

Shrinivas Zend

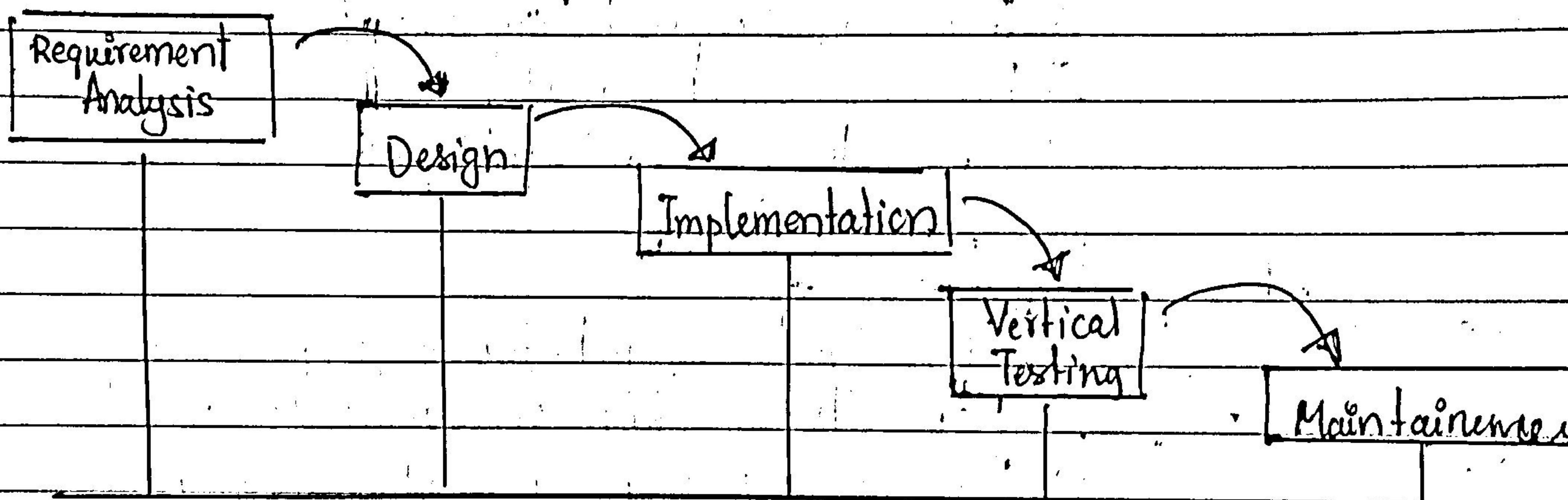
T2-3

2201125.

Assignment 1.

Q.1. Explain Software development models.

Ans. **Waterfall Model :** It is the first approach used in software development model.



It is also called as classical life cycle model or linear sequential model. In Waterfall model, any phase of development process only begins if previous phase is done.

a) Requirement Analysis — All business requirements of system are gathered and analysed by communication between stakeholders & managers. At the end of this phase, RSD is created.

b) Design — Based on RSD, design of systems is created called Software architecture. It is the blueprint of system.

c) Implementation — Actual code is created for software architecture using hardware & software requirements of system.

d) Verification / Testing — Code is ready, but it is verified against requirements of user in order to ensure software satisfies all business requirements.

e) Maintenance — While using software if user faces some issues, then those problems must be solved time to time.

