

1. Software modeling and implementation.

Process Planning.

Before starting a Software Project, it is essential to decide which tasks have to be performed and how to manage all tasks involved in the Software development.

- Project planning is an organized process from requirement gathering to testing and support.
- Focuses on all activities required for successful completion of the project.
- Prevents issues that arise in the project such as changes in projects, organization's objectives, non-availability of resources and so on.
- Helps in better utilization of resources and optimal usage of the allotted time for a project.
- Defines roles and responsibilities of the project management team members.

Project Planning Processes:

1. Stakeholders needs
2. Project objectives
3. Deliverables & Due Dates
4. Project Schedule
5. Roles & Responsibility
6. Project Budget
7. Communication plan
8. Tracking & Management.

2 Effort Estimation.

It is the process of predicting the most realistic amount of effort required or maintain software based on incomplete and uncertain input.

Cost & time required to complete development is estimated.

effort will depend upon productivity.

why Software effort estimation is important

1. Organizations have proper control over project and they can plan systematically.
2. Identification of resources
3. Having a clear understanding of product.
4. Budget and time constraint estimation
5. Impact of risks, necessary to validation

There are two types :-

Top-Down-Approach :- Effort Function depends on size. If size is larger more efforts are required.

Bottom-Up- Approach :-

- Classify the modules into simple, medium and complex
- Find the coding effort on an average
- Get total effort
- Estimate effort for remaining tasks
- Using specific factors refine estimates.

3 Project Scheduling and Staffing.

Project Scheduling :- refers to roadmap of all activities to be done with specified order and within time slot allotted to each activity. It's a work of Project manager.

~~For Scheduling~~

For scheduling a project, it is necessary to :-

1. Breaks down the project into smaller manageable form.
2. Find out the various tasks and correlate them.
3. Estimate time frame required for each tasks.
4. Divide time into work units.
5. Assign ~~Specified~~ number of work units for each task.
6. Calculate the total time required for the project from starting to finish.

Project Staffing :- After planning and scheduling, the next step in the management process is staffing. Staffing means assigning people to jobs. In other words, it is the process of management which is concerned with obtaining, utilizing and maintaining a satisfied work force.

Importance of staffing.

1. Helps in discovering and obtaining competent personnel for various jobs.
2. Provides higher performance of job.
3. Ensures the continuous survival and growth of the enterprise.
4. Optimum utilization of the human resources.

4. It avoids disruption of works by indicating in advance with shortage of personal.
5. Improves job satisfaction and morale of employees by giving fair reward for their contribution.

✓ 4 Software Configuration Management.

Software Configuration Management (Scm) is defined as a process to systematically manage, organize, and control the changes in the documents, codes, and other entities during the Software development Life Cycle.

Primary goal is to increase the productivity with minimal mistakes.

why do we need Configuration management-

- 1.) There are multiple people working on software which is Continually Updating.
- 2.) Changes in user requirement, policy, update, budget, Schedule needs to be accommodated.
- 3.) Software should be able to run on various machines and Operating System.
- 4.) Helps to develop co-ordination among stakeholders.
- 5.) beneficial to Control Cost involved in making changes to a system.

SCM defines a number of tasks :-

- Software Configuration Management
- Version Controlling to maintain of objects
- Change Control process managing a project
- Configuration auditing for maintaining quality
- Status Reporting.

1. Version Control :- Creating versions / Specifications of the new product to build new Product from the help of SCM System.

2. Change Control :- Is a Systematic approach to managing all changes made to a Product or System.

3. System Status Reporting :- Providing accurate status and current configuration data to developers, testers, end users.

4. Configuration Audit :- Verify that all the Software Product satisfies the baseline needs.

5 Quality Planning.

The quality of a product is initially created during design in the form of particular design.

SQA is a process ensuring Software Products meet quality standards and requirements.

• Objective: Prevent defects, improve quality, and ensure customer satisfaction.

• Process: Implements quality control and management activities throughout development.

• Techniques: Uses code reviews, inspections, audits and testing to evaluate Software quality.

• Quality Attributes: Focuses on reliability, maintainability, usability, efficiency and functionality.

• Continuous Improvement: monitors processes and Product to identify and implement improvements.

