

**Q.1. Analyze the Significance of FTR and identify the objectives of FTR.**

**ANS :**

## **FTR(Formal Technical Review)**

- Formal technical review (FTR) is a software quality control activity performed by software engineers (and others).
- The objectives of an FTR are:
  - ✓ To uncover errors in function, logic, or implementation for any representation of the software;
  - ✓ To verify that the software under review meets its requirements;
  - ✓ To ensure that the software has been represented according to predefined standards
  - ✓ To achieve software that is developed in a uniform manner;
  - ✓ To make projects more manageable. In addition, the FTR serves as a training ground, enabling junior engineers to observe different approaches to software analysis, design, and implementation

- In addition, the purpose of FTR is to enable junior engineer to observe the analysis, design, coding and testing approach more closely. FTR also works to promote back up and continuity become familiar with parts of software they might not have seen otherwise.
- **The FTR is actually a class of reviews that includes walkthroughs and inspections.**
- Each FTR is conducted as meeting and is considered successful only if it is properly planned, controlled and attended.

- **The objective is to maximize productivity by minimizing mistakes (errors).**
- CM is used to essential due to the inventory management, library management, and updating management of the items essential for the project.

**Q.2. Inspect and explain the working of different layers of SCM.**

**Ans :**

## **SCM Process**

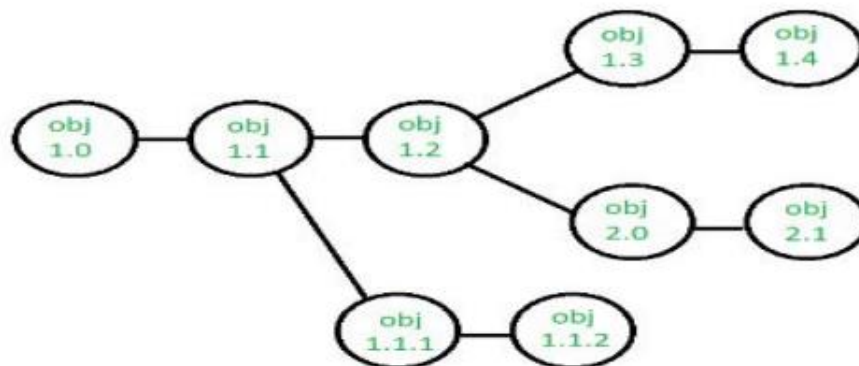
- It uses the tools which keep that the necessary change has been implemented adequately to the appropriate component. The **SCM** process defines a number of tasks:
- Identification of objects in the software configuration
- Version Control
- Change Control
- Configuration Audit
- Status Reporting



- **Identification**
- ✓ **Basic Object:** Unit of Text created by a software engineer during analysis, design, code, or test.
- ✓ **Aggregate Object:** A collection of essential objects and other aggregate objects. Design Specification is an aggregate object.
- Each object has a set of distinct characteristics that identify it uniquely: a name, a description, a list of resources, and a "realization."
- **The interrelationships between configuration objects can be described with a Module Interconnection Language (MIL).**