Advantages —

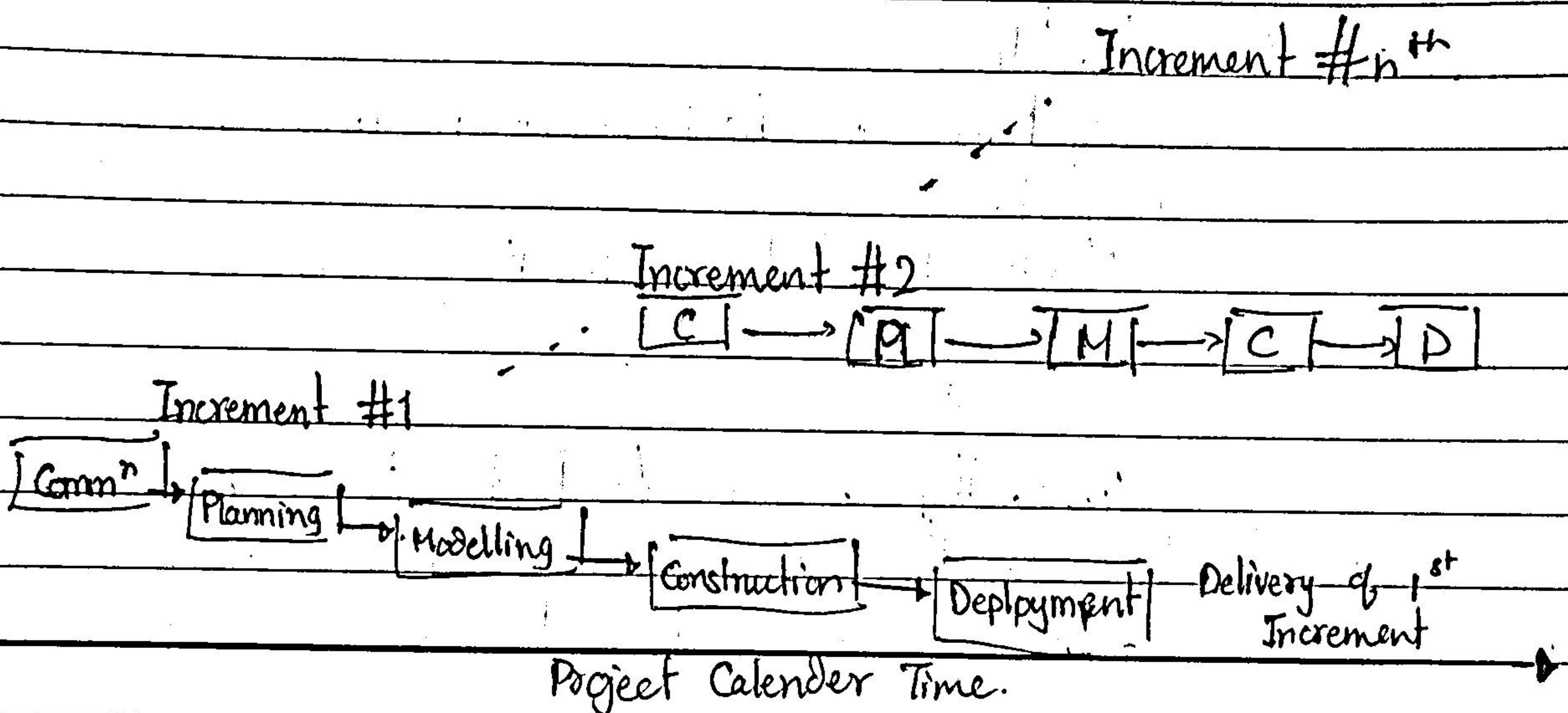
- 1) It is very simple to implement.
- 2) Phases do not overlap.
- 3) Easy to manage development phase.

Disadvantage —

- 1) It is not useful for large projects.
- 2) Not suitable when requirements aren't clear.
- 3) Very difficult to modify system requirements in middle of development process.

II) Incremental Model : The incremental model applies the waterfall model incrementally. The series of releases is referred to as 'increments', with each increment providing more functionality to customers.

Software functionality & features.



After the 1st increment, a core product is delivered, which can already be used by customer. Based on customer feedback, a plan is developed for the next increments, and the modifications are made accordingly. This process continues, with increments being delivered until complete product is delivered. The phases are :

- a) Communication — Helps to understand the objective
- b) Planning — Required as many people work on the same project but different functions at same time.

- c) Modelling — Involves business modelling, data modelling & process modelling.
- d) Construction — This involves the reuse of software components & automatic code.
- e) Deployment — Integration of all increments.

- Characteristics :

- 1) System is broken down into many mini development projects.
- 2) First tackles highest priority requirements.

