Q) What is SCM? Explain change control mechanism of SCM.

⇒ ① When we develop a software, the product undergoes many changes in their maintenance phase; these changes should be handled effectively.

② Several individuals (programs) works together to achieve these common goals.

③ Each person or program involved in software development creates several outputs called software configuration items (SCI).

④ These can include things like unfinished version of modules, test data used for debugging, or parts of the final product.

⑤ All the information created during the software development process is called software configuration.

⑥ As the software moves forward, the number of these configuration items (SCIs) increases quickly

⑦ These are handled and controlled by software configuration management.

⑧ Software configuration management is the discipline which:
   ⓐ identifies changes
   ⓑ monitors and controls the change.
   ⓒ Ensure the proper implementation of change made to the item.
   ⓓ Auditing and reporting to the change made.

⑨ The objective is to maximize productivity by minimizing mistakes (errors)

(10) Change control mechanism

```
                         → Initiate ①
                           change request
(6)
  Close
change request                              Impact      ②
                                            Assessments


Review effectiveness
& communicate
(5)                                                      
          ←  Implement          Approve/Deny
             change    ←──────   change proposal   ③
                ④
```
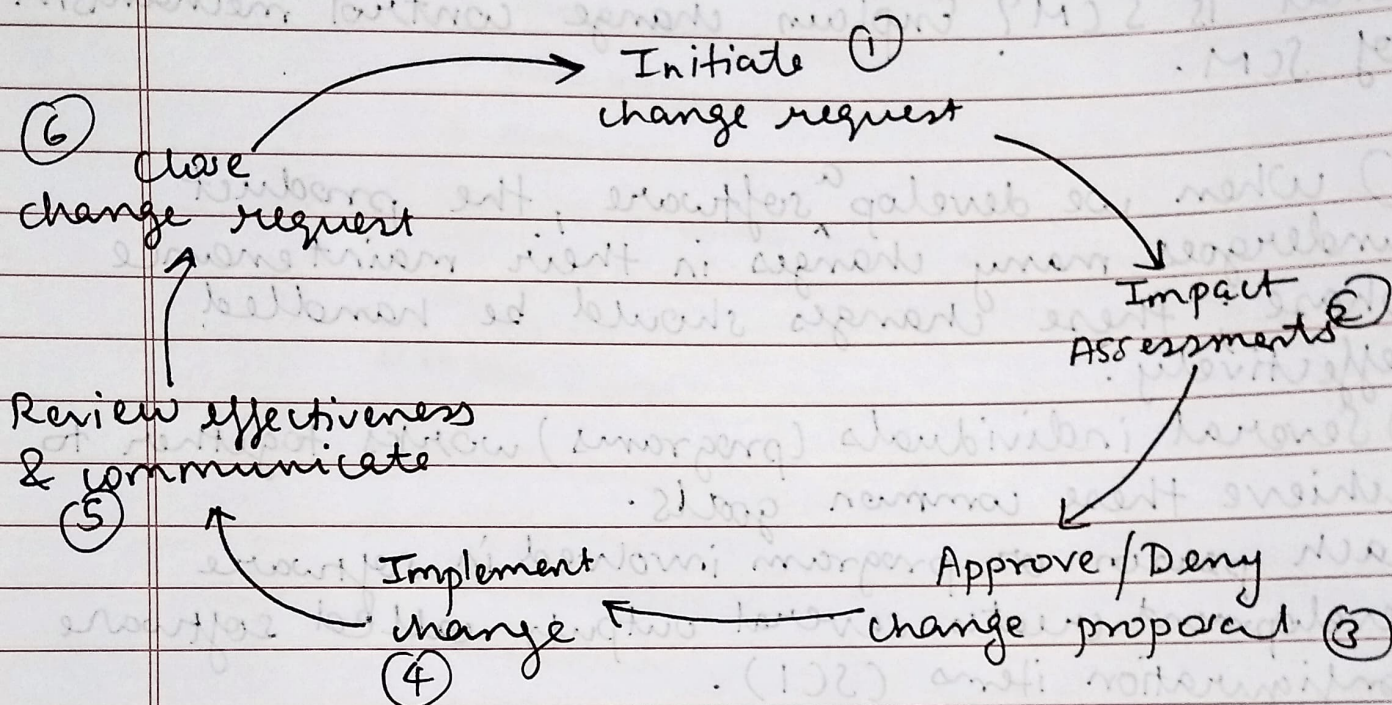
(i) A change request is submitted and reviewed
to see how it will affect the system and
what it will cost.

