

Module - II

CLASSMATE
Date _____
Page _____

CLASSMATE
Date _____
Page _____

Requirement (R)

* The description of features & functions, abilities of the target system. → (R).

* The expectation of users from the (S) product are communicated through (R).

* There are categories of (R) —



Functional (R)



Non-functional (R)

(R) which are not related to functional aspects of a (S).

• eg:- search option

eg:- security storage given to users to configuration.

Search from various

performance & cost.

Invoices.

② user should be able to ~~make~~ any report to the management.

→ (R) engineering :-

II (R) elicitation & analysis =

* Refers to the process of defining, documenting & maintaining (R) in the

engineering (R) process.

(R) eng provides the appropriate mechanism to understand what the

customers desires.

(R) eng process :-

- 1) It is a 5 step process which includes
- 2) Feasibility study & analysis.
- 3) (R) elicitation & analysis.
- 4) (S) (R) Specification (SRS).
- 5) (S) (R) validation.

I Feasibility Study :-

Objective of this is to create the reasons for developing the (S) that is acceptable to users possible to change & comfortable to established standards

- Technical (F)
- operational (F)
- Economic (F)

* This is also → gathering of (R).

dissemination gathering & (R)

classmate
Date _____
Page _____

classmate
Date _____
Page _____

- * Here, (R) are identified with the help of customers & existing system process.

- * Analysis of (R) starts with the (R) elicitation

III. SRS is a kind of doc which is created by a (S) analyst after the (R) collected from the various sources.

- * The (R) received from the customer written in ordinary lang (eng).

* After SRS is created the (R) validation discussed in the doc is to be validated.

(R) (R) validation =

→

Establishing the groundwork:

- * After SRS is created the (R) discussed in the doc due to be validated.

- * (R) validation techniques:

i) (R) review / inspection:

It is the developing systematic manual for the (R).

- * Any person who benefits directly/indirectly from the system being developed.
- * Recognizing multiple viewpoints:
- * Each of this stakeholders will

2) prototyping:

It is using an executable model of the system to check (R).

3) automated consistency analysis:

It is the checking for the consistency of structured (R) description.

4) (S) (R) management =

It is the process of managing changing (R) during the (R) engineering process & system development.

contribute data to the requirements eng process.

- Bez there are so many different stakeholders, the system's req will be examined from various perspective.
- As info is gathered from multiple viewpoints, emerging req may be inconsistent / contradictory.

* working towards collaboration:

- If there are 5 stakeholders involved in a (s) project, there may be 5 different opinions on the set of req.
- cust must work together as well as with (s) eng practitioners

(and vice versa) to create a successful system.

- A req eng's job is to identify areas of commonality as well as areas of conflict / inconsistency.

* Asking the 1st questions:

- 1st set of questions asked in context-free & focuses on the cust & other stakeholders.

* Asking the final questions:

- It is concerned with the effectiveness

of the commun activity itself.

⇒ Req Elicitation =

[req collect nwygndm]

- * Req. E is the most difficult, most error-prone & most commun intensive (crowdwaren) → development.
- * It can be successful only through an effective cust-developer partnership.
- * It is needed to know what the users really need

→ Req. E methods:

I Interviewing:

- objective of conducting an interview is to understand the cust's expectation from the (s).

II Brainstorming sessions:

- It is a grp technique & it is intended to generate lots of new ideas hence providing a platform to share views. every idea is documented so that everyone can see it.

The Facilitated application specification technique
 objective is to bridge the expectation gap - difference b/w what the developers think they are supposed to build & what user think they are going to get.

IV Quality function deployment:

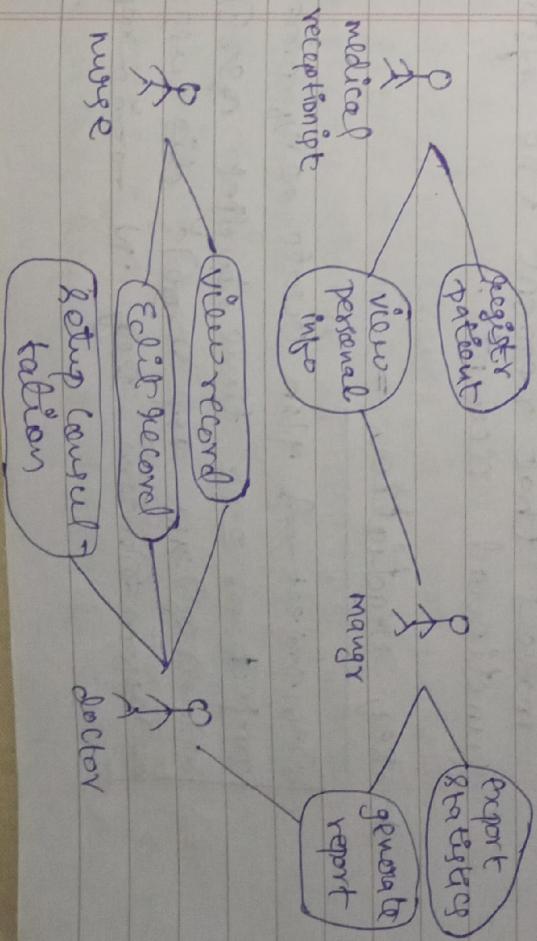
Just satisfaction is of prime concern.