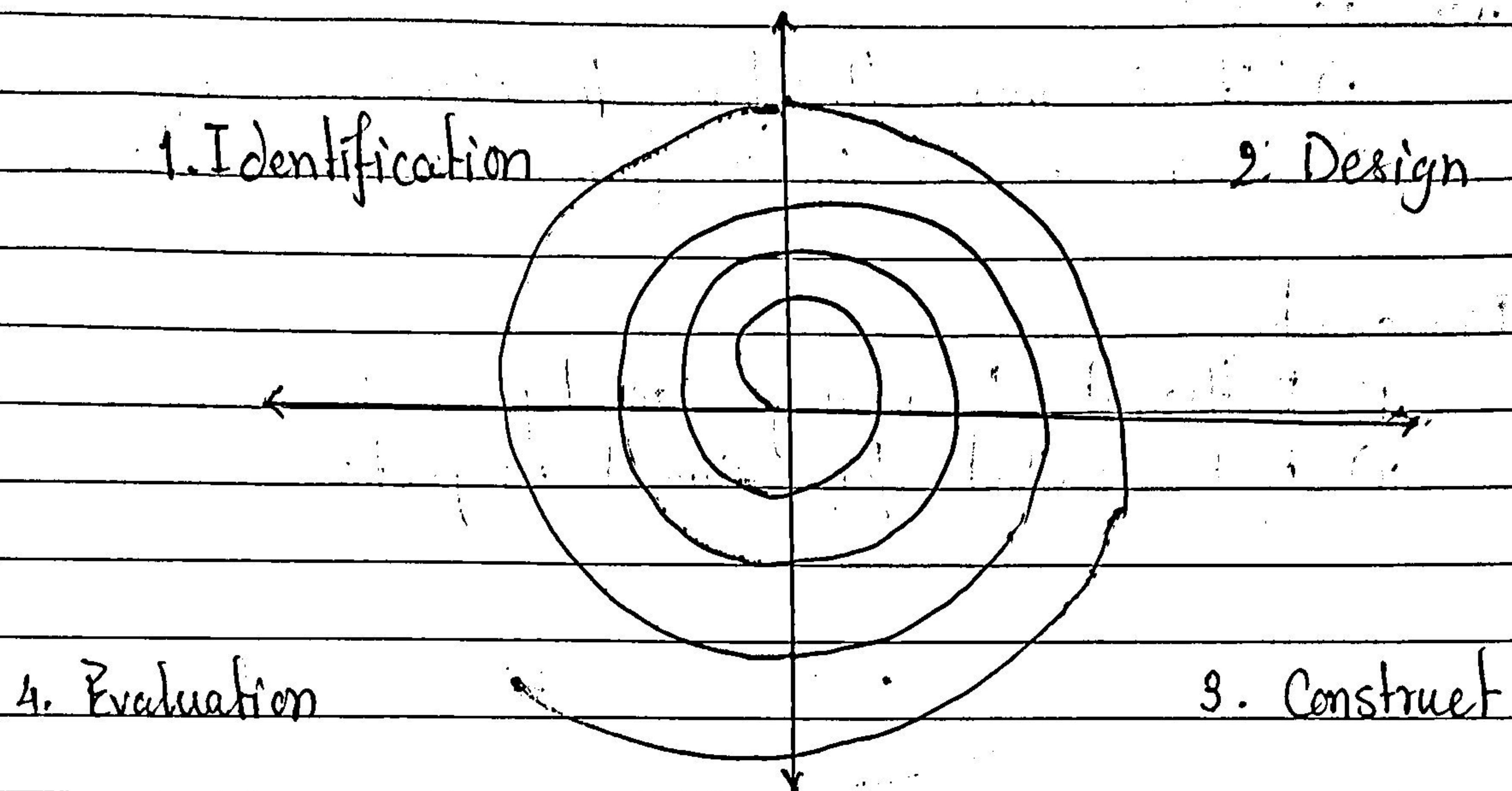
- Advantages :

- 1) Generally easier to test & debug because relatively smaller changes are made during each iteration.
- 2) Initial product delivery is faster & cheaper.