(ii) The results are shared with Change control
board (CCB) which decides if the changes
will be made.

(iii) If approved, an engeering change request (ECR)
is created; if rejected, the developer is told
the reason for rejection.

(iv) The ECR explains the change and how it will
be reviewed.

(v) The item is taken from the project database
changed and tested, then put back into the
database to create a new version.

Q) Risk in SE & Role of RMMM plan

=> ① Risk is an uncertain event with probability of occurence and potential for the loss.

② Source of risks

(i) Misunderstanding of the customer requirement

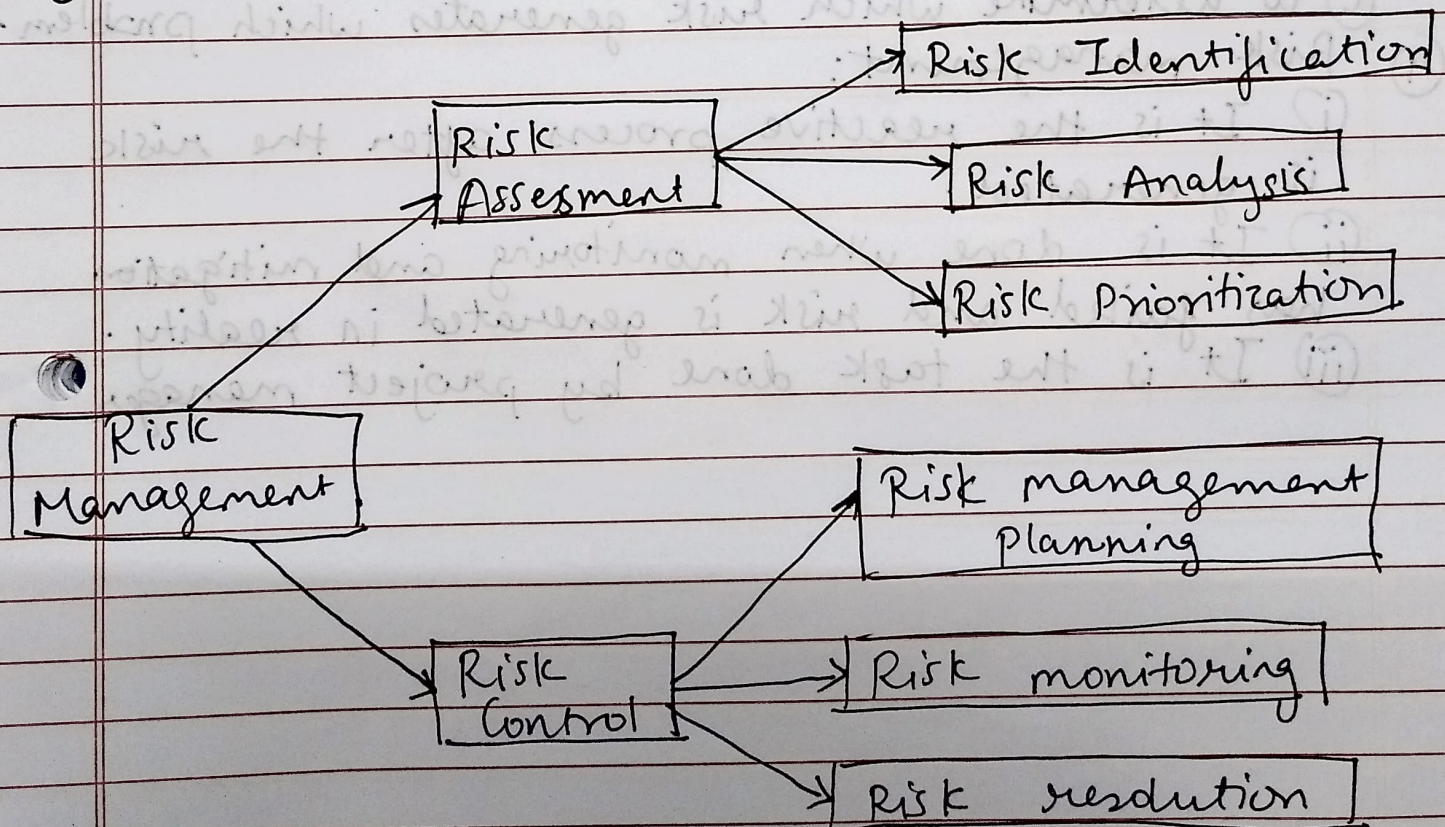(ii) No control and continuous changing of the customer requirement.

