

JANUARY - 2023

13/02/23

SUNDAY • JANUARY

| M | T | W | T | F | S | S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | | | |

01

WK 52
001 364)

JANUARY

Software Engineering & Project Management

Objective: To develop high quality software within budget, within time

IBM in 2000 (Survey)

12 noon | 50% Failed at beginning / Not completed
94% +
1pm 44% Developed but either over budget or over time.

6% Successful only

IBM in 2016 (Survey)

50% Failed at beginning / Not completed

+

12% to 22% Developed but either over budget or over time.

28% to 18% Successful

Cost of software = Hardware Cost

+ Software Development Cost

+ Software Maintenance Cost

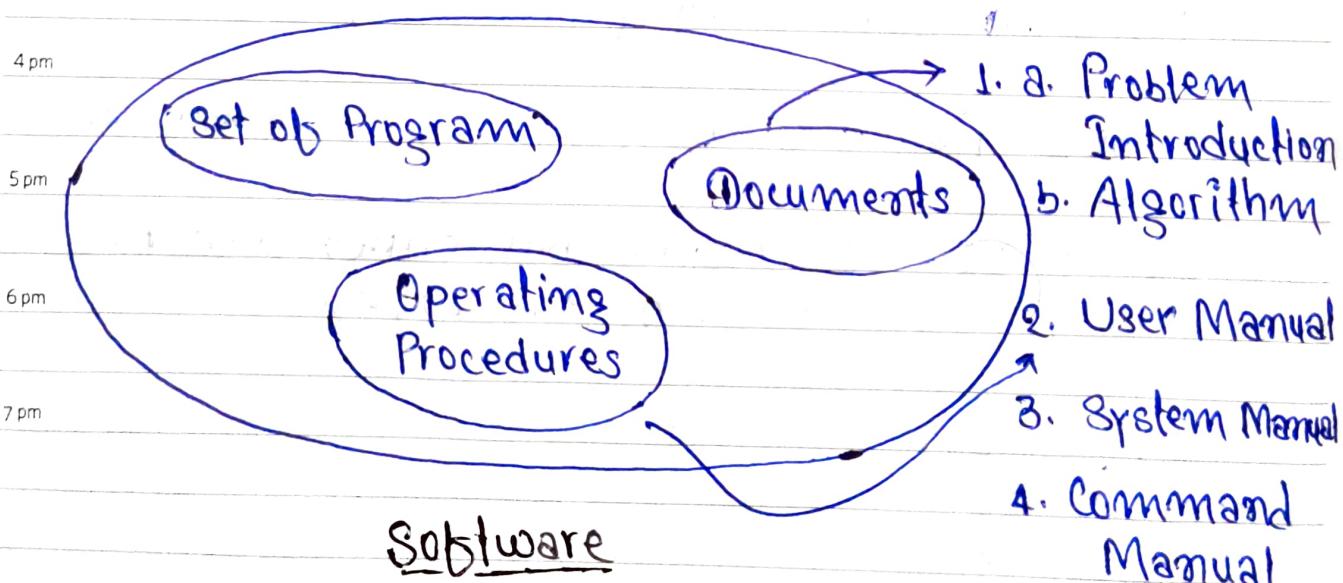
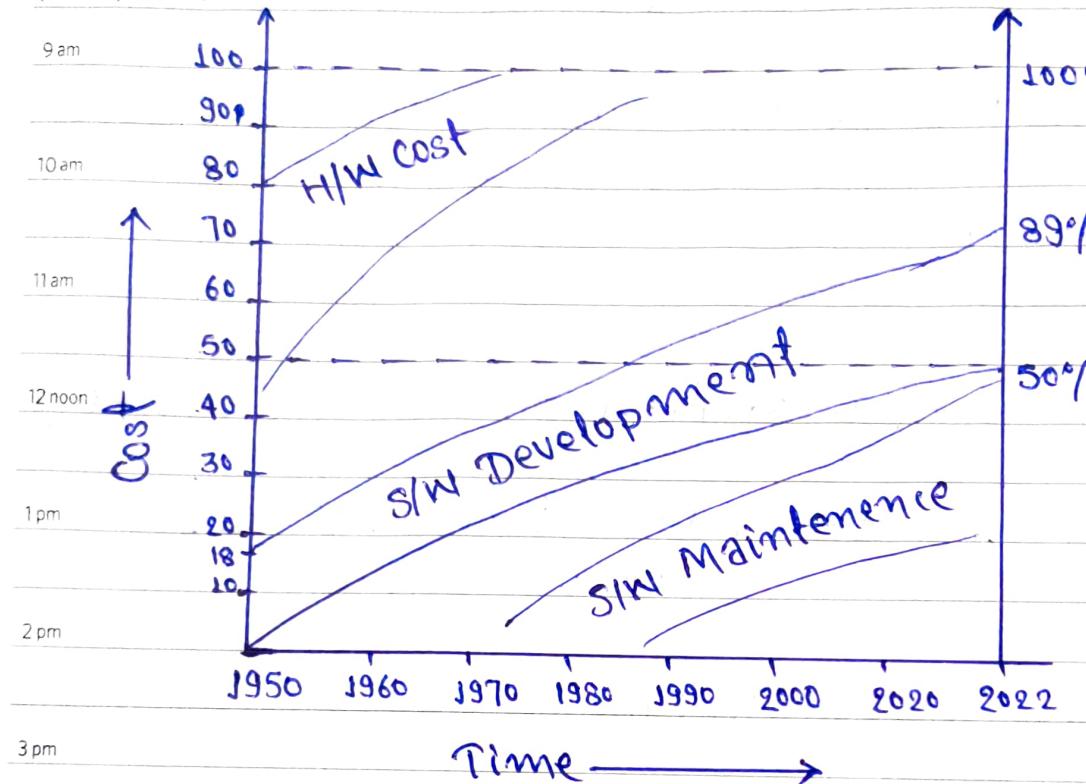
JANUARY • MONDAY

02

JANUARY - 2023

| M | T | W | T | F | S | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | | | | | | |

WK 01
(002-363)



CASE (Computer Aided Software Engine)

FEBRUARY - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | | | | | | | | | | | | |

TUESDAY • JANUARY

03

WK 01
(003-362)

Types of Software

9 am 1. System Software

General Purpose

10 am 2. Programming Software

3. Application Software

4. Administrative Software

Special Purpose

Characteristic of Software

- 1 pm 1. Software never manufacture only engineering.
2. Software never fail.

2 pm Engineering - Way to design new thing

3 pm System - Collection of inter-dependent component working together for a common goal.

4 pm Characteristic of system

1. Every system has more than one element.
2. Every element equally important
3.

14/02/23

Types of System

1. Open System
2. Close System

JANUARY • WEDNESDAY

04

WK 01
(004-361)

JANUARY - 2023

| M | T | W | T | F | S | S | M | T | W | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | | | | | | | | |

9 am

Boundary

10 am

Environment

11 am

Feedback & Control

Evaluation

12 noon

Input

I

Process

P

Output

Outer field

Inside

1 pm

2 pm

3 pm

Close System input inside the boundary & Open System input in Outer field.

4 pm

15/02/23

5 pm

Information System - Information system are integrated user machine system that provide right information to the right person for planning, organization, co-ordinating controlling, decision making within the organization.

| FEBRUARY - 2023 | | | | | | | | | | | | |
|-----------------|----|----|----|----|----|----|----|----|----|----|----|----|
| M | T | W | T | F | S | S | M | T | W | T | F | S |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 26 | 27 | 28 | | | | | | | | | | |

THURSDAY • JANUARY

05

WK 01
(005-360)

Documents

- 9am 1. Problem Introduction (Overview)
- 10am 2. Problem Solⁿ in the form of algorithm
- 11am 3. Work break down structure
(Decomposing larger problem into smaller manageable module)
- 12 noon 4. Data Flow Diagram (DFD)
 - Level 0 / Context Diagram
 - Level 1
 - Level 2
- 1pm 5. Entity Relationship Diagram
- 2pm 6. State Transition Diagram
- 3pm 7. UML
- 4pm 8. Input Screens / Output Screens
- 5pm 9. Test Cases
- 10. Glossary
- 11. References

Q. Explain document driven science in software engineering?

- Apart from that following are required
- 1. System Manual
 - 2. Command Manual
 - 3. User Manual

JANUARY • FRIDAY

06

WK 01
(006-359)

| JANUARY 2023 | | | | | | | | | | | |
|--------------|----|----|----|----|----|----|----|----|----|----|----|
| M | T | W | T | F | S | S | M | T | W | T | F |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | 29 | 30 | 31 | | | | | |

9 am 16/02/23

10 am

Characteristic of software

1. Software is very complex
2. Software does not wear-out but deteriorate.
3. Software does not manufacture but engineering and customized.
4. Software is prone to fail.
5. Software is costly to maintain.

2 pm

Software Quality

Quality is customer satisfaction

3 pm

4 pm

5 pm

6 pm

7 pm

① Dependability

② Performance

③ Security

FEBRUARY - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | | | | | | | | | | | | |

17/02/23

SATURDAY • JANUARY

07

WK 01
(007-358)

9 am

10 am

11 am

12 noon

1 pm

2 pm

3 pm

4 pm

5 pm

6 pm

7 pm

1. Dependability

- (a) Correctness
- (b) Reliable
- (c) Testable
- (d) Maintainable
- (e) Robust (Performance under extreme condⁿ)
- (f) Usable (Used whenever required)
- (g) Available (d)
- (h) Safe (Sustainable Development)

MTTF (Min Time To Failure)

MTTR (Min Time To repair)

$$\alpha = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}}$$

BPO (Business Process Outsourcing)

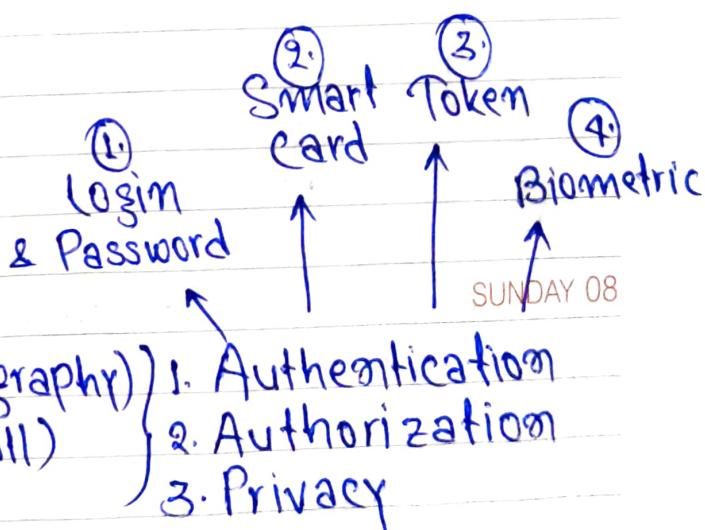
KPO (Knowledge Process Outsourcing)

2. Performance

- (a) Memory
- (b) CPU

3. Security

- (a) Internal (Cryptography)
- (b) External (firewall)



2023

JANUARY • MONDAY

09

WK'02
(009-356)

| JANUARY - 2023 | | | | | |
|----------------|----|----|----|----|----|
| M | T | W | T | F | S |
| 9 | 10 | 11 | 12 | 13 | 14 |
| 23 | 24 | 25 | 26 | 27 | 28 |
| 29 | 30 | 31 | | | |
| | | | | | |
| | | | | | |

9 am

Software classification based on Distribution

1. Retailware
2. Original Equipment Manufacturing sw
3. Fireware
4. Open Source sw
5. Shareware
6. Demoware
7. Spyware
8. Adware
9. Crippleware

2 pm

22/02/23

3 pm

Software Engineering By IEEE (1991)

Software Engineering is systematic, disciplined & quantifiable approach in software development

6 pm

① Development

② Operations

7 pm

③ Maintenance

Careers in Software Engineering

① End User

② Stack Holder

③ Sales & Marketing

④ Managers

⑤ Domain Experts

2023

TUESDAY • JANUARY

| FEBRUARY - 2023 | | | | | | | | | | | |
|-----------------|----|----|----|----|----|----|----|----|----|----|----|
| M | T | W | T | F | S | S | M | T | W | F | S |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | | | | | | | | |

10

NY 02
(010-355)

