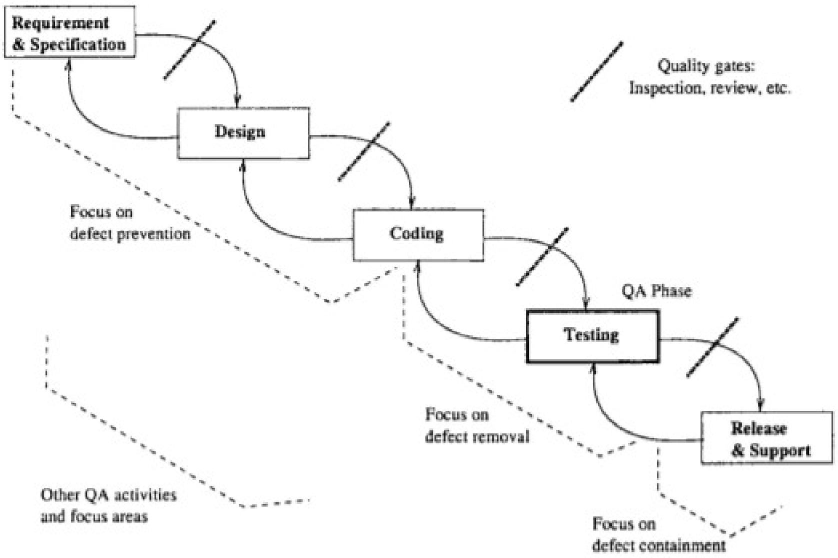
Deepraj Bhosale
181105016

STQA ASSIGNMENT-2

1. Explain QA activities in waterfall process with the help of neat diagram.

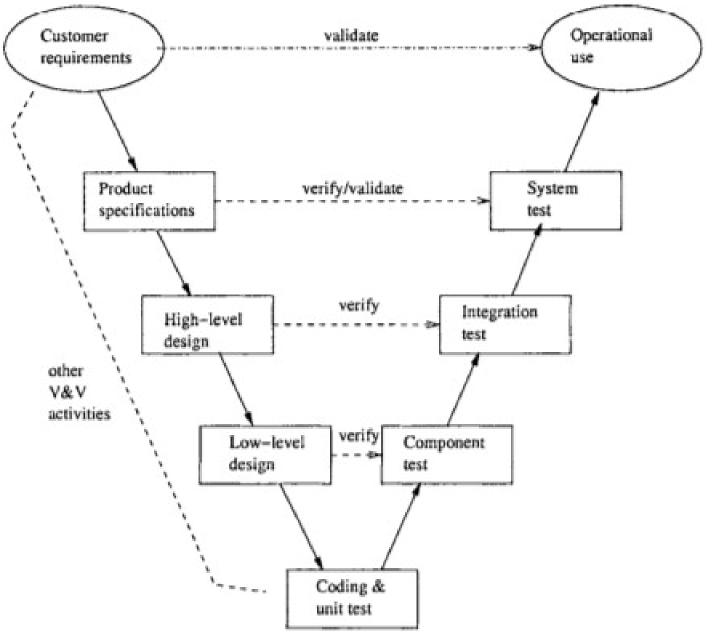

Requirement & Specification
Design
Coding
Testing
Release & Support
Quality gates: Inspection, review, etc.
QA Phase
Focus on defect prevention
Focus on defect removal
Focus on defect containment
Other QA activities and focus areas

-The phase with QA as the focus is the Testing phase.
-QA activities, typically inspections and reviews, carried out at the transitions from one phase to the next are shown as barriers or gates to pass. The exception to this is between testing and release, where the reviews are typically accompanied by acceptance testing.
-Other QA activities scatter over all other development phases: The general distribution scope is shown by the dotted bracket, with a focus on defect prevention in the early phases, a focus on defect removal during coding and testing phases, and a focus on defect containment in operational support