(iii) Unrealistic promises given to the customer.

(iv) Miscalculation of the software design

(v) Miscalculation of the team work.

(vi) Wrong budget estimation

③ Risk management activities

```
                                    ┌─────────────────────┐
                              ┌────→│ Risk Identification │
                              │     └─────────────────────┘
        ┌──────────┐          │     ┌─────────────────────┐
        │ Risk     │──────────┼────→│ Risk Analysis       │
        │ Assesment│          │     └─────────────────────┘
        └──────────┘          │     ┌─────────────────────┐
            ↑                  └────→│ Risk Prioritization │
            │                        └─────────────────────┘
            │
┌────────────┐                      ┌─────────────────────┐
│ Risk       │                 ┌───→│ Risk management     │
│ Management  │                │    │ Planning            │
└────────────┘                 │    └─────────────────────┘
      │                        │
      │      ┌──────────┐      │    ┌─────────────────────┐
      └─────→│ Risk     │──────┼───→│ Risk monitoring     │
             │ Control  │      │    └─────────────────────┘
             └──────────┘      │    ┌─────────────────────┐
                               └───→│ Risk resolution     │
                                    └─────────────────────┘
```

④ RMMM stands for Risk mitigation, monitoring and management.

⑤ The product manager typically uses RMMM plan.

⑥ Risk is document with the help of Risk Information Sheet (RIS)

⑦ Risk mitigation is an activity used to avoid problems.

Steps :
   ⓐ Finding out the risk.
   ⓑ Removing the causes for risk creation.
   ⓒ Preparing a document time to time.
   ⓓ Timely review to speed up the work.

⑧ Risk monitoring is an activity used for project tracking.

Steps :
   ⓐ To check risk has occured or not.
   ⓑ To gather the information for future risk assessment.
   ⓒ To determine which risk generates which problem.

⑨ Risk management :
   ⓘ It is the reactive process after the risk is generated.
   ⓘⓘ It is done when monitoring and mitigation has failed and risk is generated in reality.
   ⓘⓘⓘ It is the task done by project manager.

Module 6

Q1. Explain change control process in SCM in detail?

=>

a) Change Control is a structured method for managing all changes made to a product or system.

b) Its main goal is to avoid unnecessary changes, keep a record of each change, prevent interruptions and use resources effectively.

c) A six-step process used for handling a software change request:

1. Documenting the Change Request

i. When a client requests a change, it's recorded and categorized.

ii. This includes noting the importance of the change and how difficult it may be to make the change.

2. Formal Assessment

i. The reasons for the change are reviewed, and the risks and benefits of making the change are evaluated.

ii. If the change is approved, a team is assigned to work on it and if rejected, this is documented and shared with the client.

3. Planning

i. The team creates a detailed plan for designing and implementing the change.

ii. This plan also includes steps to undo the change if it doesn't work as expected.

4. Designing and Testing

i. The team designs the software update and tests it.

ii. If the update works well, they ask for approval and schedule a date for implementing it.