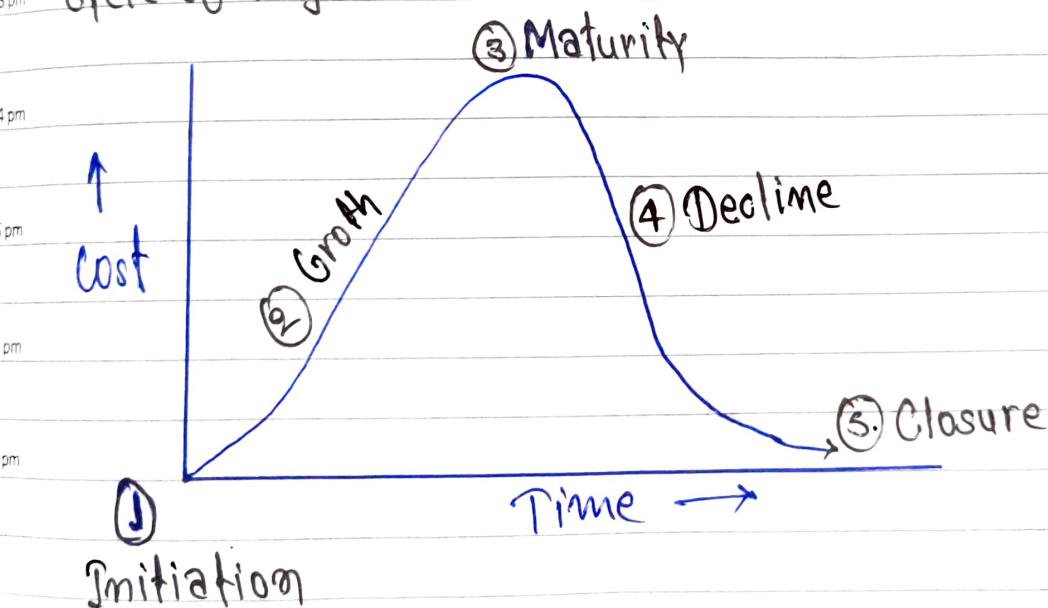
- 9 am ⑥ Analyst
- ⑦ Software designer & Architect
- 10 am ⑧ Technology Expert
- 11 am ⑨ Programmer
- 12 noon ⑩ Tester & Debugging
- ⑪ Software Engineer
- 1 pm ⑫ User support staff

Assignment

Challenges before software Engineering

2 pm

3 pm Cycle of Project



2023

23/02/23

JANUARY • WEDNESDAY

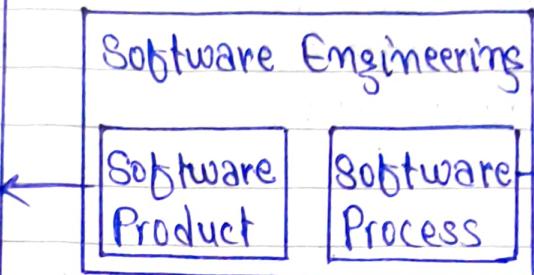
11

WK 02
(011-354)

| M | T | W | T | F | S | S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | | | | | | | 1 | 2 | 3 | 4 | 5 | 6 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | | | | |

9 am Software Engineering Activities

- 10 am • Software Analysis
- 11 am
- 12 noon
- 1 pm
- 2 pm • Software Writing/ Coding
- 3 pm • Software Operations
- 4 pm • Software Maintenance



- Software Process Development
- SW Process R & D and enhancement
- Software Process
- Mgt & Control
- SW Product Mgt & Control

5 pm

6 pm

BPR (Business Process Re-engineering)
7 pm → 95%
Only 5% Engineered 1st time.

Def of Project: activity
Project is a non-routine task with some module at specific time.

FEBRUARY - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | | | | | | | | | | | | |

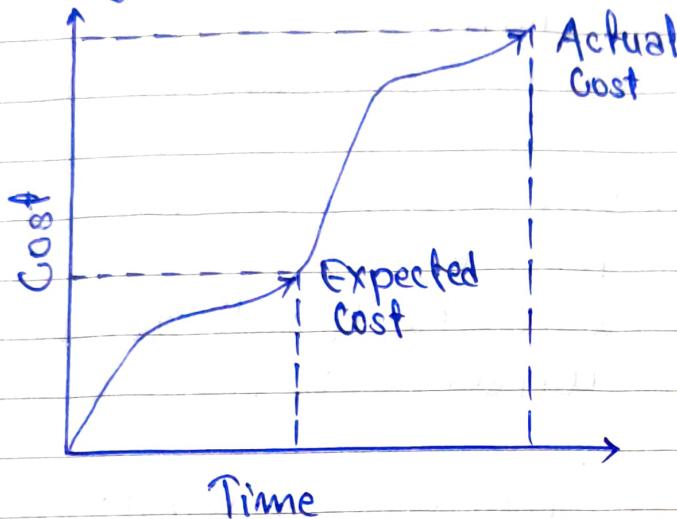
THURSDAY • JANUARY

12

WK 02
(012-353)

Challanges before SW Engineering

① Rising Cost



② Knowledge of Design Approach

③ Knowledge of Process Model

SDLC - Best tool for project development

- LSM (Linear Sequential Model) (V-Shaped Model)
- PRM (Prototype Model)
- INM
- RAD (Rapid Development Model)
- Spiral Model
- Win-win Spiral Model
- Agile SW Development

④ Handling the Project ~~at~~ at abstraction level

⑤ Measuring & Planning the Project

2023

JANUARY • FRIDAY

13

WK 02
(013-352)

| JANUARY - 2023 | | | | | | |
|----------------|----|----|----|----|----|----|
| M | T | W | T | F | S | S |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 |
| 30 | 31 | | | | | |

9 am Myths in Software Engineering

10 am 1. SW is easy to change

Fact:- (SW development & maintenance is very difficult)

11 am 2. Computer provide greater reliability than devices they replace

12 noon 3. Increasing SW reliability will increase safety

1 pm 4. Testing SW correct can remove all SW errors.

2 pm 5. Reusing SW increasing reliability & safety

6. By eliminating operators human error can be eliminated.

3 pm 7. Computer provides better information to the operators

4 pm 8. Software never fails.

5 pm

6 pm

7 pm

24/02/23

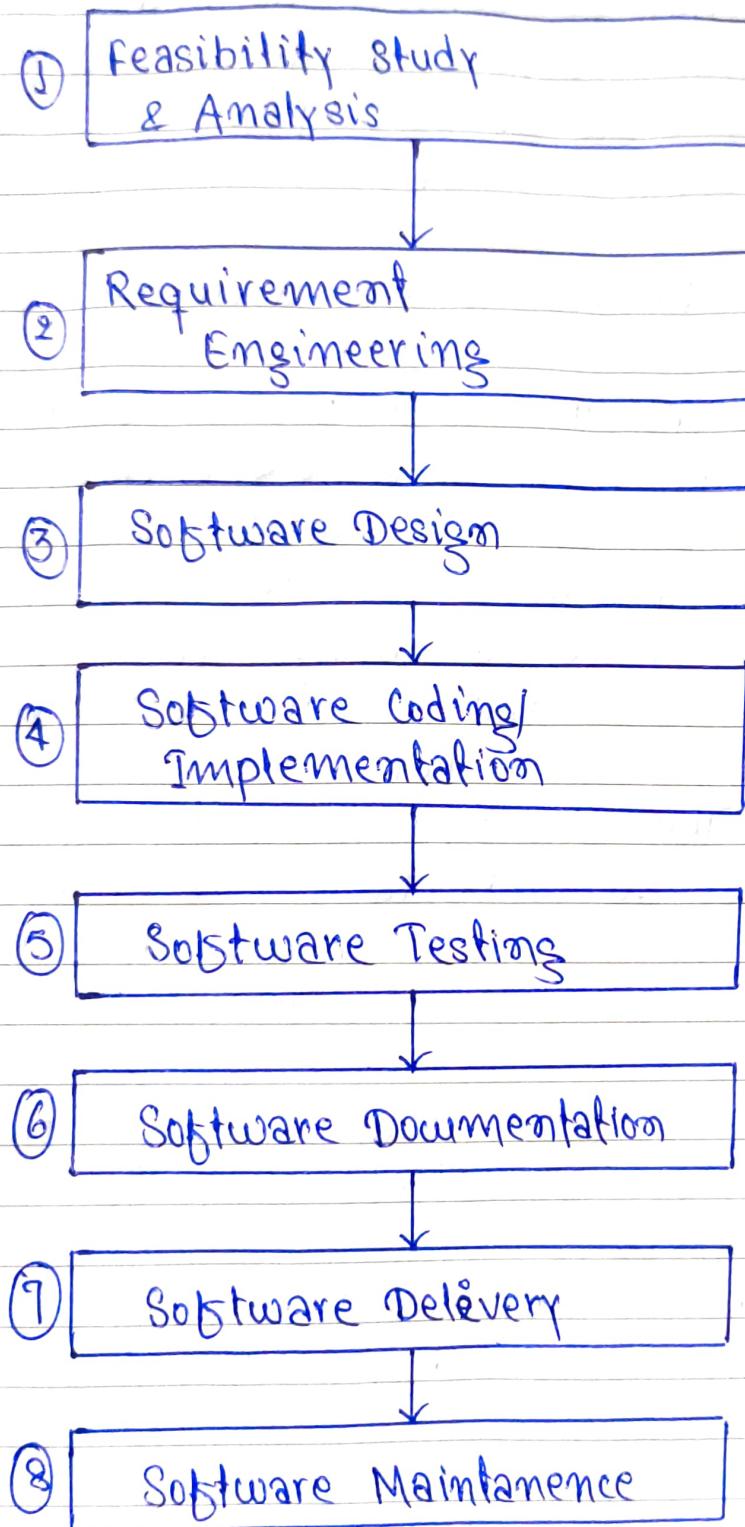
| FEBRUARY - 2023 | | | | | | | | | | | |
|-----------------|----|----|----|----|----|----|----|----|----|----|----|
| M | T | W | T | F | S | S | M | T | W | T | F |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | | | | | | | | |

SATURDAY • JANUARY

14

WK 02
(014-351)

Software Process Model



SUNDAY 15

2023

JANUARY • MONDAY

JANUARY - 2023

16

WK 03
(016-349)

| M | T | W | T | F | S | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | | | | | | |

9 am Process may be defined as phase by action that carried out in a non-zero but finite time to accomplish a predefined well established goal

10 am Software Process is for developing SW application.

11 am All development process start with their needs then planning is done to achieve the objective during planning all the available & required resources are compared to obtain an idea of feasibility of the solution & the management of required resources.

12 noon 1 pm 2 pm 3 pm 4 pm 5 pm 6 pm 7 pm

The SW process & its effectiveness may not be important for short term-vision organization or team but have a great impact for the long term project.

Long term SW developer & practices must pay fair attention follow formulae & establish for their development else they may compromise in the quality

FEBRUARY 2023

| | | | | | | |
|----|----|----|----|----|----|----|
| S | M | T | W | T | F | S |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| 22 | 23 | 24 | 25 | 26 | | |

TUESDAY • JANUARY

17

WK 03
(017-348)

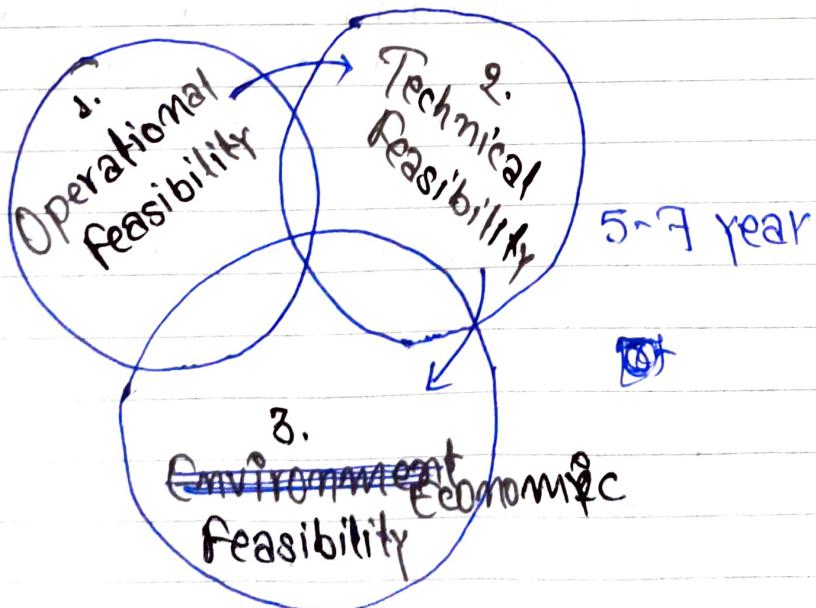
Need for mature SW process

- 1) SW development means good understanding for goal & objective
- 2) Acceptance criteria
- 3) Clear Plan
- 4) Specialized technical skills
- 5) Testing Methodology

A mature process provides predictable, measurable, manageable, flexible & practical plan for SW development.

