

# UNIT-I

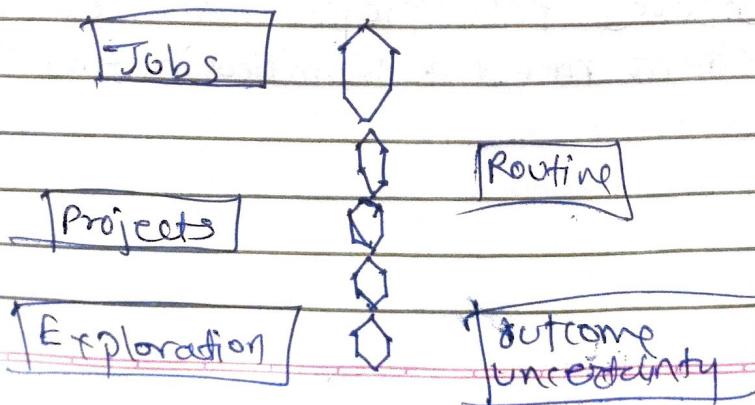
Page No.

## Intro. to Software Project Management .

Q] What is project? Why is software project management important.



- Project is set of distinct, complex & interrelated tasks with a single objective.
- It is temporary in nature, typically has project start date & end date and should get complete in specific time.
- Project planning helps in defining how to complete task before we begin. Planning is the crucial step here where it needs hard thinking.
- Other tasks like routine maintenance, will have been completed so many times that everyone is familiar with the procedure. planning appears to be unnecessary in these situations while process will help in training newcomers.
- Projects are different from routine operations as they involve non-repetitive work, limited resources & are often conducted for a client or stakeholder.



## Characteristics →

- 1) Project has deadline
- 2) Preparation is required
- 3) Work is performed for someone other than oneself
- 4) Certain Goal must be achieved
- 5) Work is carried out in stages
- 6) Limited resources are used
- 7) It's a big or complicated project

## Why Software project management is important?

- 1) Poorly managed software project can lead to huge financial losses
- 2) Many software project fails to meet goal without SPM
- 3) Triple constraint management → ~~for~~ SPM helps in balancing three constraints:
  - 1) Time
  - 2) Cost
  - 3) Quality
- 4) SPM ensures that project satisfies as per client need & submitted in timeline

## Q) Describe contract management in detail.

→

• Contract management means managing the agreement between two parties - usually a client & supplier.

• It is used when one company hire another company or developers to build their ICT (Information & Communication Technology).

# In client-side project manager:

→ Focuses on project deadline & budget

→ Does not handle technical details like coding & design.

→ Make sure the supplier is doing work according to agreement.

# The Supplier-side project manager:

→ Take care of technical tasks like planning, coding, testing, etc.

→ Make sure the work is done properly and delivered in time.

• The contract includes:

• Scope of work

• Deadline

• Budget

• Rules if something goes wrong.

• The contract helps in keeping a clear relationship between client & supplier.

• It also reduces risk of delays, confusion or extra costs.

Q] Explain the activities covered by software project management.

→ Software project management is not just about building software. It covers activities that goes beyond this.

There are three main processes

1) Feasibility Study → This determines if project has strong business case & worth pursuing. In this stage it collects the information about suggested requirements, which can be challenging at first. It includes in dept research about project.

## 2) Planning

- Once feasibility is approved, a detailed plan is prepared.
- It tells the team what to do, when how.
- Planning is not done at very beginning for a big project we start with small planning initially. It includes
  - time, cost, resources.
  - Roles & responsibilities
  - Breaking <sup>work</sup> tasks into tasks.

## 3) Project execution →

This is when the actual work is carried out. This phase often includes separate design and implementation stages. The goal is to deliver project acc to planned

## Sequence of Software development activities.

1) Requirement analysis.

2) Architecture design → Determine software design satisfy

3) Detailed design → Breaking down each software component into smaller, independently testable units.

4) Code & testing → Write code for each module, and also test it.

5) Integration → Combine diff components & test them together.

6) Qualification testing → Check if all requirements are achieved.

7) Installation → The process of deploying the new system.

8) Maintenance Support → provide service even after project deployment.