5. Implementation and Review

i. The team applies the change, and stakeholders review the results to see if it meets the intended goals.
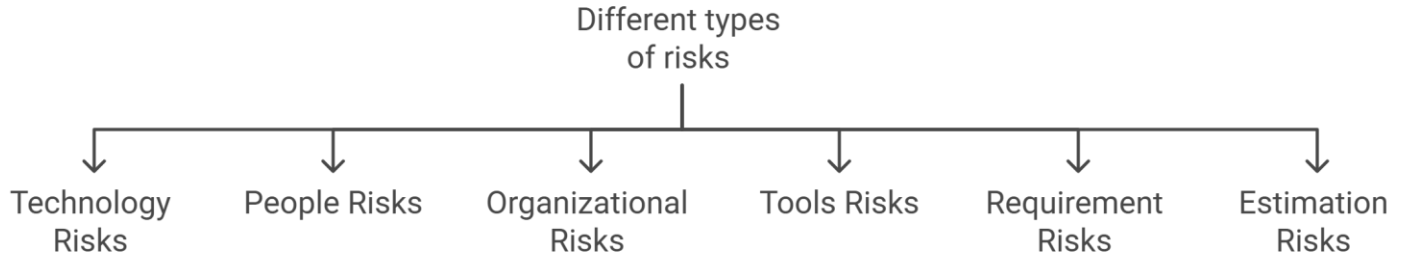
6. Final Assessment

i. If the client is satisfied with the change, the request is closed.

ii. If not, the project is reassessed, and some steps may be repeated to improve the outcome.

Yahape ek flowchart banaskte ho jo all 6 stages dikhata ho :

Documenting -> Formal Assessment -> Planning ....

Q2. Different types of risks

=>

Different types of risks → Technology Risks | People Risks | Organizational Risks | Tools Risks | Requirement Risks | Estimation Risks

1. Technology Risks

i) These are risks associated with the software or hardware technology being used to build the system.

ii) For example, if a project relies on a new programming language or framework that the team isn't familiar with, it might lead to delays or errors.

2. People Risks

i) People risks are related to the individuals working on the project.

ii) This could include risks like a lack of experience, team conflicts, or a key team member leaving unexpectedly.

iii) If a project depends heavily on one person's knowledge, their absence can create a problem.

3. Organizational Risks

i) Organizational risks come from the environment or company culture where the software is being developed.

ii) This could involve issues like lack of support from management, unrealistic deadlines, or frequent changes in project priorities.

iii) It can make it difficult for the team to focus and complete tasks on time.

4. Tools Risks

i) These are risks associated with the tools and software being used to develop the project.

ii) eg: A project team uses a code editor that keeps crashing, or a project management tool that doesn't sync properly, causing confusion and wasted time.

5. Requirement Risks

i) Requirement risks happen when there are frequent changes to the project requirements or unclear requirements from the client.

ii) This can make it difficult to plan effectively and can lead to rework, costing time and resources.

6. Estimation Risks

i) Estimation risks are related to managing time and resources.

ii) eg: The project manager estimates that a feature will take two weeks to complete, but the development team encounters unexpected challenges that stretch it to four weeks, impacting the project timeline.

Q3. What are the risk associated with software projects? how do project managers manage such risks?

=> *isme jo first question pucha hai uske liye Q2 waala answert refer kro*

Isme jo second question pucha hai uske liye RMMM ka answer likhdo

Q4. Prepare RMMM plan for the identified risk "Team members will leave the project in between the schedule"

=>

   a) Mitigation (Avoiding the Risk)
      i) Take proactive steps to reduce the chances of team members leaving.
      ii) For example, team members can train each other on key tasks so that at least two people know the important parts of the project.
      iii) Additionally, encouraging everyone to create clear, detailed documentation of their work makes it easier for others to step in if needed.
      iv) Regular check-ins with team members can also help maintain a positive work environment by addressing any issues early.

   b) Monitoring
   i) Keep an eye out for signs that a team member might be considering leaving, and track team morale.
   ii) Regular feedback sessions, like monthly or bi-weekly check-ins, give team members a chance to discuss concerns and help managers spot potential issues early.
   iii) Also, watching for changes in productivity or engagement can highlight if someone might need support.

   c) Management
   i) If a team member does leave, act quickly to minimize the impact.
   ii) Redistribute their tasks to other team members right away to keep things moving smoothly.
   iii) Using detailed documentation and existing team knowledge can help a new team member get up to speed quickly.
   iv) If losing a team member affects the schedule, communicate with stakeholders and, if necessary, adjust deadlines or reassign resources to stay on track.