It not only provides a step wise route from requirement to implementation but also sets the environment for enhancement of development of software.

Feasibility Study



01/03/23

JANUARY • WEDNESDAY

18

Wk 03
(01B-347)

| JANUARY | | | | | | |
|---------|----|----|----|----|----|----|
| M | T | W | T | F | S | S |
| 1 | 2 | 3 | 4 | 5 | 6 | |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 |
| 30 | 31 | | | | | |

9 am Trade of Analysis

Trade of Analysis is conducted to select best solution (possible solⁿ)

11 am Estimation

- Money ✓
- Material ✗
- Manpower ✓
- Machine / Equipment ✓
- Time ✓
- Efforts ✓

3 pm

4 pm Requirement Engineering

Req Eng is the 2nd phase of s/w dev process
it is process of establishing the services that the customer require from a system & constraints under which a system operates & develop

Req Eng is the key activity of s/w dev process

Outcomes of Req Engineering

- a) SRS Document
- b) User Acceptance Plan
- c) User Acceptance Plan

2023

09/03/23

THURSDAY • JANUARY

| FEBRUARY - 2023 | | | | | | |
|-----------------|----|----|----|----|----|----|
| M | T | W | T | F | S | S |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | | | | | |

19

WK 03
(019-316)

1. Context Analysis
2. Requirements Elicitation
3. Req Assessment
4. Req Specification
5. Req Exploration

Context Analysis

Reg Eng provides the bridge btwn application domain with its demands & goal & s/w & h/w soft.

Every mistake in Reg Eng process is extremely costly either in terms of correction or in terms of providing an user with invalid system that reflect inconsistent

The main task of Reg Eng process is to determine & specify explicitly the statuted an employee need. That is a req the system has fulfill.

These needs has their origin in certain context such as stake holder, goal & objective, assumptions & application domain



Realistic & Practical

Specification is the task of presizing describing the s/w previously defined as mathematically regress way.

The informal know about the functionality, behavioural, expectation performance needs as gather from customer & them

JANUARY • FRIDAY

20

WK 03
(020-345)

| JANUARY | | | | | |
|---------|----|----|----|----|----|
| M | T | W | T | F | S |
| | | | | | S |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 9 | 10 | 11 | 12 | 13 | 14 |
| 15 | 16 | 17 | 18 | 19 | 20 |
| 23 | 24 | 25 | 26 | 27 | 28 |
| 29 | 30 | 31 | | | |

9 am

these needs to be translated into formal & ambiguous specification those are well documented in SRS document.

10 am

03/03/22:

11 am

Requirement Elicitation Technique & Tools

① Observation

12 noon

→ Interviews

Advantage

1 pm

- Collect any information

2 pm

Disadvantage

- Time Consuming
- Costly

3 pm

② → Questionnaires

↳ Open Ended

↳ Close Ended

Tools used in Questionnaires

4 pm

OMR

OCR

BCR

MICR

5 pm

6 pm

7 pm

questionnaire is the best tools for s/o Engineer.

SATURDAY • JANUARY

21

WK 03
(021-344)

| FEBRUARY - 2023 | | | | | | | | | | | | | |
|-----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|
| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |

Close Ended

- MCQ
- T/F or Y/N (Dichotomous Questionnaire)
- Filling the blanks
- Rating
- Ranking

④ Delphi Method

1pm

2pm

3pm

4pm

5pm

6pm

7pm

SUNDAY 22

2023

16/03/23

JANUARY • MONDAY

23

JANUARY - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | | | | | |

WK 04
(023-342)

SRS

- Q1. Why SRS is so much important in SW dev
- Q2. describe ~~good~~ characteristic of good SRS
- Q3. also describe IEEE defining & drafting SRS for SW development.

11 am



12 noon

SRS (System Software Requirement Specification)

1 pm

is the most important document that defines all the functionality, behaviors, expectation, performance & reliability as collected from customer & then need to be translated to formal & unambiguous specification that are well documented as SRS. When a SW is developed then system against SRS description.

2 pm

A good quality SRS can provide several SRS such as establishing the bases for contractual agreement b/w the customer & developers & for performing cost schedule & resource estimates for dev. SRS is such important document that plays a vital role during verification as well as validation.

FEBRUARY - 2023

| M | T | W | T | F | S | S | M |
|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 27 | 28 | | | | | | |

(SQA) Quality Assurance

sw dev Jam
Tuesday • JANUARY

UML (Unified Modeling Lang)

WK 04
(024-341)

24

characteristic of good SRS

1> SRS should be correct

2> Consistent

3> Complete

4> Unambiguous

5> ~~Very~~ Verifiable

6> Tracable

7> Modifiable / Flexible

8> Simple

9> Relevant

10> Usable

11> Affordable

IEEE 830 standard for good SRS

I → Introduction

1.1 Purpose

1.2 Scope

1.3 Glossary

1.4 References

1.5 Overview

II → Requirement Specification

2.1 System Environ

2.2 User Description

2.3 User Case

2.4 Use Case (UML) ↗ DFD

2.5 UI Specification

2.6 Non Functional Img

ER Diagram

DFD

→ S/W Eng is Layered Technology

fully Automated

Semi Automated

4. Tools (CASE Tools)

3 Method Layer

2 Process Layer

1 Focus on Quality

2023

17/03/23

JANUARY • WEDNESDAY

25

WK 04
(025-340)

| JANUARY | | | | | | |
|---------|----|----|----|----|----|----|
| M | T | W | T | F | S | S |
| | | | | | 1 | 2 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 |
| | | | | | 30 | 31 |
| | | | | | 5 | 6 |
| | | | | | 18 | 19 |
| | | | | | 20 | |

III System Evolution

10 am Based on SRSD

III. Software Design:

11 am

12 noon

1 pm

2 pm

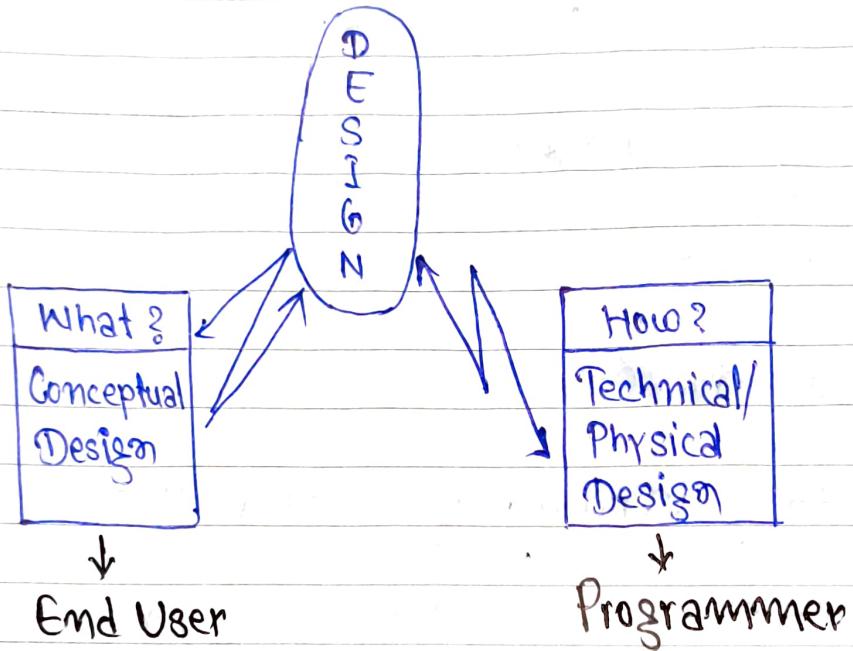
3 pm

4 pm

5 pm

6 pm

7 pm



FEBRUARY - 2023

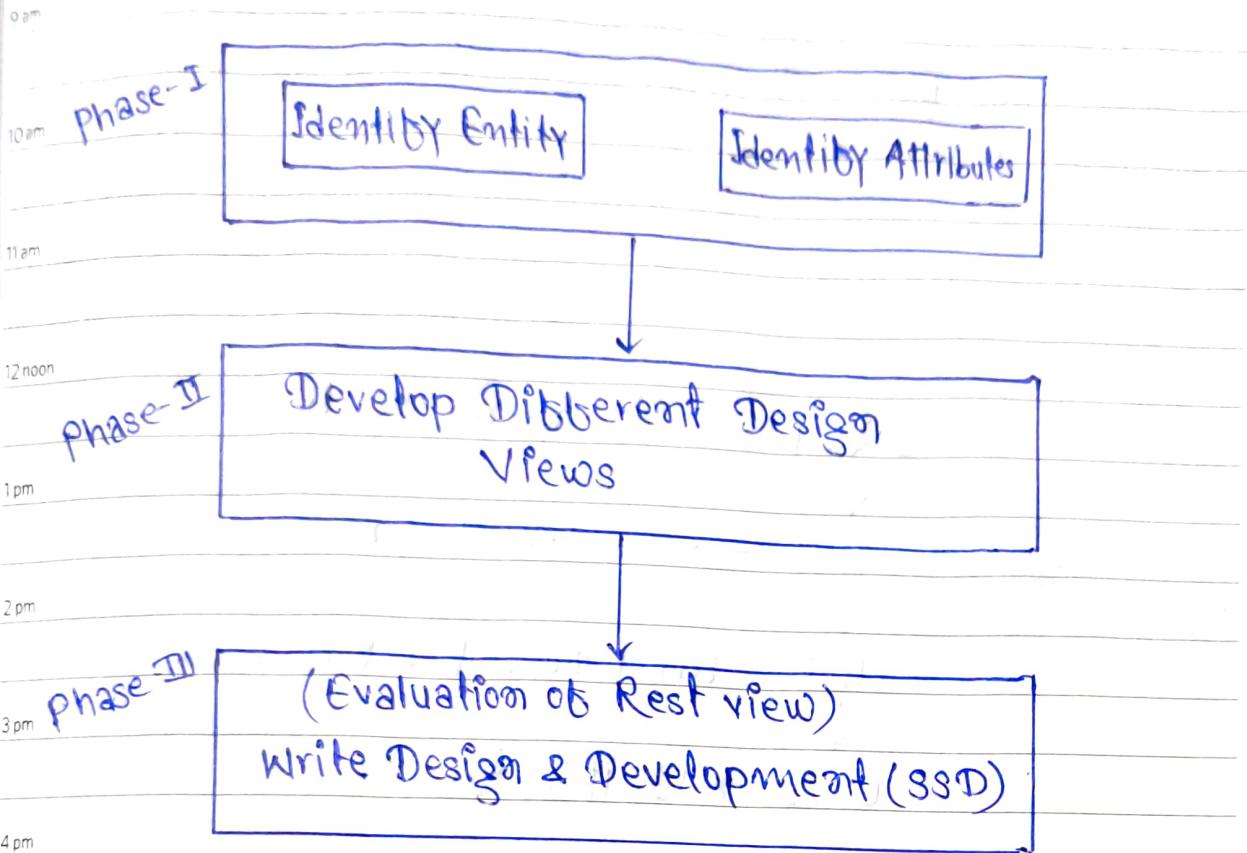
| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | | | | | | | | | | | | |

THURSDAY • JANUARY

26

WK 04
(026-339)

Software Designing



S/W Design aim to plan a blue print for the implement of the s/w. The main obj of s/w design process is to cover gap in understanding SRS & implementing them technically as s/w. The design s/w technical as s/w. The des s/w take input from SRS, dedicated to plan for the implementation of the s/w. design plans the system at diff levels of details from large grain size to fine grain size.