## Version Control

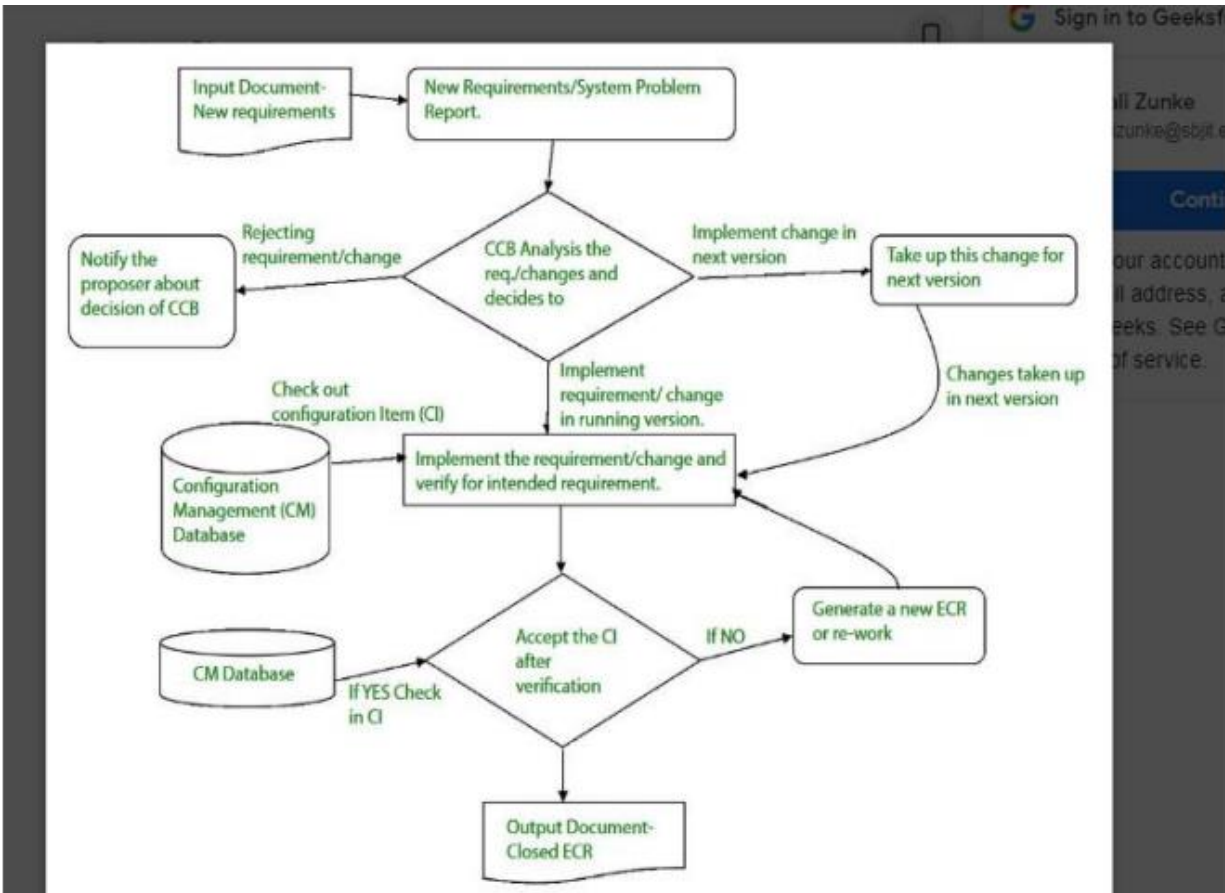
- Version Control combines procedures and tools to handle different version of configuration objects that are generated during the software process.
- Creating versions/specifications of the existing product to build new products from the help of **SCM** system.



# Change Control

- Controlling changes to Configuration items (CI).
- A change request (CR) is submitted and evaluated to assess technical merit, potential side effects, overall impact on other configuration objects and system functions, and the projected cost of the change.
- The results of the evaluation are presented as a change report, which is used by a change control board (CCB)—a person or group who makes a final decision on the status and priority of the change.
- An engineering change Request (ECR) is generated for each approved change.
- Also CCB notifies the developer in case the change is rejected with proper reason.
- The ECR describes the change to be made, the constraints that must be respected, and the criteria for review and audit.
- The object to be changed is “checked out” of the project database, the change is made, and then the object is tested again.
- The object is then “checked in” to the database and appropriate version control mechanisms are used to create the next version of the software.





## Configuration Audit

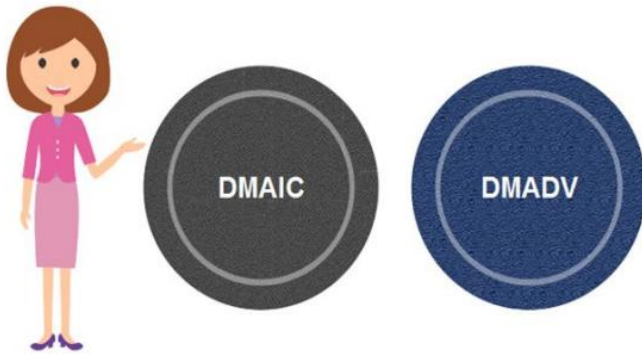
- A software configuration audit complements the formal technical review of the process and product. It focuses on the technical correctness of the configuration object that has been modified.
- The audit confirms the completeness, correctness and consistency of items in the **SCM** system and track action items from the audit to closure.

# Reporting

- Providing accurate status and current configuration data to developers, tester, end users, customers and stakeholders through admin guides, user guides, FAQs, Release notes, Memos, Installation Guide, Configuration guide etc .

**Q.3 Examine the working of Six Sigma standard for Software Quality Assurance .**

## Six Sigma Methodologies



## DMAIC

- It specifies a data-driven quality strategy for improving processes. This methodology is used to enhance an existing business process.



