

- SE ISE - 3

1) Verification (right way)

i) Implies : Are you building it right?

ii) It is done by developer

2) validation (right thing)

Implies : Are you building right thing?

It is done by tester.

Uses methods like, inspections, review, walkthrough, desk-checking.

Uses methods like, black box testing white box testing.

iv) Objective process

Subjective process.

v) takes place before validation.

takes place after verification.

vi) Does not involves executing the code.

Involves executing the code.

vii) Verification is to check whether the s/w conforms to specification validation is to check wheather s/w meets the customer expectat's & requirements.

3) Steps followed in testing:

4) Short note on architectural & component design.

- DS/W architectural design represent the structure of data & program components that are req. to build any computer based system.
- 2) It is not a operational s/w , it is just a representation/blue print.
- 3) Architectural design the system in context
 - system engineer must model the context
 - context diagram.
- 4) Architectural design represents the design of software.
- 5) Component design is the design of components & modules after the architectural design phase.

6) It defines data structures, algorithms, interface characteristics, communication mechanisms allocated to each component for system development.

Q5) Cohesion & Coupling:

1) Cohesion: It defines the degree of intra-dependability within elements of module. Greater the cohesion, better is the program design.

Types of cohesion:

- 1) coincidental cohesion.
- 2) Logical cohesion.
- 3) Temporal cohesion.
- 4) Procedural cohesion.
- 5) communicational cohesion.
- 6) sequential cohesion.
- 7) Functional cohesion.

2) Coupling: It defines the level of inter-dependability among modules of a program. It tells at what level the modules interfere & interact with each other. Lower the coupling better the program.

Types of coupling:

- 1) content → .
- 2) common → .
- 3) control → .
- 4) stamp → .
- 5) data → .

Q.6) What is structured design? Illustrate the structured design process from DFD to Structured chart.

- 1) It is a development method that allows the analyst to understand the system & its activities in a logical way.
- 2) Tools used during structured analysis:
 - Data flow diagrams
 - Data dictionary
 - Decision trees
 - — table
 - Structured English
 - pseudocode.
- 3) DFD is a technique developed by Lam Constantine to express the req. of system in graphical forms.
- 4) DFD is easy to understand.
- 5) symbols used in designing DFD:
 - square  
 - arrow  
 - circle 
 - open rectangle 

Q.7) Necessity of Real time system design process with eg.

A system must have 3 basic constraints:

- 1) must have some structure & behavior which is designed to achieve a predefined objective.

- 2) Interconnectivity & independence must exists among system components.
- 3) Objectives of organization have higher priority than the objective of subsystems.

e.g. traffic management system, payroll system, automatic library system, human resource info. system.

Q.8) What is SW architecture? diff type of SW architectural styles

- 1) It serves a blue print for system.
- 2) It defines a structured solution to meet all the technical & operational requirements.
- 3) Types of arch styles:
 - Data centered architectures
 - Data flow
 - call & return

Q.9) Basic rules of interface design.

- 1) strive for consistency.
- 2) Enable frequent users to use short cut.
- 3) offer informative feedback.
- 4) Design dialog & yield closure.
- 5) offer simple error handling.
- 6) Permit easy reversal of actions.
- 7) Support internal locus of control.

Q 10) Explain design principles in detail.

- 1) SW design should correspond to analysis model.
- 2) Choose right programming paradigm.
- 3) SW design should be ~~uninformative~~ & integrated.
- 4) SW design should be flexible.
- 5) SW design should ensure minimal conceptual & ~~er~~ errors.
- 6) SW design should represent real world problems.
- 7) SW reuse.
- 8) Designing for testability.
- 9) prototyping.

Q)

Q.11) S/W testing & good characteristics of good tester.

- 1) S/W testing is a critical element of S/W quality assurance & represents the ultimate review of specs., design & coding.
- 2) Characteristics: - all tests should be traceable to cust. req.
 - Test should be planned long before testing begins.
 - The pareto principle applies to S/W testing

Q.12) Objectives of testing:

- 1) Testing is a process of executing a prog. with the intend of finding an error.
- 2) A good test case is one that has high probability of finding an undiscovered error.
- 3) A successful test is one that uncovers as an yet undiscovered error.

Q.13) Why does S/W fails after it has passed from acceptance testing?

- 1) A QA team conducts acceptance test to ensure that S/W or app matches all req. & end user needs.
- 2) An acceptance test returns either a pass or fail results.
- 3) A fail suggests that there is a flaw present, & S/W should not go into product?

(Q.14) White box testing

- Internal structure known to tester
- performed by developers
- applicable for ~~big~~ lower level testing
- prog. knowledge required
- implementat? knowledge required
- developer should know abt. internal design of code.

Black box testing

- Int. struct. not known to tester
- performed by test engineers.
- higher level testing.
- not required
- not required
- no need to know about internal design of code.

Q.15) Alpha testing

Beta testing

1) conducted at developer's end

conducted at user's place

2) performed in environment controlled by developer.

performed in enviro. which is not controlled by developer.

3) It is performed before SW is released to end user.

performed after releasing SW to end users.

4) Developer keeps record of all errors & problems.

The end user records errors & prob.; later reports to developer.

5) Involves both black box & white box testing

involves only black box testing.

Q.16) testing principles:

1) All tests should be traceable to cust. req.

2) Test should be planned long before begin.

3) Testing should begin "in the small" & progress towards testing "in the large".

4) exhaustive testing is not possible.

5) To be most effective, an independent 3rd party should conduct testing.

6) Pareto principle can be applied to SW testing.

(Q.17) List various testing activities.

- 1) test planning
- 2) test case design
- 3) test execution
- 4) data collection
- 5) effective evaluation

(Q.18) testing strategies for conventional SW.

-
- 1) unit testing
 - 2) integration testing
 - 3) validation →
 - 4) system →

(Q.20) what is unit testing? why imp?, explain unit test considerations & test procedure.

-
- 1) Method by which individual units of source code are tested to determine if they are fit for use.
 - 2) Importance:
 - helps to find issues at early stage.
 - helps in maintaining & changing the code.
 - helps in reducing cost of bug fixes.
 - helps in simplifying debugging process.

(Q.21) Enlist challenges solved by devops.

- 1) Deployments taking too long.
- 2) Missing artefacts, scripts & other dependencies
- 3) confusion about what to test.
- 4) slow test processes.

- 5) unavailability of environments.
- 6) Fire fighting instead of innovation.
- 7) High dependency on specific individuals.

Q. 22) Benefits of Devops.

- 1) Speed: move faster as compare to other
- 2) ^{Rapid} Delivery: Fix bugs fast & deliver product fast
- 3) Reliability: high degree of reliability.
- 4) Scale:
- 5) ^{improved} collaboration:
- 6) security:

Q. 23) Principles of Devops. (How it works)

- 1) Development & operations teams are no longer "siloes". X
- 2)
 - 1) culture
 - 2) automation
 - 3) measurement
 - 4) sharing.