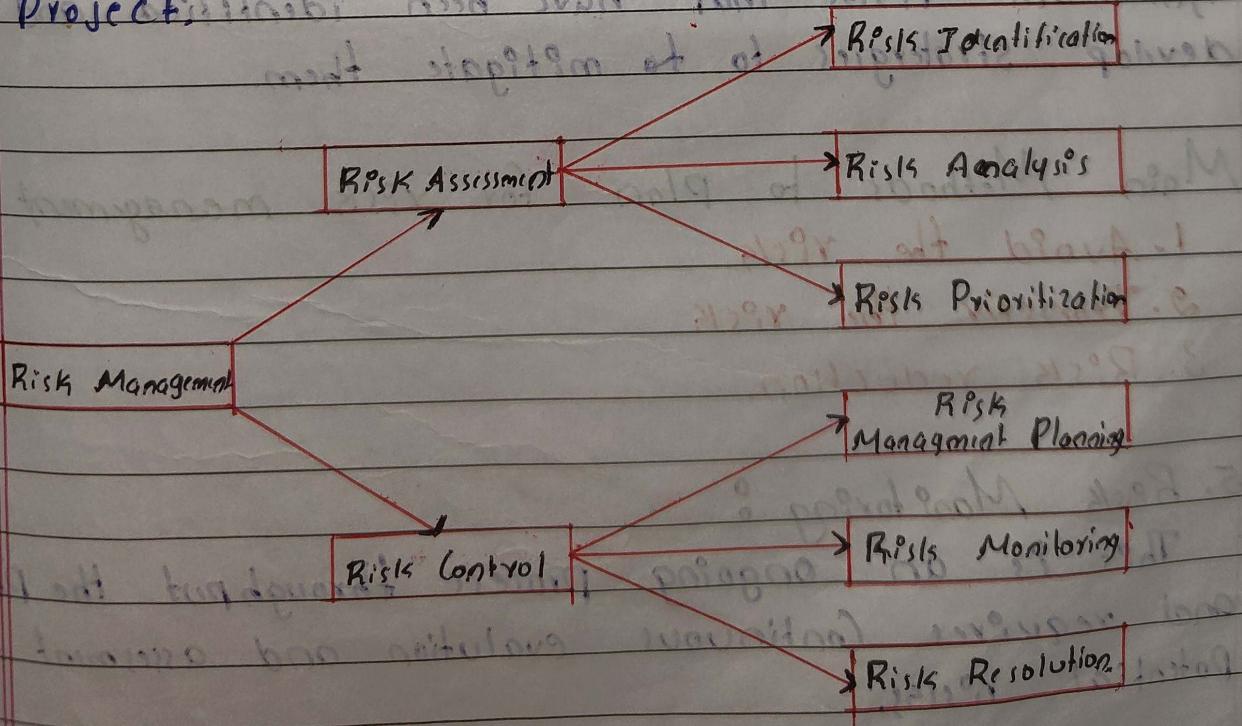
6 Risks Management

Risk is an uncertain future event with a probability of occurrence and potential for loss.

> Sources of Risk :

1. Misunderstanding of customer requirements.
 2. Uncontrolled & Continuous Changing of Customer requirements.
 3. Unrealistic Promises given to customers.
 4. Misunderstanding of real impact of new methodologies
 5. wrong budget estimation

- Risk Management is an important part of Project planning activities
 - It involves identifying and estimating the probability of risks along with their order of impact on the project activities and need for risk mitigation.



1. Risk Identification:

- The Project Organiser needs to find out risks in the project as early as possible.
- So, the impact of risks can be reduced by making effective risk management planning.

2. Risk Analysis & Prioritization:

- Identifying the problems causing risks.
- Identifying the probability of occurrence of problem.
- Identifying the impact of the problem.

3. Risk Control:

It is the process of managing risks to achieve desired outcomes.

4. Risk Planning:

Risk planning techniques consider all of the significant risks that have been identified and develop strategies to mitigate them.

Main Methods to plan for risk management:

1. Avoid the risks.
2. Transfer the risks.
3. Risk reduction.

5. Risk Monitoring:

This is an ongoing process throughout the project and requires continuous evaluation and assessment of potential risks.

6. Risk Resolution:

This process ensures that the project stays on track and risks are controlled within acceptable levels.

The effectiveness of risk resolution depends on the accuracy of risk identification, analysis and planning of risk solving.

7 Project Monitoring:

Project monitoring is an ongoing project that helps track progress, control processes, observe, and measure the project's performance, and progress against its previously defined objectives and plans.