- Disadvantages :

- 1) Resulting cost may be exceeding the cost of org.
- 2) Problems may arise related to system architecture which were not evident in earlier prototype.

III) Spiral Model : Spiral model is a combination of iterative model & waterfall model. Spiral model has four phases of environment development, each of these phases is called as (Spiral).



A) Identification — This phase identifies all business requirements of system at beginning. It involves clear understanding of requirements by communicating between stakeholders & customer.

B) Design — This phase develops conceptual designs of system based on initially gathered requirements. In further spirals, it develops logical design of physical design.

C) Construct — This phase develops a code for conceptual design to get user feedback. In next subsequent spirals, detailed working model of software is constructed with increment.

D) Evaluation of Risk analysis — In this phase management risks like cost overrun are identified & monitored.

(2)

Advantages :

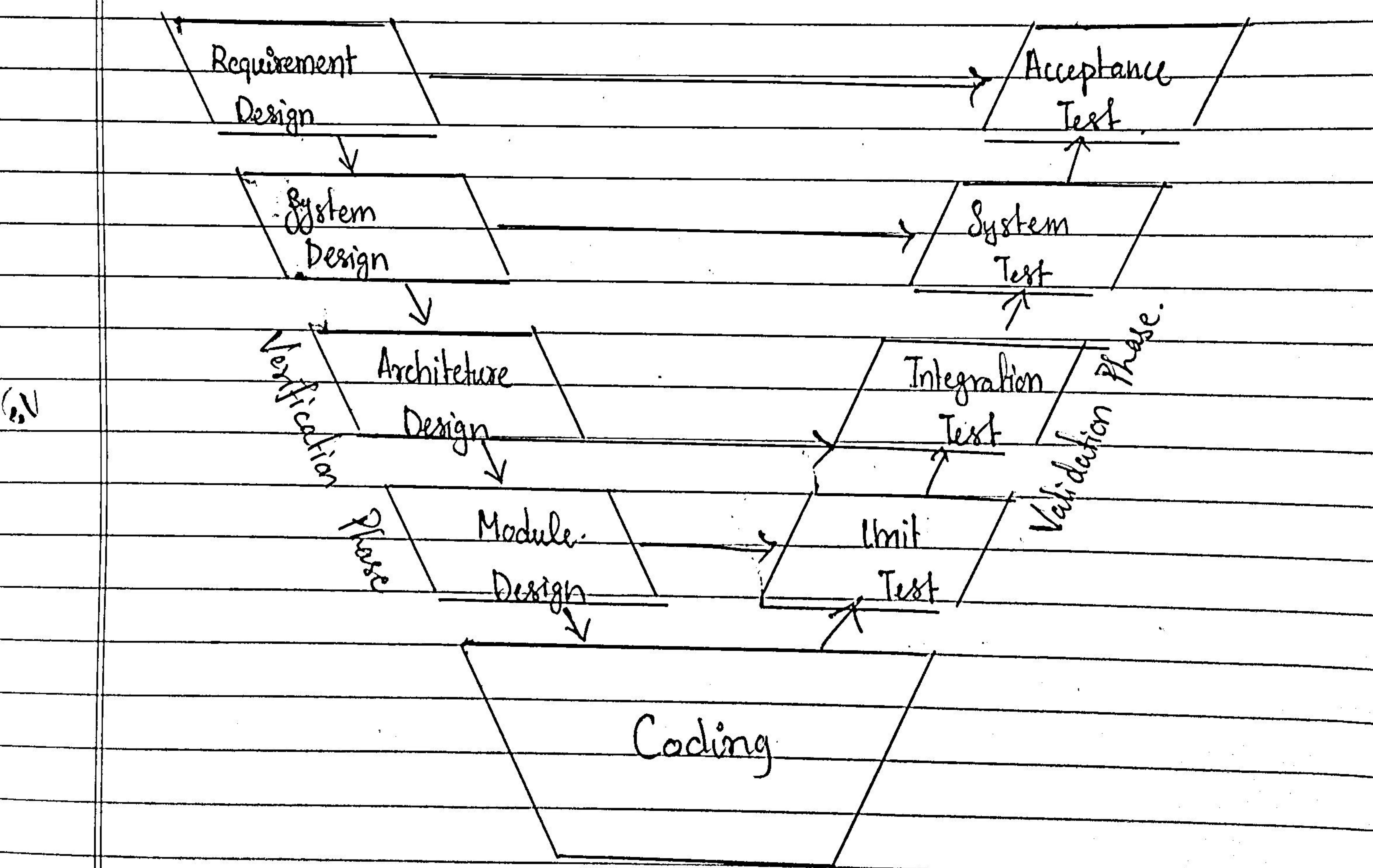
- i) It is more flexible to changing requirements
- ii) User can see system from 1 iteration to end of devel