22/03/23

JANUARY • FRIDAY

27

WK 04
(027-338)

| JANUARY 2023 | | | | | |
|--------------|----|----|----|----|----|
| M | T | W | F | S | S |
| | | | | 1 | 2 |
| 9 | 10 | 11 | 12 | 13 | 14 |
| 23 | 24 | 25 | 26 | 27 | 28 |
| | | | | 29 | 30 |
| | | | | 31 | |

9 am

SLIN development process is based on ~~phase~~
~~bired~~ document

10 am

11 am

12 noon

1 pm

Verification

SRSD → Acceptance Testing



SDD → System Testing



HLD → Functional/Integration Testing



LLD → Unit Testing

3 pm

Documentation

- (a) User Manual
- (b) System Manual
- (c) Command Manual

5 pm

6 pm

7 pm

Change is always registered

I. Development

II. Installation / Implementation

- 1. Over Night Implementation (Big-Bang Imp)
- 2. Phased "
- 3. Parallel "
- 4. Pilot "
- 5. Hybrid "

2023

23/03/23
SATURDAY • JANUARY

28

WK 04
(028-337)

| FEBRUARY 2023 | | | | | | | | | | | |
|---------------|----|----|----|----|----|----|----|----|----|----|----|
| M | T | W | T | F | S | S | M | T | W | T | F |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | | | | | | | | |

9 am VIII. Software Maintenance

As per IBM survey

Types

- I. Corrective (21%)
- II. Adaptive (25%)
- III. Perfective (50%) / Enhansive
- IV. Preventive } (4%)
- V. Gold Plating }

Software Maintenance Activities

- Problem / Bug Reporting
- Problem / Bug Analysis
- Patch Creating
- Minor Enhancement
- Minor Adaption
- Minor Design Alteration
- Version Control
- Corrective Bug Fixing

Major

Moderate

Minor

SUNDAY 29

2023

JANUARY • MONDAY

30

WK 05
(030-335)

JANUARY - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | | | | |

9 am Software Development Process Model (SDLC)

- Tools of Project Management

10 am 1. LSM (Linear Sequential Model/Waterfall Model/
Classical Development Model)

~~enhancement~~

2. PRM (Prototype Model)

- 2.1. Throwaway Model
- 2.2. Exploratory Model

pm

→ 3. V-shaped

2 pm

4. RAD (Rapid Application Development)

3 pm

5. INM (Incremental Software Development)

4 pm

6. Spiral Model (BSM)

5 pm

7. Win-Win Spiral Model

6 pm

8. Agile Software Development Model

- Extreme Programming
- Pair Programming
- Scrub Model & Agile Development Process

24/03/23

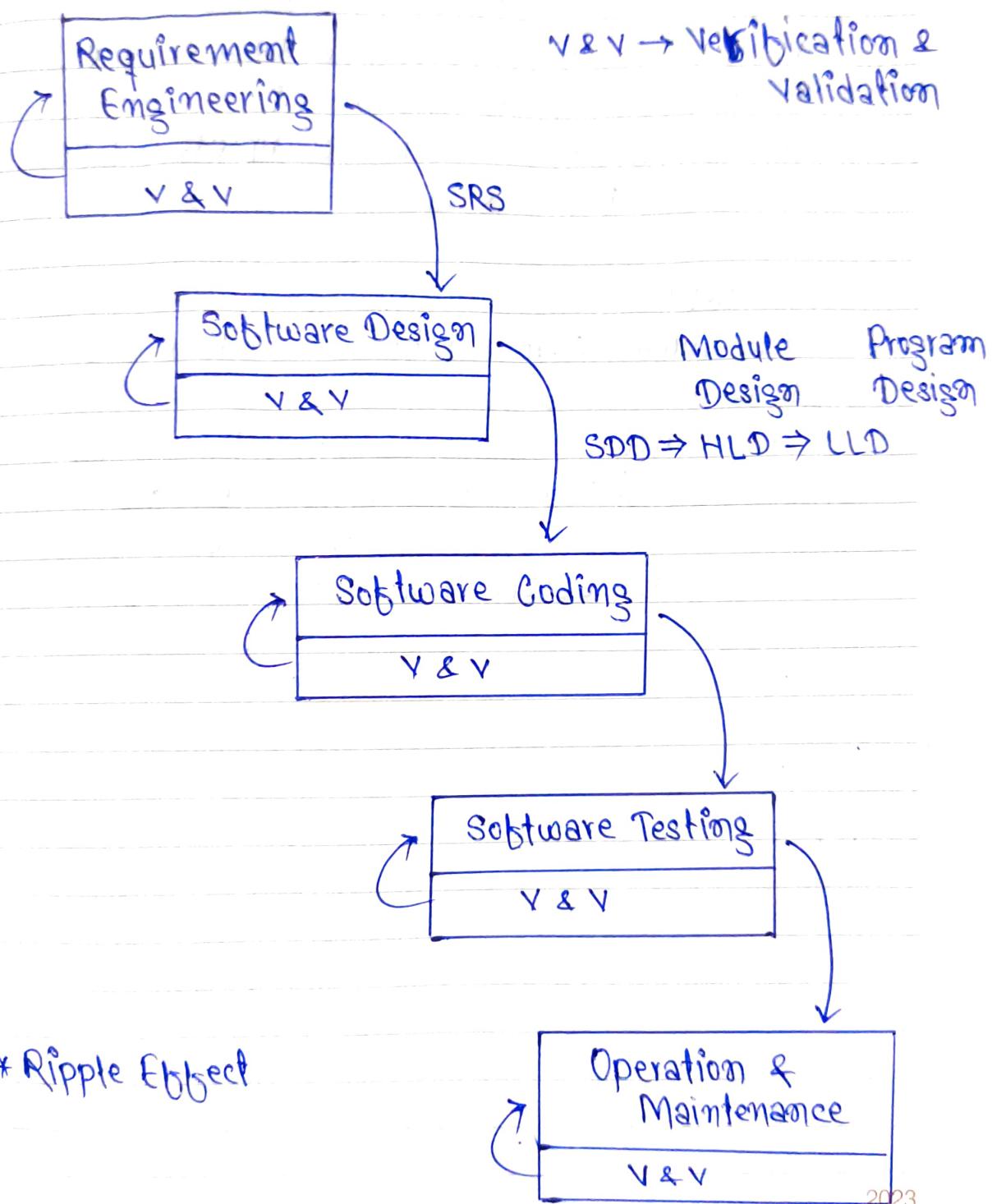
FEBRUARY - 2023
T W T F S S M T W T F S S
1 2 3 4 5 6 7 8 9 10 11 12
14 15 16 17 18 19 20 21 22 23 24 25 26
28

TUESDAY • JANUARY

31

WEEK 05
(031-134)

Linear Sequential Model / Waterfall Model / Classical Software Development Model (Developed by ROYCE in 1970)

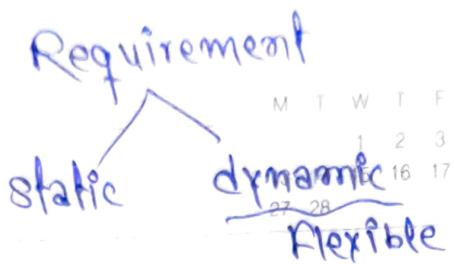


2023

FEBRUARY • WEDNESDAY

01

Wk 05
(032-333)



FEBRUARY - 2023

9 am Advantages

1. It is best for beginners.
2. This model is also best suitable for small project & development where all the requirements are fully known at the beginning of development.
3. This model produce the complete software solution at the end.

1 pm

Limitation

1. This model can not be used where requirement is not clear & stable.
2. The outcome can not be verified in the middle of the development process.
3. Once the outcome of this model is not satisfactory it is very difficult to change.

2 pm

6 pm

7 pm

MARCH - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | | | | | | | | | |

THURSDAY • FEBRUARY

02

WK 05
(033-332)

FEBRUARY

Documentation

Waterfall model is document driven model. In the waterfall model every phase generates a very specific document on the basis of the next phase activity get initiated. Following documents are generated -

- 1.) Software Requirement Specification document generated by Requirement Engineering.
- 2.) Software Design Description generated by Software Design further decomposed into High level design & low level design
- 3.) Unit Test Plan - Such plans are designed along with low level design description of every unit program.
- 4.) User Test Plans are Functional Test Plan
- 5.) System Test Plan & System Test Cases
- 6.) Installation plan & delivery document
- 7.) Maintenance & Enhancement Tracing Document.

FEBRUARY • FRIDAY

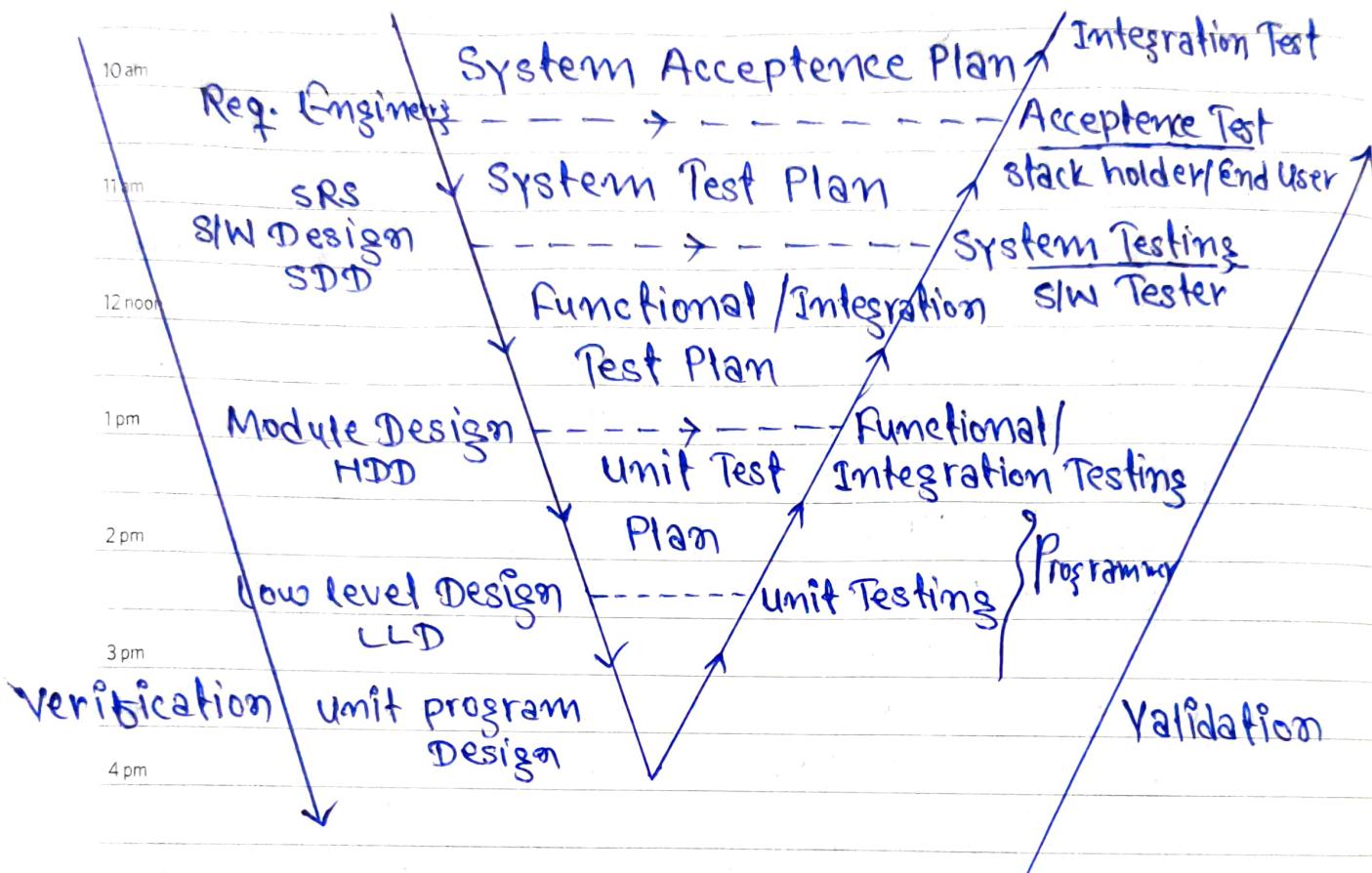
03

WK 05
(034-331)