V Use case approach:

This technique combines txt & pictures to provide a better understanding of the req.
 It describes the 'what' of a system ie not 'how'.

⇒ Developing use cases =

- * In S.Eng, use case is a list of actions evnt steps, typically defining the interaction b/w a role & a system to achieve a goal.
- * In Sys. Eng, use cases represent missions / stakeholder goals.
- * products to achieve missions
- * Each use case should be documented with a textual description ..



⇒ Building req model =

- * Req modelling in sizing is essentially the planning stage of a (S) application system.

- * A client approaches a (S) development team to create an app in [system] from scratch / update an existing one.

- a) class-oriented models:

models that represent obj-oriented class (attributes (obj property) & op (obj action (s)))

- b) Flow-oriented models: models that represent the system's functional elements.

→ Req Negotiation =

that project → plan create a compromise between what user want & what dev team can do [negotiation req. N - min. need]

→ Req Validation =

- * It is a process of ensuring the specified req meet the user needs.
- * It is concerned with finding problems with the req.
- * These probs can lead to expensive rework costs when these they are discovered in the later stages.
- * Cost of fixing a req prob by making a system change is usually much greater than repairing code errors.
- * Bcz. a change to the req usually means the design & implementation must also be changed & re-tested.
- * During req validation process, different types of checks should be carried out on the req. 5 types of checks —

a) Validity checks :

The U's proposed by stakeholders should be aligned with what the system needs to perform.

b) Consistency checks :

Req in the doc shouldn't conflict or different descriptions of the same function [Same word mean different description error]

c) Completeness checks :

Doc should ~~contain~~ include all the req and constraining

d) Realism checks :

Ensure the req can actually be implemented using the knowledge of existing technology, budget, schedule, etc

e) Verifiability :

Req should be written so that they can be tested.

⇒ Req analysis =

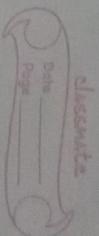
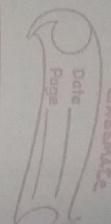
* process of defining the expectations

→ Req Analysis process :

1) Eliciting req :
process of gathering req by communicating with the user.

2) Analyzing req :
helps to determine the quality of the req.

Involves identifying whether the req are unclear, incomplete & contradictory. These issues are resolved before moving to the next step.



3) Req modelling:
here, req are usually document in different format like use case, use stories in natural lang doc.

4) Review & retrospective:

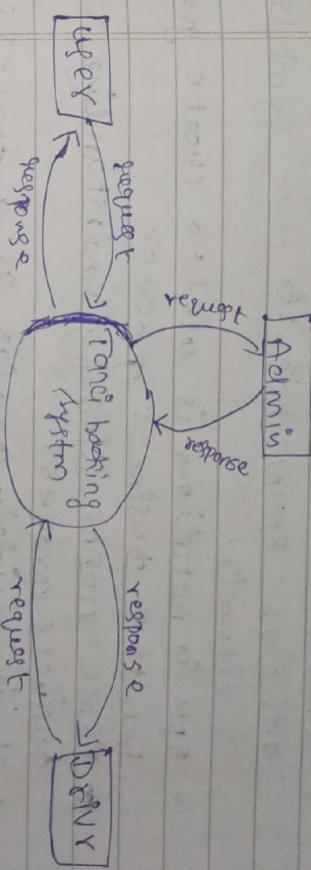
This step is conducted to reflect on the previous iterations of req gathering to make improvements in the process going forward.

→ Req Analysis Techniques =

i) DFD (Data Flow Diagram):

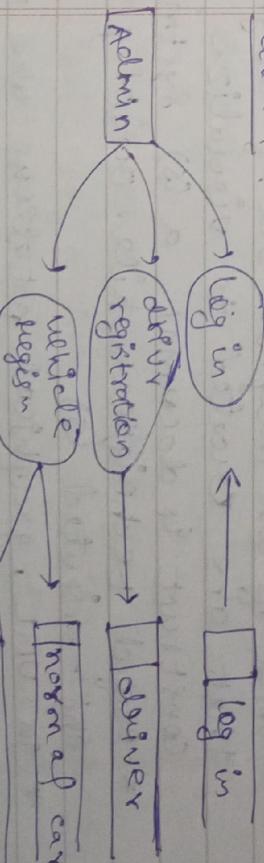
- * used to visually represent system of processes that are complex & difficult to describe in txt.
- * represent the flow of info through a process [a system]
- * it describes various entities & their relationships with the help of standardized notations & symbols.
- * DFD is displayed in a hierarchical format. [if data flow model reflects the entire system].