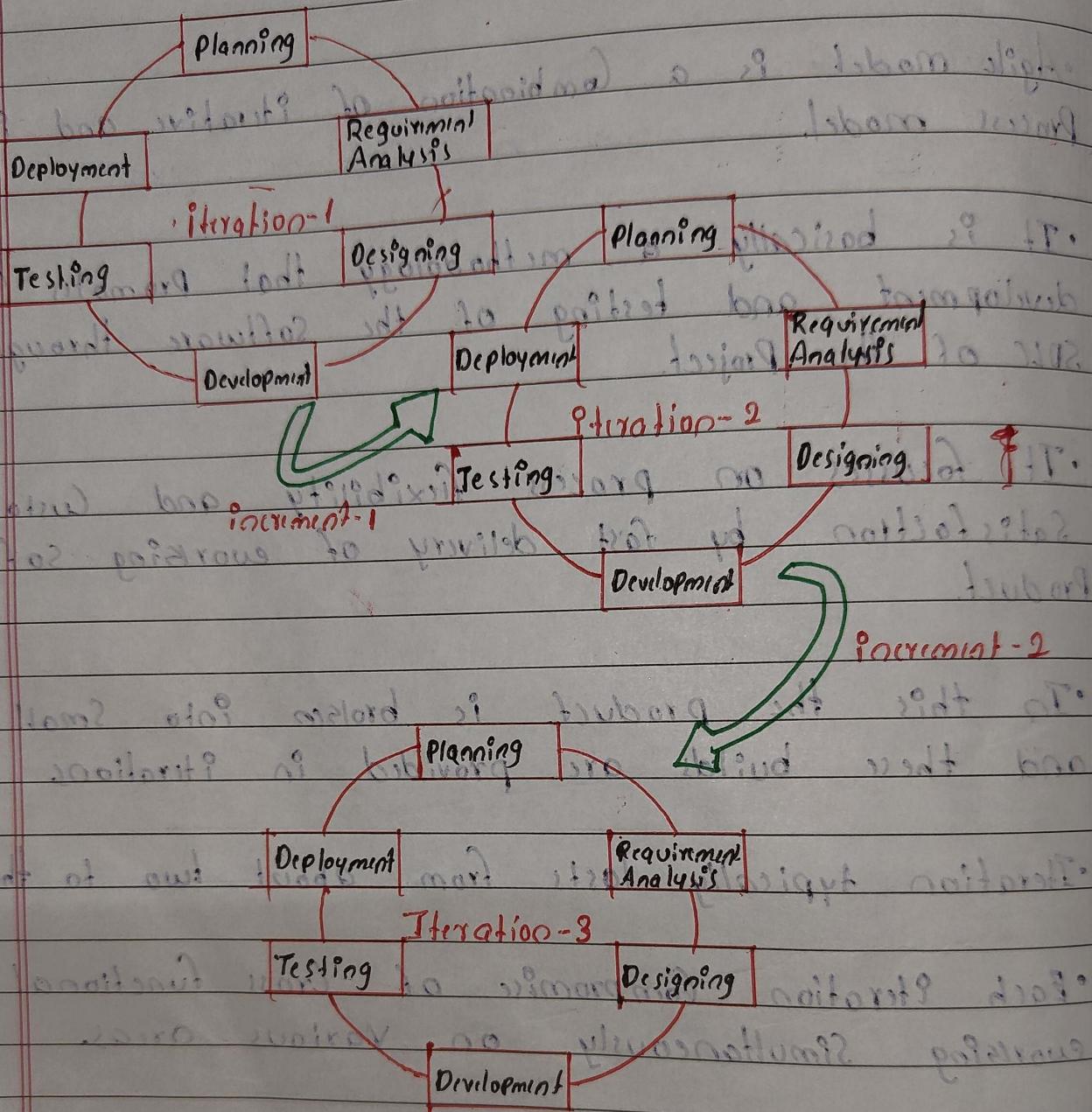
Project monitoring is like keeping an eye on a project to make sure everything is going according to plan.

Importance of Project Monitoring:

1. **Stays on tracks:** Ensures the project progresses at planned and meets deadline.
2. **Quality Assurance:** Helps to maintain the quality of the software by catching issues early.
3. **Budget Control:** Keeps Spending Under Control, Preventing Cost Overruns.
4. **Improves Communication:** Keeps everyone informed about the project's status.
5. **Goal Alignment:** Ensures the project stays aligned with its original goals and objectives.

✓ 8 Agile Model

- Agile model is a combination of iterative and incremental process model.
- It is basically a methodology that promotes continuous development and testing of the software throughout the SDLC of the Project.
- It focusses on process, flexibility and customer satisfaction by fast delivery of working software product.
- In this the product is broken into small builds and these builds are provided in iterations.
- Iteration typically lasts from about two to three weeks.
- Each iteration comprise of cross functional teams working simultaneously on various areas.
- Each build is incremental in terms of features and the final build holds all the features required by the customer.
- A working product is provided to the customer and important stakeholders at the end of the each iteration.