FEBRUARY - 2023

| M | T | W | T | F | S | S | S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 28 |
| 27 | 28 | | | | | | | | | | | | |

9 am V-Shaped Model (Extension of Waterfall Model)



29/03/23

MARCH - 2023

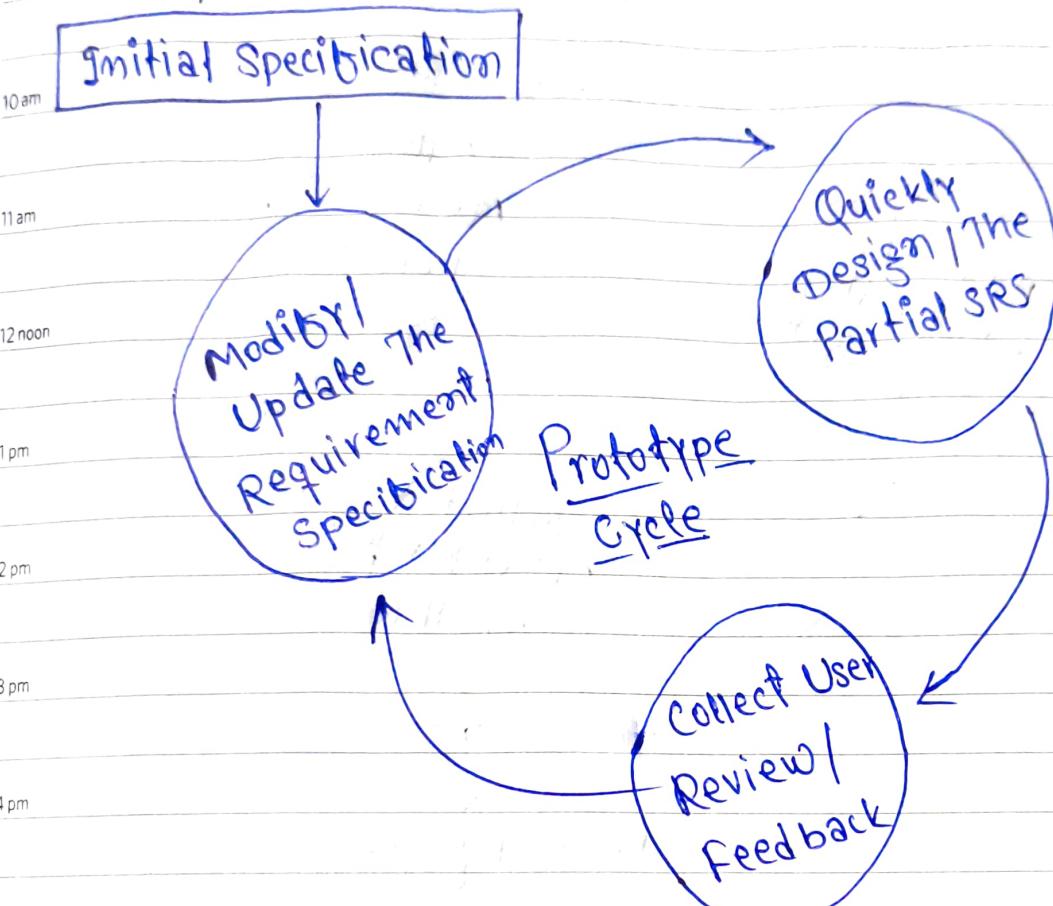
| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | | | | | | | | | |

SATURDAY • FEBRUARY

04

WK 05
(035-330)

9 am Prototyping Model (PRM)



Types of Prototype

- ① Throw Away Prototype Model
- ② Evolutionary / Exploratory Prototype Model SUNDAY 05

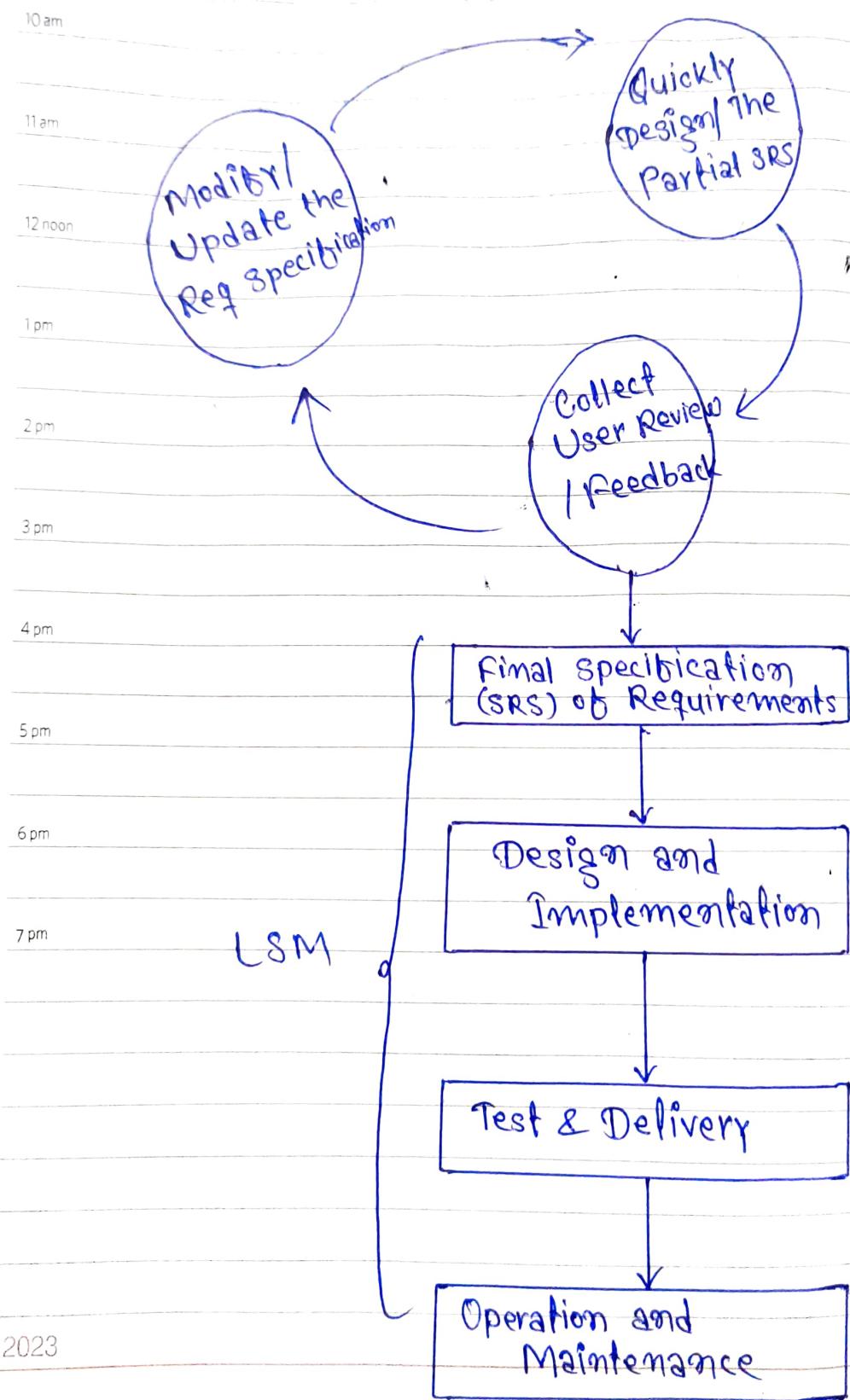
FEBRUARY • MONDAY

06

WK 06
(037 328)

| M | T | W | F | S | S | M | T | W | T | F | B | E | J | J | A | R | M |
|----|----|----|----|----|----|----|----|----|---|---|----|----|----|----|----|----|----|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | | | | | | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | | | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| 27 | 28 | | | | | | | | | | 28 | 29 | 30 | 31 | | | |

9 am Throw Away Prototype Model



MARCH - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | | | | | | | | | |

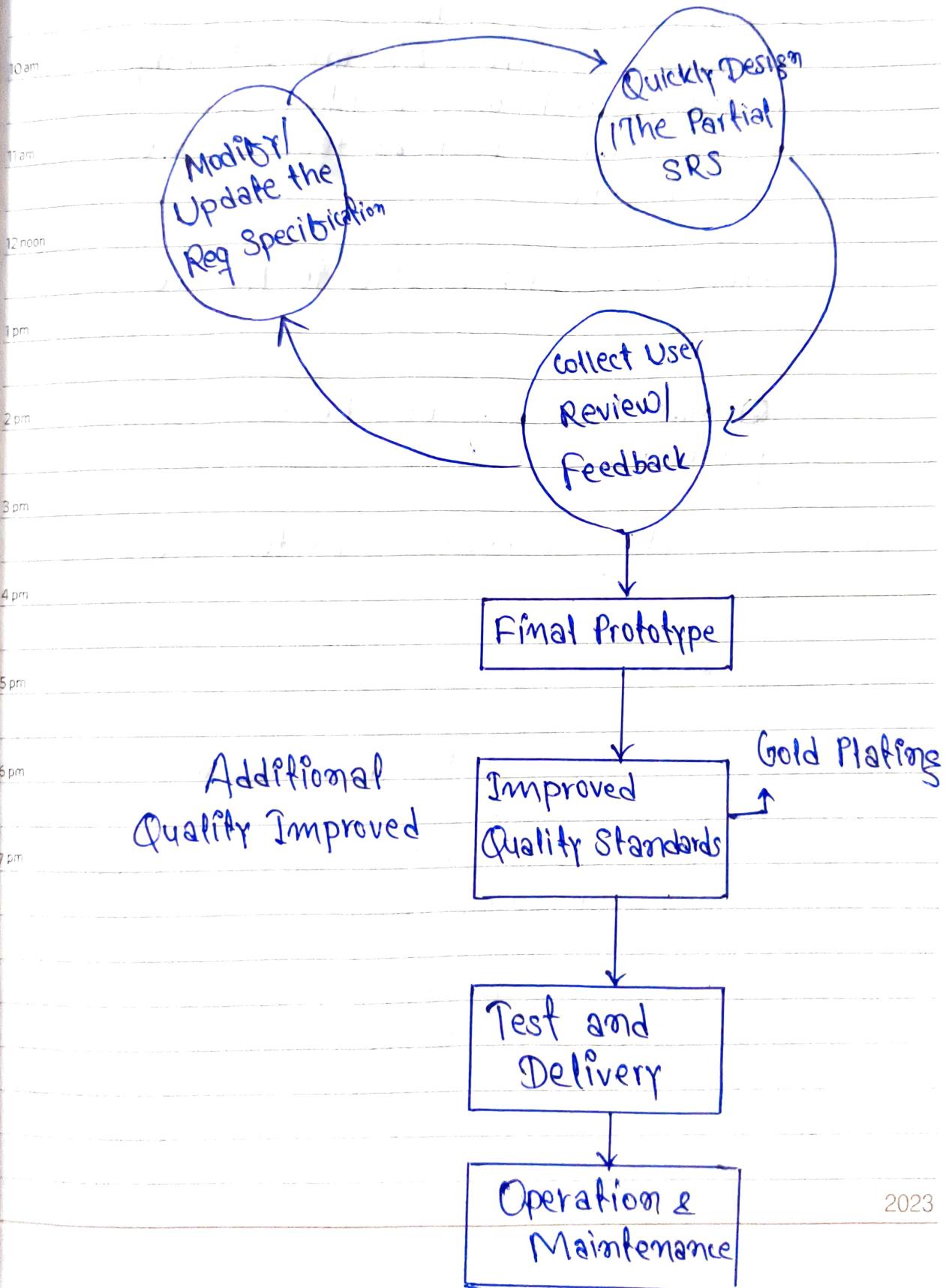
TUESDAY • FEBRUARY

07

WK 06
(038-327)

Iterative Model

Evolutionary / Exploratory Prototype Model



2023

FEBRUARY • WEDNESDAY

08

FEBRUARY - 2023

| M | T | W | T | F | S | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | |
| | | | | 27 | 28 | | | | | | | | | |

WK 06
(039-326)

9 am

Advantages

- ① Best for following situation
- A. Where requirements are not completely known
 - B. Where technology is fluctuating
 - C. Where development team is not very experienced

10 am

- ② In prototyping, Risk related to failure can be reduced.