Q5. Mention reasons for project delay. what are the risk associated with project delay ? perform risk assessment & prepare RMMM plan for the same.

=>

Reasons for Project Delay

1. **Unclear Requirements**: When project goals or client needs aren't clear, it leads to confusion and wasted time.
2. **Scope Creep**: Adding extra tasks or features mid-project adds work and can slow progress.
3. **Resource Shortages**: Not having enough team members, tools, or support can cause delays.
4. **Technical Issues**: Problems with software, tools, or technology create setbacks.

Risks Associated with Project Delay

1. **Higher Costs**: The longer the project takes, the more it costs in terms of labor and materials.
2. **Lower Client Satisfaction**: Clients may become unhappy with delays, which can harm relationships.
3. **Quality Problems**: Rushing to catch up on delays may lower the quality of the final product.

Risk Assessment

Risk	Likelihood	Impact	Overall Risk
Higher Costs	High	High	High
Lower Client Satisfaction	Medium	High	High
Quality Problems	High	Medium	Medium

RMMM Plan for Project Delay

1. **Mitigation (Avoiding Delays)**
   a. Get a solid understanding of the project goals with clients right from the start.
   b. Set a process for adding only essential new features to avoid scope creep.

2. **Monitoring**
    a. Check progress weekly to catch any potential delays early.
    b. Make sure the team has what they need to stay on schedule.

3. **Management**
    a. If delays happen, bring in extra help or adjust the team to speed up progress.
    b. Keep clients updated on any schedule changes and discuss realistic solutions if needed.

Q6. Identify risk for you final year examination & prepare RMMM plan.

=>

Risk for Final Year Examination

1. **Poor Time Management**: Struggling to balance study time, assignments, and personal activities or delaying study until the last minute.
2. **Lack of Preparation**: Incomplete or disorganized study notes, not fully understanding key topics, or not practicing enough.
3. **Health Issues**: Falling sick close to or during exams due to stress, poor sleep, or not taking care of health.
4. **High Stress Levels**: Feeling overwhelmed with pressure, leading to burnout, lack of focus, or anxiety.

RMMM Plan for Final Year Examination

1. **Risk Mitigation (Avoiding Risks)**
   a. **Create a Study Schedule**: Plan out study sessions for each subject, leaving time for review and breaks to avoid cramming.
   b. **Organize Study Materials**: Gather all notes and resources early, and keep them organized to make study sessions easier and more productive.
   c. **Take Care of Health**: Focus on healthy habits like balanced meals, regular sleep, and short exercise routines to keep energy up.
   d. **Practice Stress Management**: Use relaxation techniques like deep breathing, stretching, or meditation to keep stress manageable.

2. **Risk Monitoring**
   a. **Track Study Progress**: Regularly review if you're on track with your study plan and adjust if needed.
   b. **Monitor Understanding of Topics**: Check your understanding of key topics by doing practice questions or explaining concepts to someone else.
   c. **Observe Health and Energy Levels**: Pay attention to any signs of tiredness or illness and take breaks or rest when needed.
   d. **Gauge Stress Levels**: Be aware of stress indicators like difficulty concentrating or feeling overwhelmed, and take small breaks to recharge.

3. **Risk Management**
    a. **Adjust Study Plan if Needed**: If you're behind, focus on high-priority topics and streamline your study sessions for effectiveness.
    b. **Seek Help on Tough Topics**: Reach out to teachers, classmates, or use online resources to understand difficult subjects better.
    c. **Take Extra Rest if Unwell**: If feeling unwell, take some time to rest and recover, even if it means a short break from studying.