Q] How plans, methods & methodologies differ from each other?  
→

plan →

- The plan is blueprint of project that how it will be carried out
- It includes exact steps, start/end dates, responsibilities & required resources
- plan helps in organizing task, assigning work, tracking progress,
- A plan is used after a method is selected; and it tells team exactly what to do & when.

method

- It is general approach or technique used to perform a general activity.
- It tells how to do task, it does not tell who & when
- Methods can be reused & helps in producing consistent results.
- Method is not project specific. it can be used in many different projects.
- Plan uses method as its foundation.

### 3) Methodology:

- methodology is collection of related methods grouped together in structured framework.
- It defines principles, practices, tools, roles & techniques to be used throughout a project lifecycle.
- It gives a high-level structure to manage projects from start to finish.
- It shows overall way a project is planned, developed & delivered.

### Q) Explain ways to categorize software project

Software project can be grouped in 4 ways.

- 1) Compulsory user vs Voluntary user
- 2) Information system vs embedded system
- 3) Outsourced project
- 4) Objective-driven development.

#### i] Compulsory vs voluntary user

→ Compulsory: software

⇒ User ~~is~~ must use the soft for their work

Example → Employees using an attendance or billing system

• Easy to collect requirements from this users

## Voluntary users

- User choose to use software.
- Harder to collect requirement.
- People playing games or using shopping apps.
- Design & functionality should be good in order to satisfy this kind of customers.

## 2) Information system vs embedded system.

- It depends on type of system being built.

### Information system

- Used in business or office work.
- focus is on data & reports.
- Example - Banking Software.

### Embedded System:

- Control machines or physical devices.
- Must be fast & reliable.
- Example - Automatic door, ac control system.

## 3) Out sourced projects.

- One company gives project to another company.
- Some times when company has big project to build then it is always smart idea to give it to another company so that the project will be completed quickly; it saves money.

- Usually the companies give small & short project to other companies.
- It needs good communication b/w companies.

#### 4) Objective -driven development:

- This depends on Goal of project.

##### Product -Based Project:

- Goal is to make software product.
- Buyer decides whether to buy it or not.

##### Object driven project:

- Goal is to solve specific problem.
- Done in two step → 1) Research & give suggestions  
2) Build software based on those suggestions.

Q Explain in brief about stakeholders of a project.

→ Stakeholders are individual, group or organization that have interest in the project or are affected by its outcome.

- Stakeholders include project team members, client, users, management & even external contractors.
- It is important to identify stakeholders early in the project to ensure proper communication & expectations.
- They help in decision-making, requirement gathering, testing & feedback during the project lifecycle.

Stakeholders are classified into three stages:

1) Internal to the project Team:

- These are the people who are part of the core team & directly managed by the project manager
- ex → Developers, testers, designers & project leads

2) External to the project Team but within the same organization

- The stakeholders are in the same company but not part of the main project team
- ex → department Heads, domain experts

### c) External to the organization.

- These stakeholders belong to outside the organization like client or third party contractor.
- Different stakeholders may have different needs it is responsibility of manager to maintain this need.
- Proper communication strategy is prepared at the beginning to ensure smooth coordination between all stakeholders.

### Q Define business case & explain the concept of business case.

- Business case is a formal document or statement that justifies the start of a project by showing that its expected benefits to the costs & efforts are involved.
- It is usually built during feasibility study phase of the project.
- Business case includes details like:
  - How much money the project will cost
  - What benefits it will give (profit, saving)
  - When these benefits will receive.
- Cost benefit depends upon timely completion, earlier delivery means earlier return on investment (ROI).

- Business model also helps to make sometimes to show how the project will have earn money or give other benefits.
- Business case
- Business case should be checked regularly to ensure
  - Cost is not exceeding
  - Features are not removed too much
  - Delivery is not delayed.
- In short Business case helps to decide if the project is worth doing & keep checking if it's still useful as work continues.

Q] Identify the management responsibilities of the manager in view of software project management.

→ 1) planning the project.

→ what to do?

→ when to do?

→ who will do?

→ Gather requirements

2) Organizing Work

→ Manager establish proper structure, communication.

### 3) Directing the team

- provide technical & motivational direction to the team members
  - Guide the team to follow particular path.

4) monitoring progress

- Regularly check whether the project is on track in terms of schedule, cost & quality

5) Controlling the project  
→ Handout

- Handle situation where there is issue in project.
  - controls cost, quality & delay using project management techniques

6) Innovating when needed

- Introduce new tools, techniques or workflows to overcome challenges.