11 am

- ③ Financial Risk can also be controlled in prototyping

12 noon

- ④ We can check & evaluate prototype any time

1 pm

Disadvantages

2 pm

- ① Prototyping can not be done through contractual teams.

3 pm

- Prototyping is best for in house software development team.

- ② Most of the time prototyping are evaluated as final product.

31/03/23

MARCH - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | | | | | | | | | |

THURSDAY • FEBRUARY

09

INR 06
(040 325)

Component Based SW Development Models

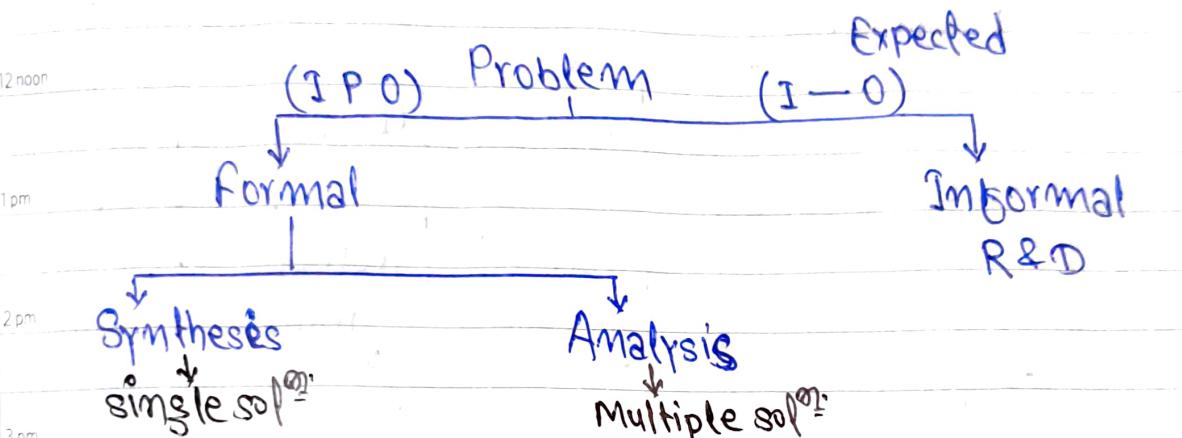
RAD Model (Rapid Application Development Model)

Only for Business System or Expert System

1 → Worked with Domain Expert Teams

2 → Reusability

3 → Worked within time frame (60 to 90 days)



→ A problem which has only one specific solution is known as Synthesis problem

→ A problem which has many solutions and out of those the best solution is known as problem of Analysis

4 → JAD Technique (Joint Application Development)

5 → 4 GT & Tools

(4th Generation Language & Technique)

FEBRUARY • FRIDAY

FEBRUARY - 2023

10

WK 06
(041-324)

9 am

10 am

11 am

12 noon

1 pm

2 pm

3 pm

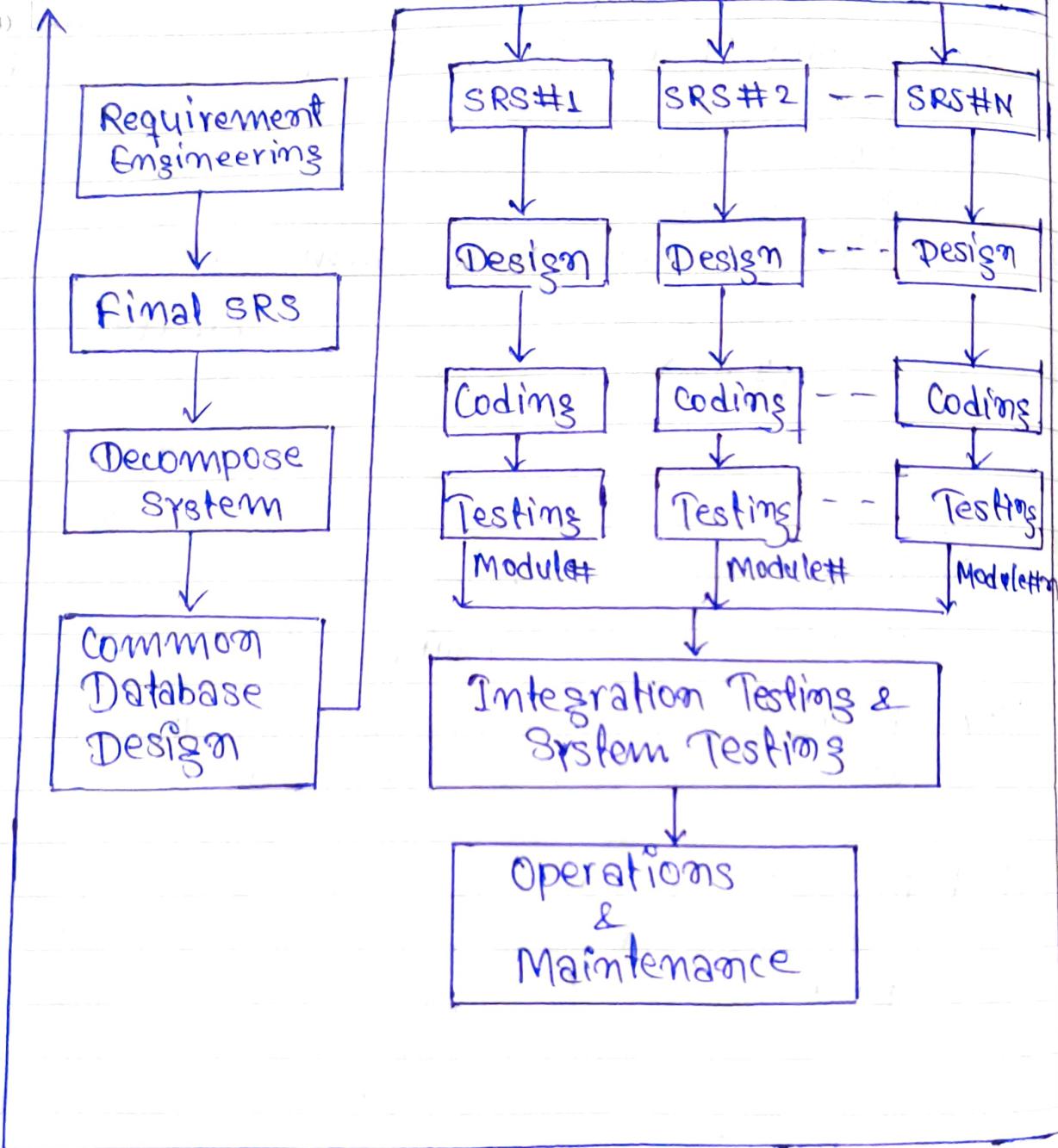
4 pm

5 pm

6 pm

7 pm

| M | T | W | T | F | S | S | S | M | T | W | T | F | S | S | |
|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| | | | | | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | |
| | | | | | 27 | 28 | | | | | | | | | 12 |
| | | | | | | | | | | | | | | | 26 |



60 - 90 days

05/04/93

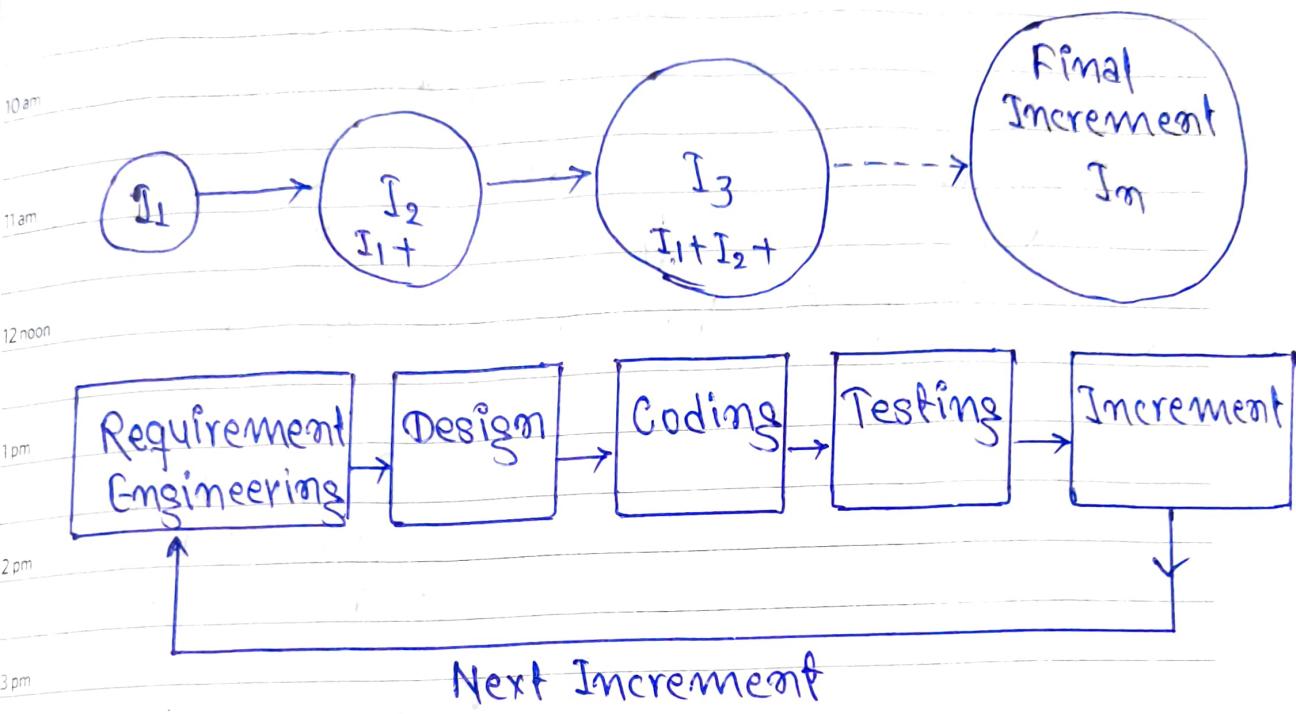
SATURDAY • FEBRUARY

11

WK 06
(042-323)

| MARCH - 2023 | | | | | | |
|--------------|----|----|----|----|----|----|
| M | T | W | T | F | S | S |
| - | 1 | 2 | 3 | 4 | 5 | 6 |
| 7 | 8 | 9 | 10 | 11 | 12 | |
| 13 | 14 | 15 | 16 | 17 | 18 | |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 26 | 27 | 28 | 29 | 30 | 31 | |

Incremental Model (INM)



SUNDAY 12

2023

FEBRUARY • MONDAY

13

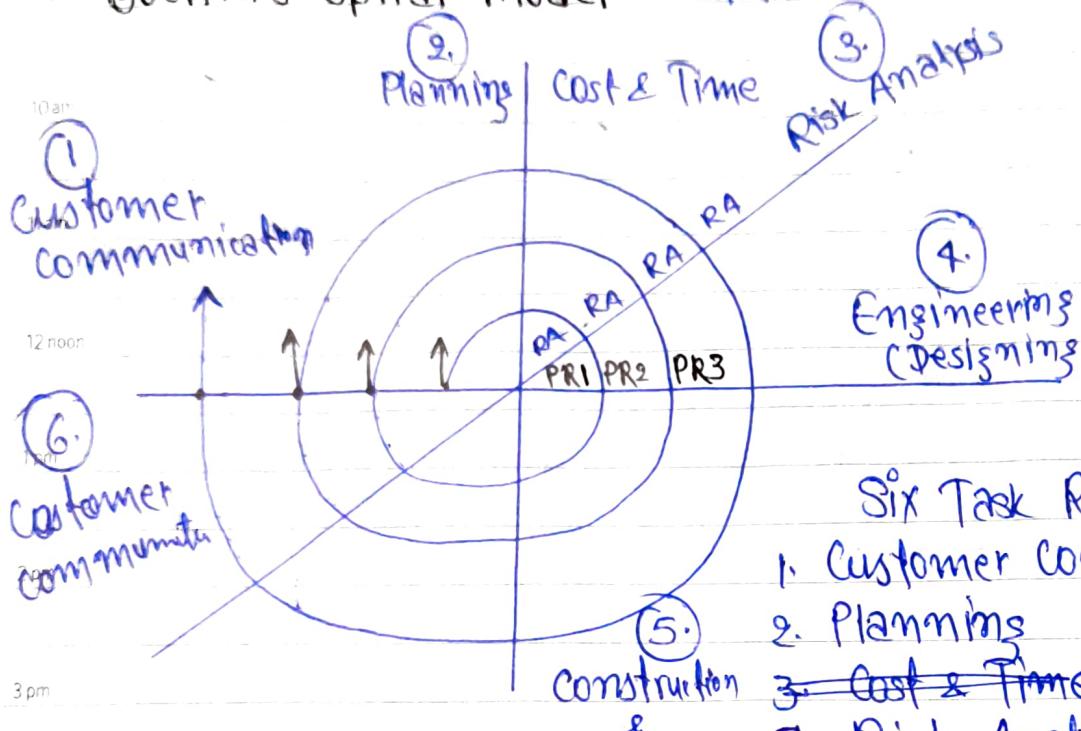
PR → Prototype

RISK → Causes for Failure

| S | T | F | S | M | T | W | F | S |
|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| 27 | 28 | | | | | | | |