- **Define:** It covers the process mapping and flow-charting, project charter development, problem-solving tools, and so-called 7-M tools.
- **Measure:** It includes the principles of measurement, continuous and discrete data, and scales of measurement, an overview of the principle of variations and repeatability and reproducibility (RR) studies for continuous and discrete data.
- **Analyze:** It covers establishing a process baseline, how to determine process improvement goals, knowledge discovery, including descriptive and exploratory data analysis and data mining tools, the basic principle of Statistical Process Control (SPC), specialized control charts, process capability analysis, correlation and regression analysis, analysis of categorical data, and non-parametric statistical methods.

- **Improve:** It covers project management, risk assessment, process simulation, and design of experiments (DOE), robust design concepts, and process optimization.
- **Control:** It covers process control planning, using SPC for operational control and PRE-Control.

## DMADV

- It specifies a data-driven quality strategy for designing products and processes. This method is used to create new product designs or process designs in such a way that it results in a more predictable, mature, and defect free performance.



- **Define:** It defines the problem or project goal that needs to be addressed.
- **Measure:** It measures and determines the customer's needs and specifications.
- **Analyze:** It analyzes the process to meet customer needs.
- **Design:** It can design a process that will meet customer needs.
- **Verify:** It can verify the design performance and ability to meet customer needs.



**Q4 . Analyze the significance of following activities:**

**1) SQA**

**2) Software Reliability**

**ANS -**

## **Major Software Quality Assurance Activities**

- **SQA Management Plan:**  
Make a plan for how you will carry out the **sqa** through out the project. Think about which set of software engineering activities are the best for project. check level of **sqa** team skills.
- **Set The Check Points:**  
**SQA** team should set checkpoints. Evaluate the performance of the project on the basis of collected data on different check points
- **Multi testing Strategy:**  
Do not depend on a single testing approach. When you have a lot of testing approaches available use them.
- **Measure Change Impact:**  
The changes for making the correction of an error sometimes re introduces more errors keep the measure of impact of change on project. Reset the new change to change check the compatibility of this fix with whole project
- **Manage Good Relations:**  
In the working environment managing good relations with other teams involved in the project development is mandatory. Bad relation of **SQA** team with programmers team will impact directly and badly on project. Don't play politics.

# Software Reliability

- Software Reliability means **Operational reliability**. It is described as the ability of a system or component to perform its required functions under static conditions for a specific period.
- Software reliability is also defined as the probability that a software system fulfills its assigned task in a given environment for a predefined number of input cases, assuming that the hardware and the input are free of error.
- Software Reliability is an essential connect of software quality, composed with functionality, usability, performance, serviceability, capability, installability, maintainability, and documentation.
- Software Reliability is hard to achieve because the complexity of software turn to be high.
- While any system with a high degree of complexity, containing software, will be hard to reach a certain level of reliability, system developers tend to push complexity into the software layer, with the speedy growth of system size and ease of doing so by upgrading the software.
- **For example**, large next-generation aircraft will have over 1 million source lines of software on-board;
- next-generation air traffic control systems will contain between one and two million lines;
- the upcoming International Space Station will have over two million lines on-board and over 10 million lines of ground support software;
- several significant life-critical defense systems will have over 5 million source lines of software.
- While the complexity of software is inversely associated with software reliability, it is directly related to other vital factors in software quality, especially functionality, capability, etc.

## **Q5 . List and explain different types of Software Reviews**

### **ANS – Types of Software Reviews:**

There are mainly 3 types of software reviews:

#### **Software Peer Review:**

Peer review is the process of assessing the technical content and quality of the product and it is usually conducted by the author of the work product along with some other developers.

Peer review is performed in order to examine or resolve the defects in the software, whose quality is also checked by other members of the team.

#### **1. Peer Review has following types:**

##### **(i) Code Review:**

Computer source code is examined in a systematic way.

##### **(ii) Pair Programming:**

It is a code review where two developers develop code together at the same platform.

##### **(iii) Walkthrough:**

Members of the development team is guided by author and other interested parties and the participants ask questions and make comments about defects.

##### **(iv) Technical Review:**

A team of highly qualified individuals examines the software product for its client's use and identifies technical defects from specifications and standards.

##### **(v) Inspection:**

In inspection the reviewers follow a well-defined process to find defects.

## **2. Software Management Review:**

Software Management Review evaluates the work status. In this section decisions regarding downstream activities are taken.