## F) Representing the project

- Acts as a middle person between developer, user, customers, supplier & stakeholders

→ Maintains communication

## 8) Managing project stage

- Initiation, execution, closure,  
planning monitoring delivering

Q Explain traditional project management & modern project management



### I) Planning

- Traditional : One big plan made at beginning
- Modern → small, flexible plans made step-by-step

### 2) Change Request

- Traditional : Changes not allowed
- Modern : Always welcome

### 3) Delivery

- Traditional : full project delivered at the end
- Modern : Delivered in modules

### 4) Customer Involvement

- Traditional → only at beginning
- Modern → Regularly involved

### 5) Risk handling

- Traditional → Only at beginning
- Modern → Throughout the cycle

### 6) Quality Check

- Traditional → Quality checked only at end
- Modern → Quality checked in every iteration

## 7] Team Structure

- Traditional → manager decides everything
- Modern → Team ~~works together~~

### 8) Documentation

- Traditional → Heavily focused on docs.
- Modern → less focus on docs, more on Software.

### 9) Example

- Traditional → Waterfall model
- Modern → Scrum, Agile

## UNIT-II

Page No.	
Date	

# Project Design & Evaluation

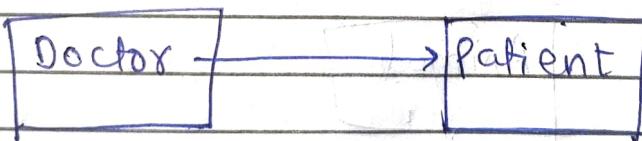
- Q) Explain relationships in class diagram with the help of suitable example.

Relationships in a class diagram shows how different classes are connected or interact with each other.

There are 5 main types

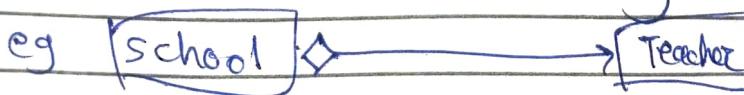
### 1) Association

- Association shows relationship between two classes that how one class is interacting with another class.
- It uses "uses-a" or "Knows-a" relationship
- Bi-directional or Uni-direction are two types
- Simple association & direct association are two types



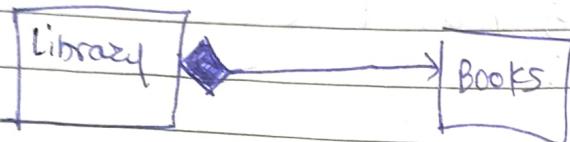
### 2) Aggregation ('has-a')

- It shows 'has-a' kind of relationship
- It is represented using hollow diamond



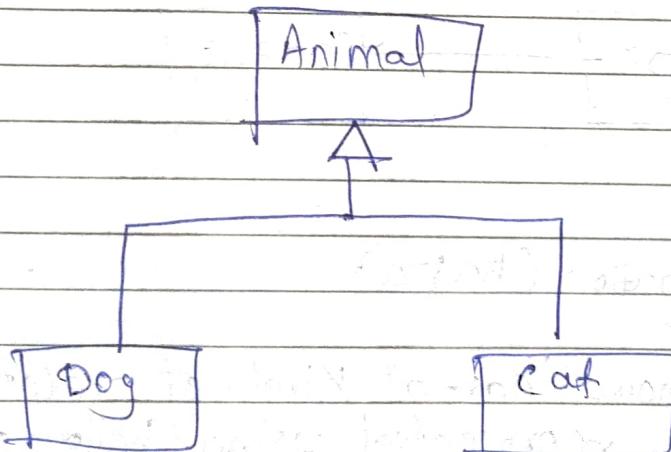
### 3) Composition (strong has-a)

- It is similar to aggregation but comes with strong relationship of has-a.
- It shows relationship where the part cannot exist without a whole.



### 4] Generalization

- It is hierarchical relationship where one class inherits properties and behavior from another.
- It shows parent-child class relationship.
- Represented by hollow triangle pointing to the parent class.