- ### Types of Agile Method
- Scrum
 - eXtreme Programming (xp)
 - Crystal
 - Dynamic Software Development Method (DSDM)
 - Feature Driven Development (FDD)
 - Lean Software Development

Advantages of Agile Model

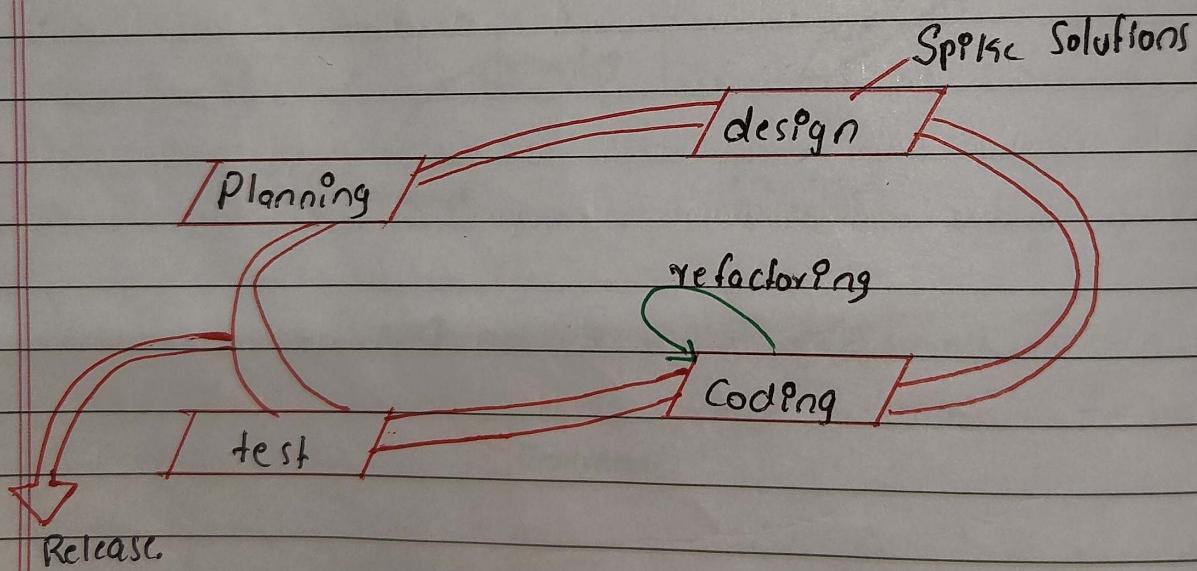
- Fast Delivery
- Face to Face communication with Clients
- Efficient design Fast Feedbacks
- Improves Quality.

Disadvantages of Agile Model

- Less Documentation
- Requires good collaboration between team members
- Client oriented team.

9 Extreme Programming

- XP is ~~fast~~ lightweight, efficient, low-risk, flexible, predictable, scientific and fun way to develop a Software.
- Small to medium sized teams that work under vague and rapidly changing requirements.
- The five values of XP are communication, simplicity, feedback, courage, and respect.
- Follows object-oriented approach.



Simple Solutions :- A Simple Solution is a very simple program to explore.

Refactoring :- Improves the internal structure of the code but external behaviour not affected.

Advantages

1. Fewer documentation required
2. Collaboration with customer via video conference
3. Flexibility for developers regarding choice of language
4. Easy to manage communication between developer and client

Disadvantages

1. Depends heavily on customer collaboration
2. Transfer of technology to new team members may be quite challenging.

10 Agile Project Management.

Agile Project Management is a way to manage projects that focusses on flexibility, collaboration and delivering small pieces of work frequently. It's designed to handle changes and ensure that the final product meets the customer's needs.

Key Concepts of Agile Project management.

1. Flexibility
2. Small, Frequent Deliverables
3. Collaboration
4. Continuous Improvement

Benefits of Agile Project Management.

Best Quality: Frequent testing and feedback help to catch issues early, leading to higher-quality products.

Customer Satisfaction: Regular deliveries and ongoing involvement ensure that the final product meets the customer needs.

Adaptability: The team can respond to changes quickly.