2. Compare validation and verification. Explain validation and verification activities associated with the V-Model

Verification	*Validation*
*1.*Verification is a static practice of verifying documents, design, code and program.	1.Validation is a dynamic mechanism of validating and testing the actual product.
*2.*It does not involve executing the code.	2. It always involves executing the code.
*3.*It is human based checking of documents and files.	*3.*It is computer based execution of program.
*4.*Verification uses methods like inspections, reviews, walkthroughs, and Desk-checking etc.	*4.*Validation uses methods like black box (functional) testing, gray box testing, and white box (structural) testing etc.
5.Verification is to check whether the software conforms to specifications.	5.Validationis to check whether software meets the customer expectations and requirements.
*6.*It can catch errors that validation cannot catch. It is low level exercise.	*6.It can catch errors that verification cannot catch. It is High Level Exercise.*
7.Target is requirements specification, application and software architecture, high level, complete design, and database design etc.	*7.*Target is actual product-a unit, a module, a bent of integrated modules, and effective final product.
8.Verification is done by QA team to ensure that	*8.Validation is carried out with the involvement*

the software is as per the specifications in the SRS document.	*of testing team.*
9. It generally comes first-done before validation.	*9.*It generally follows after verification.



Business Requirement Analysis

This is the first phase in the development cycle where the product requirements are understood from the customer's perspective. This phase involves detailed communication with the customer to understand his expectations and exact requirement. This is a very important activity and needs to be managed well, as most of the customers are not sure about what exactly they need. The acceptance test design planning is done at this stage as business requirements can be used as an input for acceptance testing.

System Design

Once you have the clear and detailed product requirements, it is time to design the complete system. The system design will have the understanding and detailing the complete hardware and communication setup for the product under development. The system test plan is developed based on the system design. Doing this at an earlier stage leaves more time for the actual test execution later.

Architectural Design

Architectural specifications are understood and designed in this phase. Usually more than one technical approach is proposed and based on the technical and financial feasibility the final decision is taken. The system design is broken down further into modules taking up different functionality. This is also referred to as High Level Design (HLD).

The data transfer and communication between the internal modules and with the outside world (other systems) is clearly understood and defined in this stage. With this information, integration tests can be designed and documented during this stage.

Module Design

In this phase, the detailed internal design for all the system modules is specified, referred to as Low Level Design (LLD). It is important that the design is compatible with the other modules in the system architecture and the other external systems. The unit tests are an essential part of any development process and helps eliminate the maximum faults and errors at a very early stage. These unit tests can be designed at this stage based on the internal module designs.

Coding Phase

The actual coding of the system modules designed in the design phase is taken up in the Coding phase. The best suitable programming language is decided based on the system and architectural requirements.

The coding is performed based on the coding guidelines and standards. The code goes through numerous code reviews and is optimized for best performance before the final build is checked into the repository.

Validation Phases

The different Validation Phases in a V-Model are explained in detail below.

Unit Testing

Unit tests designed in the module design phase are executed on the code during this validation phase. Unit testing is the testing at code level and helps eliminate bugs at an early stage, though all defects cannot be uncovered by unit testing.

Integration Testing

Integration testing is associated with the architectural design phase. Integration tests are performed to test the coexistence and communication of the internal modules within the system.

System Testing

System testing is directly associated with the system design phase. System tests check the entire system functionality and the communication of the system under development with external systems. Most of the software and hardware compatibility issues can be uncovered during this system test execution.

Acceptance Testing

Acceptance testing is associated with the business requirement analysis phase and involves testing the product in user environment. Acceptance tests uncover the compatibility issues with the other systems available in the user environment. It also discovers the non-functional issues such as load and performance defects in the actual user environment.

3. Give some examples of QA activities which are classified as validation activities.

-System testing, where the focus is the overall set of system functions to be provided to users;

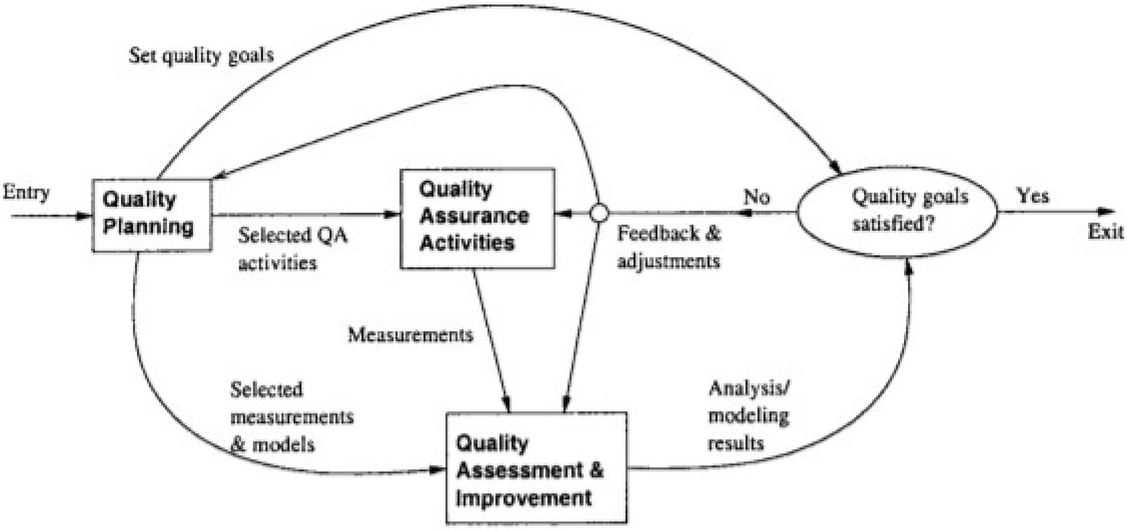
- Acceptance testing and beta testing, where the focus is the assessment of software acceptance or performance by users;

