators.

eg:- spreadsheets, report generator etc.

③ ii) Intermediate sector

- Application generators and composition aids

- Application composition sector

- System Integration

iii) Infrastructure sector

This category provides infrastructure for the software development like OS, DBMS, User Interface management system, networking system.

eg:-

Consider a database application project with

1. The application has four screens with four views each and seven data tables for three servers and four clients

2. Application may generate two reports of six section each & from seven data tables for two servers and three clients.

8/8

Q3] 200 KLOC

Q] Cal Avg staff size, prod effort.
development time

801^a :

Effort: $a \times (KLOC) \exp a2$

$$E = 3.8 (200)^{1.12}$$

$$= 1133.12 \text{ PM}$$

Development Time:

$DT = I \times (\text{effort}) \exp b2 \text{ months}$

$$DT = 2.5 (1133.12)^{0.35}$$

$$= 29.3 \text{ PM}$$

$$\text{Avg staff size (SS)} = \frac{E}{D} \text{ persons}$$

$$= \frac{1133.12}{29.3}$$

$$= 38.67 \text{ persons}$$

$$\text{Productivity} = \frac{KLOC}{E} = \frac{200}{1133.12}$$

$$= 0.1765 \text{ KLOC/PM}$$

$$\therefore P = 176 \text{ LOC/PM}$$