Disadvantages :

- i) Difficult to manage development process
- ii) Not useful for small projects.

W V-Model : The V-model represents a development process that may be considered an extension of Waterfall. Instead of moving down in linear way, the process steps are bent upwards after the coding phase to form typical V-shape.

The horizontal & vertical axes represent time or project completeness (left to right) & level of abstraction respectively.



Shirong Zeng

T2-3

2201125.

Assignment 2.

Q.1. Differentiate between CPM and PERT

Ans.

	CPM	PERT
1) CPM stands for Critical Path Method.	PERT stands for Project Evaluation & review Technique.	
2) CPM is a technique of Project management which is used to manage only certain activities of any project.	PERT is a technique of project management which is used to manage uncertain activities of any project.	
3) It is deterministic model.	It is a Probability model.	
4) It majorly focuses on time-cost trade off as minimizing cost is more important.	It majorly focuses on time as meeting time target or estimation of percent completion is more important.	
5) It has respective nature of job.	It has non-respective nature of job.	

Q.2. Explain the difference between :

Ans. 1) Total Slack & Free Slack

Total Slack	Free Slack
1. The amount of time an activity can be delayed without delaying the project completion date.	The amount of time an activity can be delayed w/o delaying the start of its immediate successor(s).
2. Total Slack = Late Start - Early Start	Free Slack = Earliest start of next activity - Earliest finish of curr activity
3. If used up, it may affect the overall project schedule.	It does not affect project completion date but may impact only one successor task.
4. Helps in identifying activities that can be delayed w/o affecting deadlines.	Useful for scheduling flexibility w/o disturbing the next activity.

Activity-on-Node

- Activities are represented as nodes, and arrows show dependencies.

- Can represent all types of dependencies. (finish to start, start-to-start, etc)

- No dummy activities are required.

- Easier to construct & interpret.

- Used in Precedence diagramming method (PDM) and Modern Scheduling Software.

Activity-on-Arrow

- Activities are represented as arrows and nodes indicate events (start/end points).

- Primarily represents finish to start relationships.

- Dummy activities are sometimes needed to maintain logical sequencing.

- More complex.

- Used in arrow diagramming method (ADM) but less common today.

Q.3. Explain risk identification, risk projection & RMM plan.

Ans. Risk Identification is the process of recognizing potential risks that may affect project success. It involves analyzing internal & external factors that could cause project delays, cost overruns or failures.

Source of Risk —

- 1) Technical Risks , 2) Financial Risks
- 3) Schedule Risks , 4) Operational Risks.

Techniques for Risk Identification —

- 1) Brainstorming Sessions , 2) SWOT analysis
- 3) Expert Judgement , 4) Fishbone diagrams.

Risk Projection, involves assessing the probability and impact of identified risks. This helps prioritize risks based on their severity.

Risk impact categories —

- 1) Low impact , 2) Moderate impact.
- 3) High impact.

Risk Probability levels —

- 1) Low (0 to 30%)
- 2) Medium (30 to 70%)
- 3) High (70 to 100%)

• RMM Plan is a structured approach to managing risks throughout the project lifecycle.

a) Risk Mitigation :

- 1) Avoiding high risk activities.
- 2) using back up plans.
- 3) allocating additional resources.

b) Risk Monitoring :

- 1) Continuous risk assessments
- 2) Regular risk reports & meetings
- 3) Using key risk indicators

c) Risk Management :

- 1) Acceptance.
- 2) Transfer
- 3) Reduction.