Q7. Version Control

=>

1) Version control is a process and set of tools used to manage different versions of files and documents created during software development.
2) It helps keep track of changes and allows teams to work together on the same project efficiently.
3) A version control system usually has four main features:
    i) **Project Database**: A storage place where all project files are kept and organized.
    ii) **Version Management**: Tracks all versions of each file, so you can go back to previous versions if needed.
    iii) **Make Facility**: A tool that helps gather all necessary files to create a working version of the software.
    iv) **Build a Specific Version**: Allows the developer to put together a specific version of the software by collecting the right files.

4) Many version control systems use something called a **"change set"**, which is a collection of all changes made to the project files.
5) Each change set includes details like why changes were made, who made them, and when they happened.
6) These change sets allow developers to build specific versions of the software by applying selected changes.
7) To make this process work, a system model is used, which includes:
    i) **Template**: A guide that shows how the components are arranged and in what order they need to be built.
    ii) **Construction Rules**: Steps on how to build the software correctly.
    iii) **Verification Rules**: Checks to make sure the software was built properly.
8) This approach helps developers organize, track, and manage changes efficiently, making it easier to create and update software.

Q8. What are software Quality Concepts ? Explain different Software Quality assurance Metrics in detail

1. Quality Assurance (QA):

   i)   QA ensures that the software system meets specific requirements and customer expectations.

   ii)   It defines clear standards and methods to guide the development process.

   iii)   QA focuses on important qualities like correctness, efficiency, flexibility, maintainability, portability, and usability.

2. Quality Control (QC):

   i)   QC aims to meet the quality standards defined by the customer's requirements.

   ii)   It ensures that the final product is delivered on time and within budget.

   iii)   QC checks if the product has achieved the expected level of quality before it reaches the customer.

3. Quality Planning:

   i)   Quality planning involves selecting the right procedures and standards for a particular project.

   ii)   This plan can be customized as needed to guide the development process effectively and ensure high-quality output.

Software Quality Metrics

Software Quality Metrics (SQM) help measure various aspects of software quality to make sure the product is reliable, functional, and satisfies customer expectations.

1. Customer Problem Metrics:

   i)   Measures the problems customers encounter while using the product.

   ii)   **Formula:** Problems per User Month (PUM) = Total problems reported by a customer / Total number of license months.

2. Customer Satisfaction Metrics:

i)   Assesses overall product quality based on customer feedback and satisfaction.
ii) Satisfaction is typically measured through ratings like Very Satisfied, Satisfied, Neutral, Dissatisfied, or Very Dissatisfied.

3. Software Maintenance Metrics:

i)   Tracks defects that appear after the product has been released to customers.
ii) Helps evaluate the quality of the development and testing processes by looking at the number and severity of issues found in the customer environment.

Q9. Explain SQA in detail with the steps involved in SQA Plan?

=>

Software Quality Assurance (SQA)

1. Software Quality Assurance (SQA) is a process to ensure that software meets quality standards, follows industry guidelines, and fulfills user needs.
2. SQA focuses on identifying and fixing errors early in development.

Key SQA Activities

1. **Standards Compliance**: Ensures software follows industry standards (e.g., IEEE, ISO).
2. **Reviews and Audits**:
   a. *Reviews*: Conducted by developers to find early errors.
   b. *Audits*: Performed by SQA teams to confirm quality processes are followed.
3. **Testing**: Planned testing activities to catch errors and confirm functionality.
4. **Change Management**: Manages changes to avoid introducing new errors.

Steps in the SQA Plan

1. **Create an SQA Management Plan**: Outline quality objectives, roles, and responsibilities.
2. **Set Quality Checkpoints**: Define checkpoints to assess quality at different stages.
3. **Measure Impact of Changes**: Track changes to ensure fixes don't introduce new issues.
4. **Use a Multi-Testing Strategy**: Apply various tests (e.g., unit, integration, system tests).
5. **Document Records and Reports**: Keep records of tests, errors, changes, and solutions for reference.