-Usage-based statistical testing, where the operational environment by target users is simulated during software testing before product release;

-Software fault tolerance, which focuses on providing continued service expected by customers even when local problems exist;

-Software safety assurance activities, which focus on providing the expected accidentfree operations or reducing accident damage when an accident is unavoidable.

4. With the help of neat diagram explain quality engineering process.



1. Pre-QA activities: Quality planning. These are the activities that should be carried out before carrying out the regular QA activities. There are two major types of pre-QA activities in quality planning, including: (a) Set specific quality goals. (b) Form an overall QA strategy, which includes two sub-activities: i. Select appropriate QA activities to perform. ii. Choose appropriate quality measurements and models to provide feedback, quality assessment and improvement.

2. In-QA activities: Executing planned QA activities and handling discovered defects. In addition to performing selected QA activities, an important part of this normal execution is to deal with the discovered problems. These activities were described in the previous two chapters.

3. Post-QA activities: Quality measurement, assessment and improvement These are the activities that are carried out after normal QA activities have started but not as part of these normal activities. The primary purpose of these activities is to provide quality assessment and feedback so that various management decisions can be made and possible quality improvement initiatives can be carried out .

5. List and explain major activities involved in quality assessment and improvement.

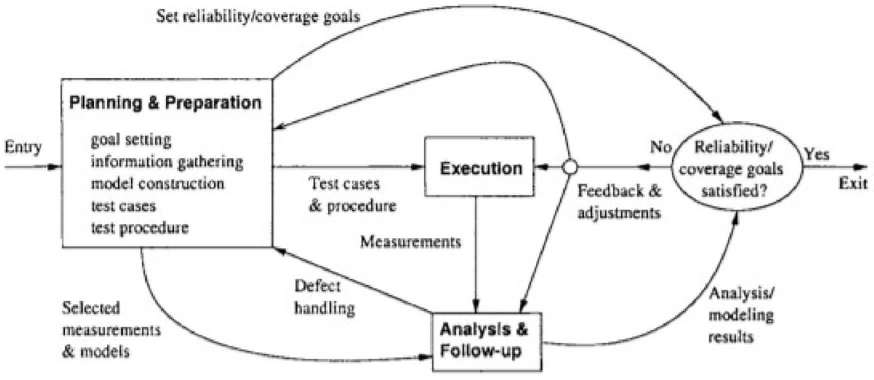
-Measurement: Besides defect measurements collected during defect handling, which is typically carried out as part of the normal QA activities, various other measurements are typically needed for us to track the QA activities as well as for project management and various other purposes. These measurements provide the data input to subsequent analysis and modeling activities that provide feedback and useful information to manage software project and quality.

-Analysis and modeling: These activities analyze measurement data from software projects and fit them to analytical models that provide quantitative assessment of selected quality characteristics or sub-characteristics. Such models can help us obtain an objective assessment of the current product quality, accurate prediction of the future quality, and some models can also help us identify problematic areas.
-Providing feedback and identifying improvement potentials: Results from the above analysis and modeling activities can provide feedback to the quality engineering process to help us make project scheduling, resource allocation, and other management decisions. When problematic areas are identified by related models, appropriate remedial actions can be applied for quality and process improvement.
-Follow-up activities: Besides the immediate use of analysis and modeling results described above, various follow-up activities can be carried out to affect the longterm quality and organizational performance. For example, if major changes are suggested for the quality engineering process or the software development process, they typically need to wait until the current process is finished to avoid unnecessary disturbance and risk to the current project.