WK 07
(044-321)

9 am Bochim's Spiral Model → Meta Model



Project Risk

1. People (69%)
2. Process (18%)
3. Technology (13%)

Six Task Regions

1. Customer Communication
2. Planning
3. Cost & Time
3. Risk Analysis
4. Engineering
5. Construction & Release
6. Customer Communication

Risk Management

1. Identify Risk
2. Risk assessment
3. Just planning (how to solve), execution
4. Risk resolve are mitigation

Now

Win Win Spiral Model

Extreme Programming

Pair Programming

2023

Scrub model for Agile Model

06/04/23

MARCH - 2023

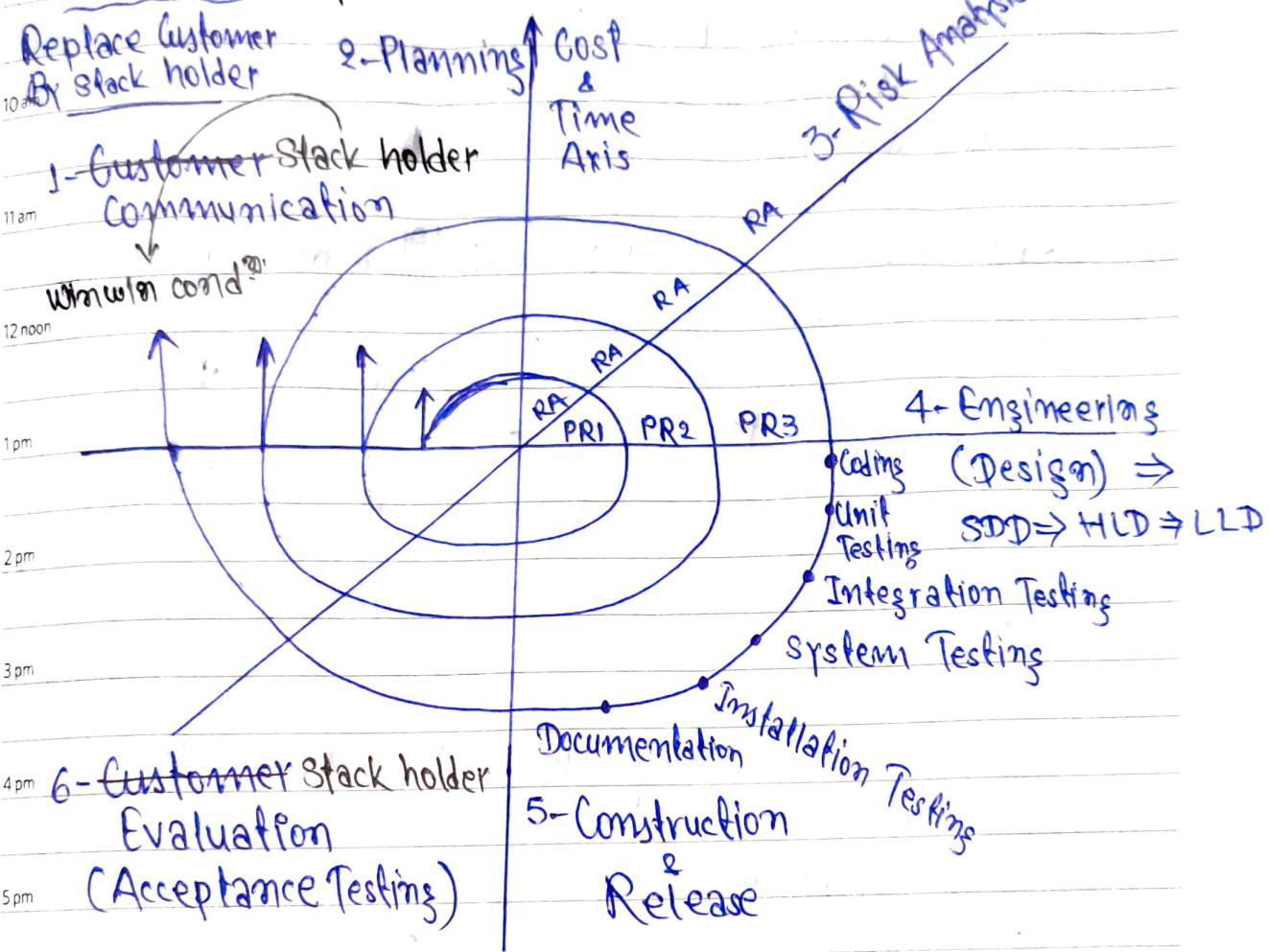
| M | T | W | F | S | S | M | T | W | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | 29 | 30 | 31 | | | | | |

TUESDAY • FEBRUARY

14

WK 07
(045-320)

9 am Win Win Spiral Model → (Extension of Spiral Model)



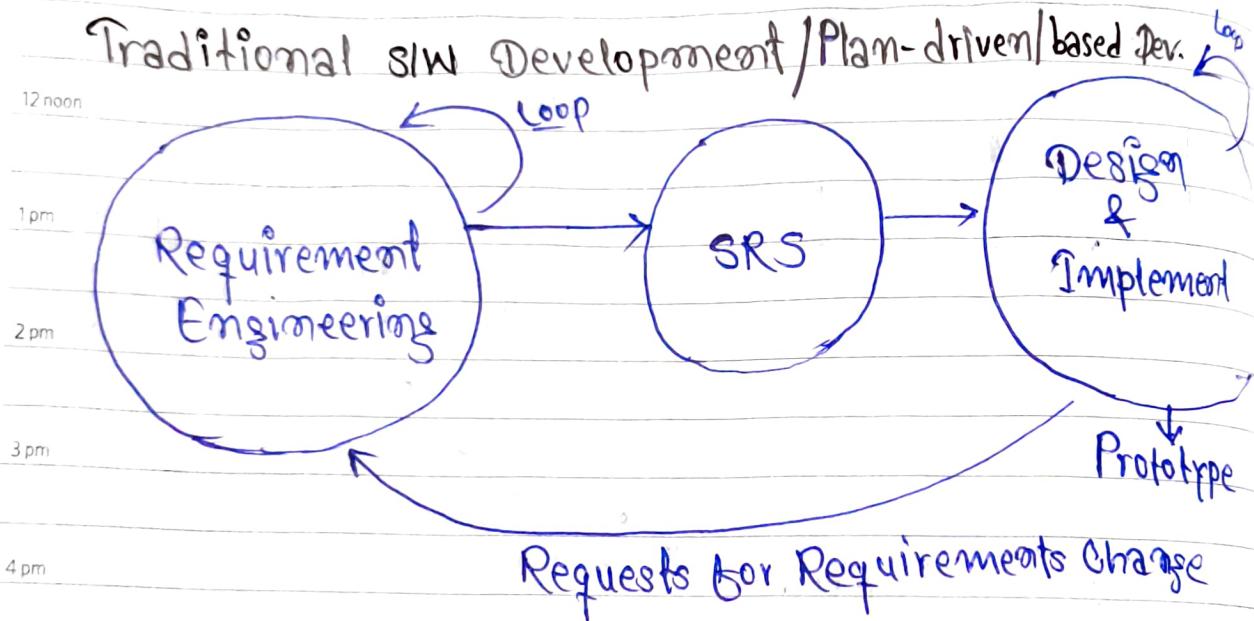
12/04/23
FEBRUARY • WEDNESDAY

15

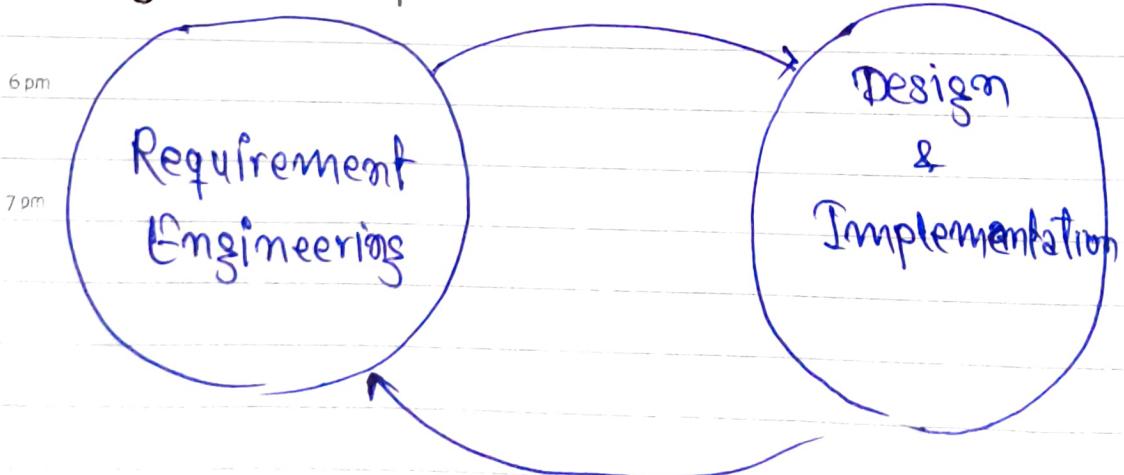
WK 07
(046-319)

| M | T | W | T | F | S | S | M | T | W | T | F | S |
|---|---|---|---|---|----|----|----|----|----|----|----|----|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| | | | | | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| | | | | | 27 | 28 | | | 21 | 22 | 23 | 24 |
| | | | | | | | | | 25 | | | |

- 9 am Agile Computing / Model
Extreme Programming (XP)
10 am Pair Programming
Scrub Model
11 am



Agile Development



MARCH - 2023

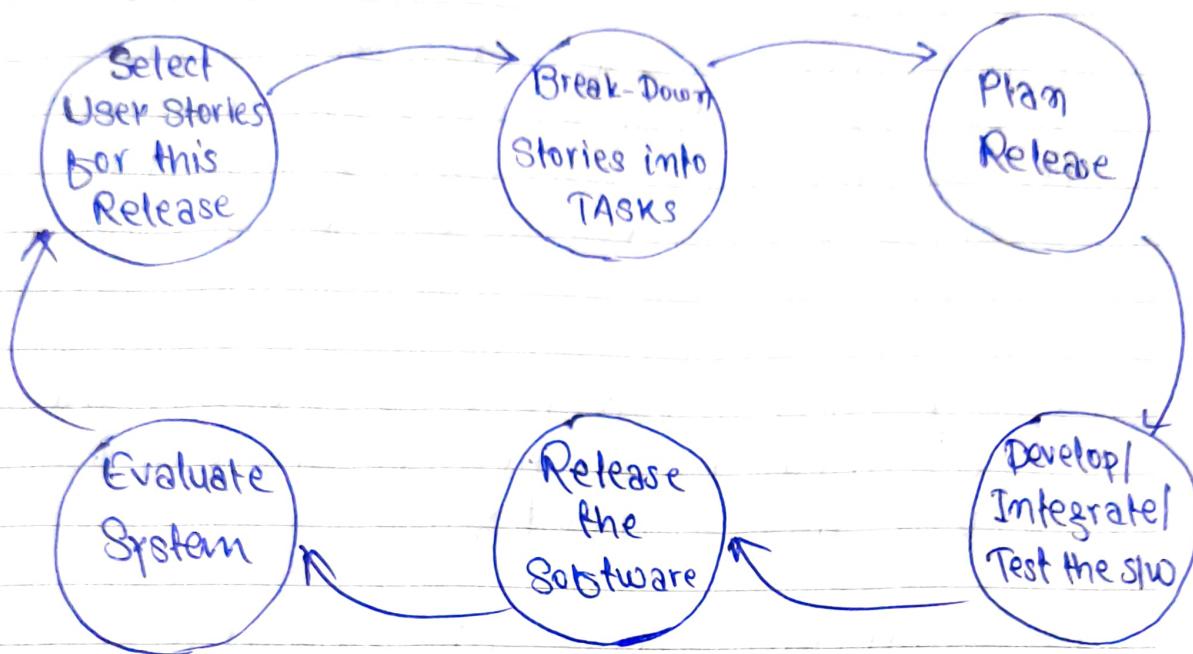
| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| - | - | - | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | - | - | - | - | - | - | - | - | - |

THURSDAY • FEBRUARY

16

AAKASH
(0471 HR.)