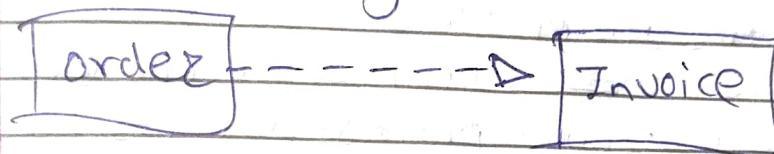


## 5] Dependency

- A temporary relationship where one class depends on another for performing some task.

- Uses temporary relationship.

- Represented using dotted arrow



Q) What is project Evaluation? Explain its importance.

→ Project evaluation is process of measuring success of a project, program or portfolio.

• It is done by collecting data about project on various aspects like cost, scope, time, risks & outcomes.

• The goal is to analyze whether the project is → Achieving its objectives,  
→ Running within budget.  
→ Completing within the schedule.

• Evaluation is performed during and after the project using tool like project dashboards & performance trackers.

• It also includes key questions like

→ What is working?

→ What needs improvement?

→ Is the project outcome valuable?

## project evaluation steps →

- 1) Planning
- 2) Implementation
- 3) Completing
- 4) Reporting & Dissemination.

## Importance of project evaluation.

- 1) Informs stakeholders:
  - keeps users, clients & stakeholders updated about progress, changes, risks
- 2) Improves financial Assessment
  - Helps determine financial value of project.
- 3) supports Risk management:
  - Helps team to handle risks.
- 4) Guides future projects:
  - post project evaluation help in learning from past mistakes.
  - improves future planning, execution & decision-making
- 5) Checks whether project goals are achieved or not.

Q] Describe cost benefit evaluation technique

→ I] Net profit

- 1) Net profit is the difference between total revenue and total cost of project
- 2) It tells how much profit a project generates in its entire lifetime
- 3) It is the simplest cost-benefit evaluation method
- 4) It does not consider the time value of money i.e. when cash is received.
- 5) It also ignores investment risk & size
- 6) A project with high net profit might be less desirable if returns come very late.

example.

A earns 5 lakh over 5 years

Project B earns 7 lakh over 2 years

Net profit prefer A over B

Advantage

Simple & easy to calculate

Disadvantage

Time manner is not considered

P.T.O.

## 2) Payback period

- 1) The payback period is the time required to recover the initial investment.
- 2) It measures how quickly the project will 'break-even'.
- 3) A shorter payback period is preferred to reduce risk.
- 4) It is easy to calculate and suitable when quick returns are important.
- 5) It ignores time value of money & does not consider projected lifespan.

### Example

If a project costs 100000 & return 20000 per year  
payback period =  $100000 \div 20000 = 5$  years.

### Advantages

- Lower investment risk

### Disadvantage

- Ignores profit after break-even

### 3) Return on Investment (ROI)

→ ROI shows how much percentage return is earned compared to investment.

$$\text{ROI} = \left( \frac{\text{Average Annual profit}}{\text{Total investment}} \right) \times 100$$

- It helps to compare different projects easily using % return.
  - It is useful in business decision-making.
  - But it doesn't consider timing of cash flows.
  - ROI does not include interest rates or inflation
- Ex. - If total profit = 50,000 & invest = 100,000,  
 $\text{ROI} = (50,000 \div 100,000) \times 100 = 50\%$ .

Jane (Ad & du)

## 4) Net present Value

- It consider both profitability & timing of cash flows.
- It calculates the present value of future cash inflows and subtract the investment.
- NPU uses discount rate to adjust future money to today's value.
- A positive NPU means project is profitable.

$$NPV = \underbrace{\sum_{t=1}^n \frac{\text{Value in year } t}{(1+r)^t}}_{\text{Present Value}} - \text{Initial Investment}$$

$t$  = time in years

$r$  = discount rate

Q3]

### Explain GQM paradigm

- The GQM paradigm is a goal-oriented software measurement approach. It helps ensure that the data collected is relevant, purposeful, and directly tied to project objectives.

- Goal Question metric has three basic elements

1)  $\textcircled{1}$  Goal    2)  $\textcircled{2}$  Question    3)  $\textcircled{3}$  metric

#### 1) $\textcircled{1}$ Goal

- It is description of what you want to achieve in the software process or product.

• It should be clear measurable & relevant

### • Example

• Improve the reliability of the payment module by reducing defect density by 20% in 6 months.

### 2) Question

• Goals are broken down into specific modules that identify what information is needed to determine if the goal is achieved.