Also → level o DFD | context diagram.
* eg → Taxi booking system level o:

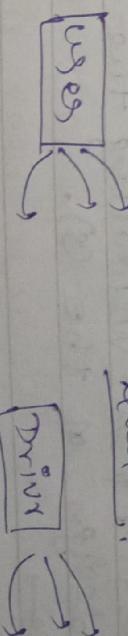


Admin }
User } 3 modules
Driver }

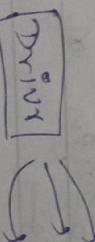
Level 1:



Level 2:



Level 3:



2) BPMN [Business process modeling notation]:

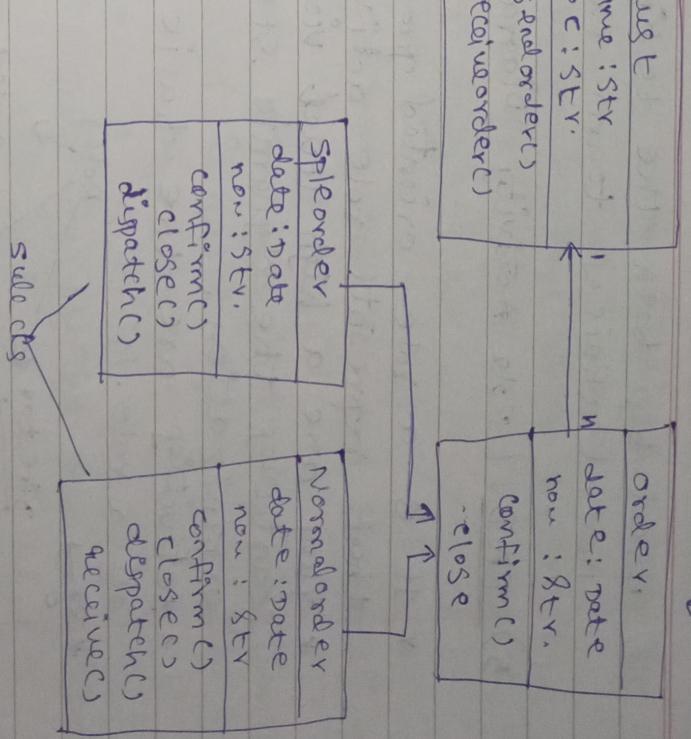
- * Similar to creating process flowcharts although BPMN has its own symbols & elements.
- * BPMN notation is used to create graph for the business process.
- * These graphs simplify understanding the business process.
- * It is widely popular as a process improvement methodology.

3) UML [unified modeling lang.]:

- * Consist of an integrated set of diagrams that are created to visualize, construct & document a (S) system.
- * useful technique while creating obj.-oriented (S).
- * In UML, graphical notations are used to represent the design of a (S) project.
- * Algo help in validating the architecture design of the (S).
- * eg →

4) Flowchart :

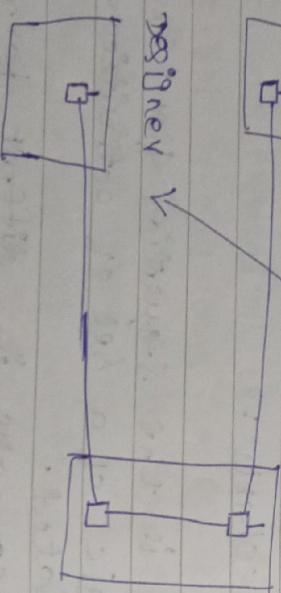
- * It is the sequential flow & control logic of a set of activities that are related.
- * They are in different format such as linear, cross-functional & top-down.
- * It can represent system interactions, data flows, etc.
- * It is easy to understand & can be



used by both the technical & non-technical team members.

5) RAD [Role Activity Diagrams]:

- * It is a role-oriented process model that represents role-activity alignment.
- * They are a high level view that captures the dynamic structure of an organization.
- * Activities are the basic parts of a role.
- * e.g. →
 - Director
 - Roles
 - Project manager.



6) Gap Analysis:

- * It is a technique which helps to analyze the gaps in performance of a (S) system.
- * Gap denotes the difference b/w the present state & the target state.
- * It also → need analysis
- * involves the steps that are to be taken to ensure all the business reqs are met successfully.

to know what is scheduled to be completed by which date.

* The start & end dates of all the tasks in the project can be seen in a single view.

7) IDEF [Integrated def for modeling]:

- * Represents the OS & their relationship to child & parent systems with the help of a box.
- * Provides a blueprint to gain an understanding of the organization system.