XP release Cycle



Extreme Programming (XP)

Extreme programming is the best known & most widely use of agile model. The name of ~~coined~~ BECK (2000). Because the approach was developed by pushing recognized by good practice such as ~~not~~ iterative development to extreme levels.

In extreme programming requirements are ~~expressed~~ explained as scenario.

Which are implemented directly as a form of test programmer what in pair & develop test for each task before writing the code. All test must be successfully execute when new code is integrated into system. There is a sort time that between the release of system

2023

FEBRUARY • FRIDAY

FEBRUARY - 2023

17

| M | T | W | T | F | S | S | M | T | W | T | F | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 27 | 28 | | | | | | | | | | | |

WK 07
(048-317)

9 am Advantage

→ Extreme programming reflect the principle of Agile method with no. of good practices

① Incremental development is supported through frequent release of the system.
small

② Customer involvement is support through continuous engagement of the customer with development team.

③ People not process are supported through pair programming, collective ownership collect the

④ Change in embrast through regular

a
test first development, refactoring to avoid the code the generation & continuous integration of new functionality

⑤ Maintaining Simplicity is support by const ~~not~~ refactoring the impure code quality

12/04/23

MARCH - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | | | | | | | | | |

SATURDAY • FEBRUARY

18

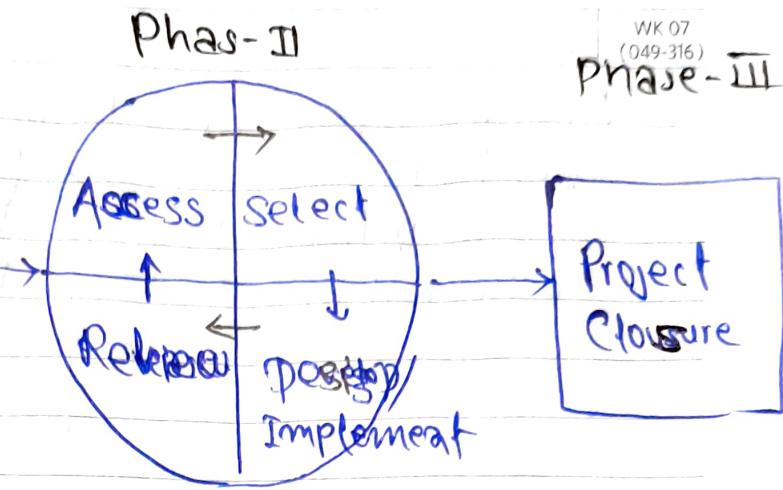
WK 07
(049-316)

Phase-III

Scrum Model

Sprint Cycle Phase-I

- Outline
- Planning & Architectural Design



Types

- Functional
- Non Functional

Requirement Engineering

1. Context Analysis
2. Requirement Elicitation
3. Requirement Assessment
4. Requirement Specification
5. Requirement Exploration

- functional req are concerned with functionality its service & behaviour.
- Non functional req are performance, user-friendly reliability, maintenance, security

→ Req also broadly categories

System Req

SUNDAY 19

Service Req

Behaviour Req

Performance Req

Operational Req

Constant Req

User Req Req

2023

12/04/23

FEBRUARY • MONDAY

20

WK 08
(051-314)

| FEBRUARY | | | | | | | | | | | |
|----------|---|---|---|---|----|----|----|----|----|----|----|
| M | T | W | T | F | S | S | M | T | W | T | F |
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| | | | | | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| | | | | | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| | | | | | 22 | 23 | 24 | 25 | 26 | 27 | 28 |

- 9 am Non fun req are broadly categories as
- Product Req
 - Organisational Req
 - External Req
- 10 am
- Portability
 - Reliability
 - Usability
 - Efficiency
 - Performance
- 11 am
- 12 noon

- 1 pm Org Req
- Delivery
 - Implementation
 - Standards
- 2 pm

- 3 pm External Req
- Ethical
 - Interoperability
 - Privacy
 - Safety
 - Legality
- 4 pm
- 5 pm
- 6 pm

S/W Requirements Sources

- 7 pm 1. Stake holder
- 2. Domain Expert
- 3. Developer
- 4. Tester
- 5. End User
- 6. Past Experience or case studies

MARCH - 2023

TUESDAY • FEBRUARY

21

WK 08
(052-313)

| T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
| 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 28 | 29 | 30 | 31 | | | | | | | | | |

SW Req eng is the branch of SW Eng focusing on real world goal, function

20/04/23

FEBRUARY • WEDNESDAY

22

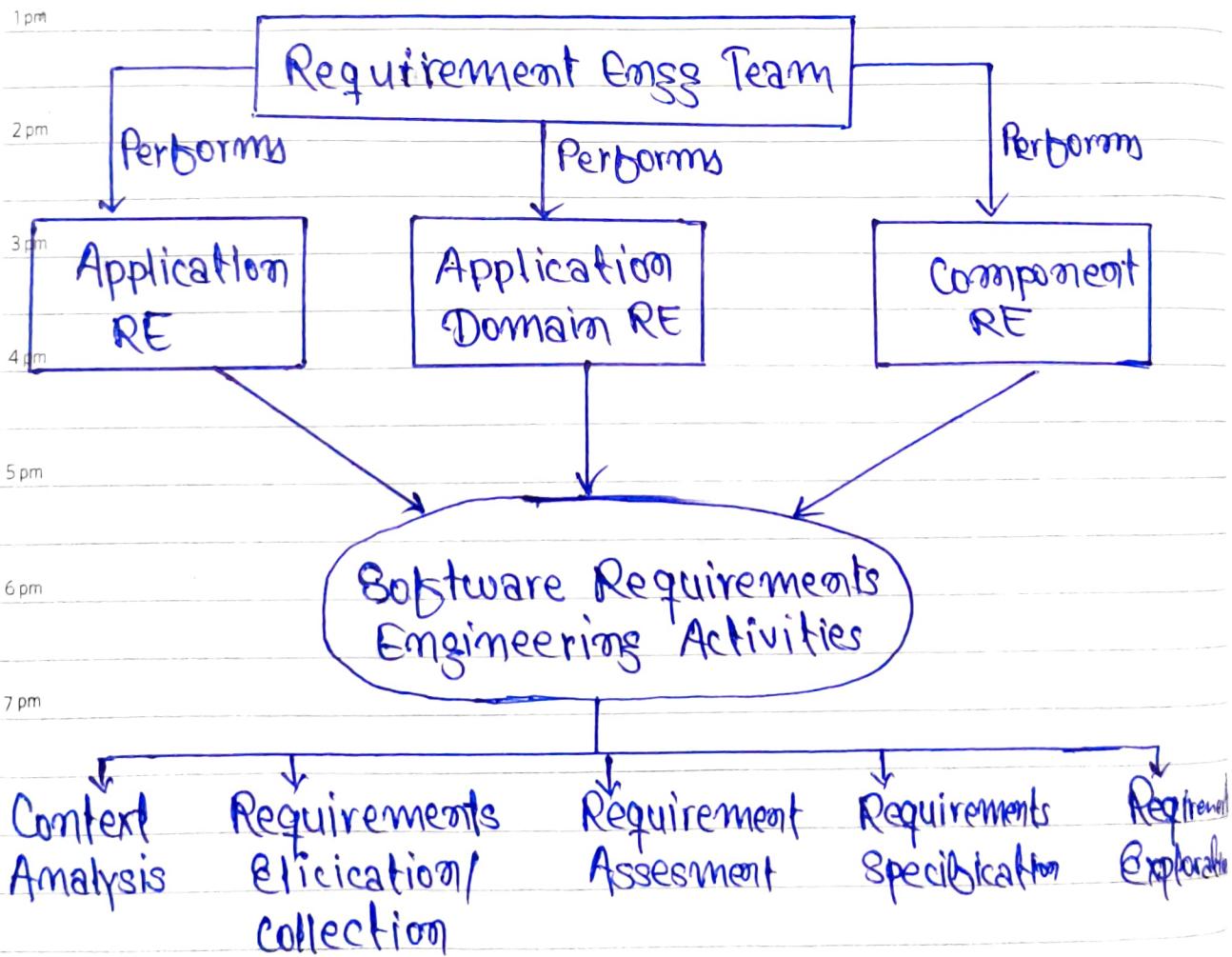
WK 08
(053-312)

FEBRUARY - 2023

| M | T | W | T | F | S | S | M | T | W | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 27 | 28 | | | | | | | | | | | |

9 am Software Requirements Engineering Subclasses

1. Business Req Enge
2. Product Line Req Enge
3. System Req Engs
4. Application Req Engs
5. Component Req Enge
6. Domain Req Enge



| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | | | | | | | | | |

23

WK 08
(054-311)

- Business Req Engg is the req engg sub activity consisting of cohesive collection of all the task that are primarily perform to produce the req & other work product req for business enterprise. The objective of this engg category to develop a business case & strategic vision & the future expectation of business enterprise.
- Product Line Represent a set of product that share a majority of feature. In a product line req engg the focus is to identify the product such as product family req & potential family members are identified, analysed & reusable domain framework is develop.
- System Req Engg, present a, balance view of system issues, concept models, technique & tools found in req engg research & practice. System Req Engg is a presented from business behavioural & slw engg ~~prac~~ prespractice to establish a general framework of a system

FEBRUARY • FRIDAY

FEBRUARY 2023

24

| M | T | W | T | F | S | S | M | T | W | F | S |
|----|----|----|----|----|----|----|----|----|----|----|----|
| | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| | | | 27 | 28 | | | | | | | 25 |

WK 08
(055-310)

→ Application Req Engg performs the req for the application that is req to be engineered. The objective of this engg is to produce & maintain a high quality req engg document stratified following -

① Starting all functionality (functional req)

② Data req

③ External req (Interface Req)

④ Quality req

⑤ Constraints of the development problem.

2 pm

→ Component Req Engg during which the req of an specific component are engineered. This activity dedicated to the component its scope, & boundary of delivery. The obj of this activity is to derive & maintained a complete set of high quality component of a system.

→ Domain Req Engg, is the req engg sub activity during which a reusable req for an application domain or engineered so that we can reuse any no. of time as per domain requirements.

MARCH - 2023

| M | T | W | T | F | S | S | M | T | W | T | F | S | S |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | |
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | | | | | | | | | |

SATURDAY • FEBRUARY

25

WK 08
(056-309)

Requirement Elicitation Important Method :

1. Observation
2. Interview
3. Survey & Questionnaire
4. Market Analysis
5. Cooperative Req Caputure (CRC)
6. Soft System Methodology (SSM)
7. Delphi Technique
8. Brain Stroming
9. Group Meeting

2pm

CRC



JAD (Joint Application Development)

Soft System Methodology (SSM)

SSM is a org approach for req engg approach it not only consider the technical problem alone but also it tries to improve the org structure of the system, where the product being develop. SSM done as following activities-

1. Problem situation unstructured
2. " " expressed
3. Building Root definition for the relevant system
4. Conceptual Modeling
5. Comparison with Real world
6. Implementation of feasible & desirable changes

SUNDAY 26

2023