#### Example

- 1) What is the current defect density?
- 2) How many defects are found per 1000 lines of code
- 3) What is causing error?

#### 3) Metric

• Identify the data to answer the questions

Ex

→ mean time of failure.

→ No. of defects found per cycle.

There are two approach in GQM

- 1) Top-down → Goal → Question → metric
- 2) Bottom up → metric → question → Goal.

## Q) four stages of AIGM

1) Planning

2) Definition → objective, questions & metrics are clearly defined

3) Data Collection: current data is collected

4) Interpretation → Analyse data to answer the question  
• verify whether the original goal has been achieved

Ex:

Goal → Improve System performance

Question → How many req can system handle?

metrics → Req per Sec, throughput

Q] Enlist the techniques of process analysis & explain in brief.

### ① Business process modeling Notation.

- i] BPMN is graphical representation of business process modelling.
- ii] It fills the gap between business stakeholders & IT developers.
- Various object used are → Artifacts, relationships, swimlanes.
- The biggest advantage of using BPMN is that it supports most of modelling tools & easier sharing.

### ② UML (Unified modeling Language)

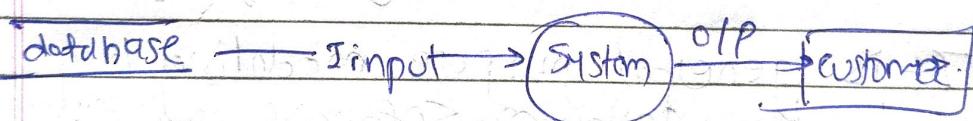
- It is general purpose modeling language used for designing Software system.
- It provides object like - state, object, Activity, class diagram.
- UML is of two types. → Behavioural & structural
  - Behavioural tells how the system is/shall perform.
  - Structural tells what the system consists of.

### 3) flow chart technique .

- It is visual representation of sequential flow & control logic .
- It shows activities & actions performing .
- It shows diamond as decision , oval for start/end  
rectangle for process setup  
Arrow → direction .

### 4] Data flow diagram .

- It shows how System process data in term of inputs & outputs
- Components include



There are two types in it

#### 1) Logical Data flow diagram (DFD)

It shows what happened ?

#### 2) physical dfD → how it happens ?

Q) What is process Analysis? Explain the Elements of Analytical model.

→ process analysis is the technical representation of system, acts as link bet<sup>n</sup> requirement phase & design model phase.

Elements of the Analytical model:

- 1) Data dictionary → It consists the description of all data object user or created by the software.
  - It stores the collection of data present in software. It acts as centralized depository.

2) ER diagram

- Entity relationship diagram shows the attributes & relationships between object.

3) Data flow diagram

- ~~Sto~~ It is visual model of how data moves through the system & how it transformed from input to output.

4) State Transition diagram

- Shows the state of a system & how it transitions bet<sup>n</sup> them based on events

5) Process specification.

- Describes each process shown in DFD in detail. It shows complete detail like how algorithm is used to transform

input to output

### 6.) Control Specification

- It's like state transition it shows what logic is used in response to events.

Q) Describe Sequence diagram & its notations with the help of suitable example.

- Sequence diagram shows how object interacts with order to time
- Sequence diagram is kind of interaction diagram.
- It shows the sequence of event that occurs to achieve functionality

Notations used

#### 1] Synchronous message

- The sender wait for receiver to complete the message

Notation → Solid line with filled arrow

#### 2] Asynchronous message

- The sender doesn't wait for the receiver to respond.

Notation → Solid black line with opened arrow ( $\rightarrow$ )

### 3) Return Message

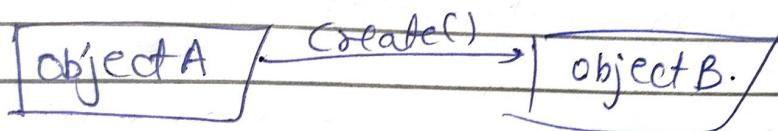
- Receiver sends back acknowledgement.

Notation → dotted line with arrow ( $\cdots \rightarrow$ )

### 4) Create object message

- One object creates new object during the interaction.

Notation → Arrow pointing to a new object's lifeline.



### 5) Destroy object message.

- An object is deleted or destroyed.

Notation: A large X symbol at the end of object's lifeline.