6. Explain steps involved in quality planning.

1. Setting quality goals by matching customer's quality expectations with what can be economically achieved by the software development organizations in the following sub-steps: (a) Identify quality views and attributes meaningful to target customers and users. (b) Select direct quality measures that can be used to measure the selected quality (c) Quantify these quality measures to set quality goals while considering the marattributes from customer's perspective. ket environment and the cost of achieving different quality goals.
2. In forming a QA strategy, we need to plan for its two basic elements: (a) Map the above quality views, attributes, and quantitative goals to select a specific set of QA alternatives. (b) Map the above external direct quality measures into internal indirect ones via selected quality models. This step selects indirect quality measures as well as usable models for quality assessment and analysis.

7. With the help of neat diagram explain generic testing process.



-Testplanning andpreparation, which set the goals for testing, select an overall testing strategy, and prepare specific test cases and the general test procedure.
-Test execution and related activities, which also include related observation and measurement of product behavior.
-Analysis and follow-up, which include result checking and analysis to determine if a failure has been observed, and if so, follow-up activities are initiated and monitored to ensure removal of the underlying causes, or faults, that led to the observed failures in the first place.

8. Explain the sub-activities involved in test planning and preparation

-Goal setting: This is similar to the goal setting for the overall quality engineering process. However, it is generally more concrete here, because the quality views and attributes have been decided by the overall quality engineering process. What remains to be done is the specific testing goals, such as reliability or coverage goals, to be used as the exit criteria.
-Test case preparation: This is the activity most people naturally associate with test preparation. It includes constructing new test cases or generating them automatically, selecting from existing ones for legacy products, and organizing them in some systematic ways for easy execution and management. In most systematic testing, these test cases need to be constructed, generated, or selected based on some formal models associated with formal testing techniques.
-Test procedure preparation: This is an important activity for test preparation. For systematic testing on a large scale for most of today's software products and softwareintensive systems, a formal procedure is more of a necessity than a luxury. It can be defined and followed to ensure effective test execution, problem handling and resolution, and the overall test process management.

9. Compare white box testing with black box testing.

Sr. No.	Key	Black Box Testing	White Box Testing
1	Definition	Black box testing is the testing model in which the tester do not have the knowledge about the internal implementation of the application and testing performed by tester is at very high level that focuses on the behaviour of the application.	On other hand White box testing is the testing model in which tester is aware of internal implementation of the application and does the testing on the basis of that.
2	Also known as	Black box testing in programming world is also known with other terms such as data-driven, box testing, data-, and functional testing.	On other hand White box testing is also called structural testing, clear box testing, code-based testing, or glass box testing.
3	Type	Black box testing is the type of testing in which testing is based on external expectations as internal behaviour of the application is not made provided to the tester.	On other hand White box testing is the type of testing in which internal behaviour is known to the tester and hence he can test accordingly.
4	Automation	Automation is hard to practice in case of Black box testing as he is unaware of internal implementation of the	On other hand in case of White box testing automation is easier to implement as tester is well known

Sr. No.	Key	Black Box Testing	White Box Testing
		application and for which there is dependency on the developer.	with internal implementation of the application and hence he can automate the test cases accordingly.
5	Expectation	Expectation from Black box testing is that it makes a clear vision about the functionality of the application with external input parameters with its report.	While user does White box testing with the expectation that this testing will also check the quality and performance of the application by executing the code of the application.

10. Explain steps involved in CBT model construction and test preparation.

-Defining the model: These models are often represented by some graphs, with individual nodes representing the basic model elements and links representing the interconnections. Some additional information may be attached to the graph as link or node properties (commonly referred to as weights in graph theory).

-Checking individual model elements to make sure the individual elements, such as links, nodes, and related properties, have been tested individually, typically in isolation, prior to testing using the whole model. This step also represents the self-checking of the model, to make sure that the model captures what is to be tested.

-Dejning coverage criteria: Besides covering the basic model elements above, some other coverage criteria are typically used to cover the overall execution and interactions. For example, with the partition-based testing, we might want to cover the boundaries in addition to individual partitions; and for FSM-based testing, we might want to cover state transition sequences and execution paths.

-Derive test cases: Once the coverage criteria are defined, we can design our test cases to achieve them. The test cases need to be sensitized, that is, with its input values selected to realize specific tests, anticipated results defined, and ways to check the outcomes planned ahead of time