## **3. Software Audit Review:**

Software Audit Review is a type of external review in which one or more critics, who are not a part of the development team, organize an independent inspection of the software product and its processes to assess their compliance with stated specifications and standards. This is done by managerial level people.

**Q6 . Explain following Quality Standards:**

**1. ISO 9000**

**2. CMMI**

## **ISO 9000 Certification**

- **ISO** (International Standards Organization) is a group or consortium of 63 countries established to plan and fosters standardization.
- **ISO** declared its 9000 series of standards in 1987. It serves as a reference for the contract between independent parties.
- The **ISO** 9000 standard determines the guidelines for maintaining a quality system.
- The **ISO** standard mainly addresses operational methods and organizational methods such as responsibilities, reporting, etc.
- **ISO** 9000 defines a set of guidelines for the production process and is not directly concerned about the product itself.



**ISO 9000 is a series of three standards:**



- **ISO 9001:** This standard applies to the organizations engaged in design, development, production, and servicing of goods. This is the standard that applies to most software development organizations.
- **ISO 9002:** This standard applies to those organizations which do not design products but are only involved in the production. Examples of these category industries contain steel and car manufacturing industries that buy the product and plants designs from external sources and are engaged in only manufacturing those products. Therefore, **ISO 9002** does not apply to software development organizations.
- **ISO 9003:** This standard applies to organizations that are involved only in the installation and testing of the products. For example, Gas companies.

# ISO 9000 Execution



- **Application:** Once an organization decided to go for **ISO** certification, it applies to the registrar for registration.
- **Pre-Assessment:** During this stage, the registrar makes a rough assessment of the organization.
- **Document review and Adequacy of Audit:** During this stage, the registrar reviews the document submitted by the organization and suggest an improvement.
- **Compliance Audit:** During this stage, the registrar checks whether the organization has compiled the suggestion made by it during the review or not.
- **Registration:** The Registrar awards the **ISO** certification after the successful completion of all the phases.
- **Continued Inspection:** The registrar continued to monitor the organization time by time.

# Advantages

- Business **ISO**-9000 certification forces a corporation to specialize in “how they are doing business”. Each procedure and work instruction must be documented and thus becomes a springboard for continuous improvement.
- Employees morale is increased as they’re asked to require control of their processes and document their work processes
- Better products and services result from continuous improvement process.
- Increased employee participation, involvement, awareness and systematic employee training are reduced problems.

# CMMI

- CMM Integration project was formed to sort out the problem of using multiple CMMs. CMMI product team's mission was to combine three **Source Models** into a single improvement framework for the organizations pursuing enterprise-wide process improvement.
- These three Source Models are –
- ✓ Capability Maturity Model for Software (SW-CMM) - v2.0 Draft C.
- ✓ Electronic Industries Alliance Interim Standard (EIA/IS) - 731 Systems Engineering.
- ✓ Integrated Product Development Capability Maturity Model (IPD-CMM) v0.98.

## **Objectives of CMMI :**

- Fulfilling customer needs and expectations.
- Value creation for investors/stockholders.
- Market growth is increased.
- Improved quality of products and services.
- Enhanced reputation in Industry.



## **CMMI Model – Maturity Levels :**

In CMMI with staged representation, there are five maturity levels described as follows :

- **Maturity level 1 : Initial**
  - processes are poorly managed or controlled.
  - unpredictable outcomes of processes involved.
  - ad hoc and chaotic approach used.
  - No KPAs (Key Process Areas) defined.
  - Lowest quality and highest risk.
- **Capability level 1 : Performed**
  - process performance may not be stable.
  - objectives of quality, cost and schedule may not be met.
  - a capability level 1 process is expected to perform all specific and generic practices for this level.
  - only a start-step for process improvement.
- **Capability level 2 : Managed**
  - process is planned, monitored and controlled.
  - managing the process by ensuring that objectives are achieved.
  - objectives are both model and other including cost, quality, schedule.
  - actively managing processing with the help of metrics.

- **Capability level 3 : Defined**
  - a defined process is managed and meets the organization's set of guidelines and standards.
  - focus is process standardization.
- **Capability level 4 : Quantitatively Managed**
  - process is controlled using statistical and quantitative techniques.
  - process performance and quality is understood in statistical terms and metrics.
  - quantitative objectives for process quality and performance are established.
- **Capability level 5 : Optimizing**
  - focuses on continually improving process performance.
  - performance is improved in both ways – incremental and innovation.
  - emphasizes on studying the performance results across the organization to ensure that common causes or issues are identified and fixed.