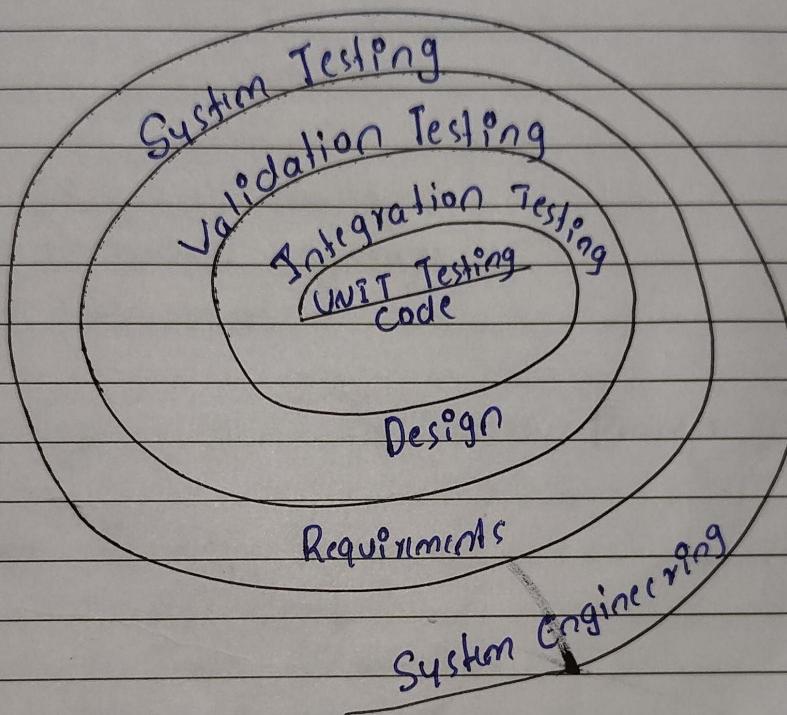
Improved Team Morale: Clear goals, regular progress, and collaboration can lead to a more motivated and engaged team.

Unit - IV

Software Quality Management and Advanced Topics.

1 Software Quality Management Process - Test Strategies.

1. Software Testing Strategies.



- The Software is developed by moving ^{Inwards} ↑ in the Spiral
- Testing is Conducted by moving ^{Outward} ↓ in the Spiral

1. System Engineering :- defines Testing Individual parts of the Software , like functions or methods , to ensure they works correctly on their own

2. Integration Testing :- Components are integrated together Step by Step to form a Complete Software

3. Validation Testing :- Requests Requirements established are validated against the developed software.

4. Unit testing :- focusses on testing each module / component independently based on implementation.

5. System Testing :- Testing the entire system as a whole to ensure it implements the specified requirements.

2 Software Process Improvement.

Software Process Improvement is a set of activities that will lead to better Software Process and in-built result higher-quality Software delivered in a more timely manner.

There are three groups

- technical managers
- Software engineers
- Individuals

Advantages of Software Process Improvement

- Higher Quality Software.
- Increased Efficiency
- Improved Team Collaboration
- Better Risk Management
- Enhanced Customer Satisfaction.

Disadvantages of Software Process Improvement

- Initial Cost and Time Investment
- Resistance to Change
- Overhead Complexity
- Not Always Visible Results.
- Potential for Misalignment.

3 Software Security and Cyber Security.

Software Security

- Focuses on individual software applications, code, and data.
- Ensuring the security of software applications and data means protecting them from maliciousness.
- Focuses on writing safe code, checking for maliciousness, controlling access, using encryption.
- Part of the software development process.
- Handles software specific issues during incident response.
- Developers work on secure coding and protecting applications.
- Makes sure software meets security standards.

Cyber Security

- Cyber Security covers a wider range of digital assets, including network systems, data and user training.
- Protecting an organization's entire digital ecosystem against a wide array of cyber threats.
- Covers network, device, data protection, handling security incidents and managing user access.
- Combines different security measures for overall protection.
- Deals with security issues across the whole organization not just software.
- Trains employees on various security topics, including software security.
- Ensures the organization follows laws and industry rules.

4 Software Security Importance.

Software Security involves designing and implementing software applications to protect against threats, vulnerabilities and malicious attacks.

Importance of Software Security.

1. Protection of Sensitive Data
2. Maintaining Trust and Reputation
3. Compliance with Regulations
4. Preventing Financial Loss
5. Avoiding Operational Disruptions
6. Protecting National Security
7. Mitigating Risks of Cyber Attacks
8. Ensuring Software Integrity

1. Protection of Sensitive Data.

Securing Software prevents unauthorized access to sensitive data such as personal information, financial records and business information.

2. Maintaining Trust and Reputation

Organizations with secure software build and maintain trust with their customers and partners.

3. Compliance with Regulations.

Many industries are subject to strict regulatory requirements regarding data protection and security.

4. Preventing Financial Loss
Security breaches can result in substantial financial loss due to fraud, theft of intellectual property.

5. Avoiding Operational Disruptions

Security incidents can disrupt normal business operations, affecting productivity.

6. Protecting National Security

In cases of software used by government or critical infrastructure, ensuring security is vital for national security.

7. Mitigating Risks of Cyber Attacks

With the increasing sophistication of cyber attacks, robust software security measures are essential to protect against threats.