Ans. Q.4

WBS - Computerization Project of ABC City Bank

1. Project Initiation

- 1.1 Requirement Gathering , 1.2. Feasibility Study
- 1.3. Stakeholder Meetings , 1.4. Project Charter.

2. System Design.

- 2.1 High level Architecture , 2.2 UI/UX Design
- 2.3 Database design , 2.4 Security Planning

3. Hardware & Infrastructure Setup.

- 3.1 Server Procurement , 3.2 Network Setup.
- 3.3 Workstation Setup , 3.4 Backup & Recovery Set.

4. Software Development

- 4.1 Core Banking Modules . 4.2 Admin Dashboard
- 4.3 Mobile Banking , 4.4 Integration w/ APIs

5. Testing .

- 5.1 Unit Testing , 5.2 Integration Testing .
- 5.3 System Testing , 5.4 User Acceptance Testing.

6. Deployment .

- 6.1 Pilot launch . 6.2. Full scale Rollout , 6.3 Performance Mon.

7. Training & Support.

- 7.1 Staff training Sessions 7.2 User Manuals & Docs
- 7.3 Technical Support Desk.

Responsibility Matrix

- a) R (Responsible)
- b) A (Accountable)
- c) C (Consulted)
- d) I (Informed)

Task	Project Manager	Business Analyst	Developers	IT Security Team	Testers
Requirement gathering	A	R	C	I	I
System Design	A	C	R	C	I
Software Development	I	I	R	C	I
Security Implementation	I	I	C	R	I
Testing & QA	I	C	C	R	R
Deployment	A	C	R	R	C
Training & Support	A	R	C	I	R

Ans. Q.S. SCM is the process of systematically handling changes to software in a way that maintains its integrity, traceability, and consistency over time. It's basically about keeping your project organized, controlled and documented - especially when multiple people are working on the same codebase or system.

o Key Objectives:

- 1) Control Changes - Ensure only authorized & intended changes are made.
- 2) Track Versions - Know which version of software is being used, by whom, and why.
- 3) Maintain Integrity - Avoid conflicts or bugs due to random or unmanaged changes.
- 4) Enable Collaboration - Multiple team members can work on different parts smoothly.
- 5) Support Reproducibility - You can recreate any past version.

o Core Components of SCM:

- a) Configuration Identification.
- b) Configuration Control.
- c) Configuration Status accounting.
- d) Configuration Auditing.

o Tools used in SCM:

- a) Version Control System - Git, SVN, Mercurial.
- b) Build Tools - Jenkins, Maven, Gradle.
- c) Issue Tracking - Jira, Trello, Buggilla.
- d) CI/CD Pipeline - Github Actions, GitLab CI.

s. Q.C. o A Gantt chart is a visual project management tool that displays tasks, their durations, dependencies, and progress over time using a bar chart format.

o Key Significance —

1) Visual Representation of Project Timeline: Displays tasks, durations & deadlines in a bar chart format.

2) Task Scheduling & Dependencies: Helps in organizing tasks in correct sequence and identification of dependencies.

3) Resource Allocation & Workload Management: Ensures optimal use of resources by preventing overallocation or underutilization.

4) Tracking Progress & Milestones: Allows monitoring of task completion status and achieving key project milestones on time.

5) Improves Communication & Coordination: Acts as a common reference for teams & stakeholders to enhance collaboration.

6) Risk Identification & Mitigation.

o Benefits of Gantt Chart —

1) Provides a clear picture of entire project.

2) Helps track deadlines & avoid delays.

3) Ensures all team members stay informed.

4) Flexible & adaptable.

o Limitations of Gantt Chart —

1) Can become cluttered with too many tasks.

2) Time-consuming to update.

3) Shows what needs to be done but not how.

4) Changes in one task can impact multiple other tasks.