

Unit-2 Requirements Analysis & Specification

Ques - 1) Differentiate between Functional Requirement & non-functional requirement.

Ans:

Functional Requirement

(1) A Functional requirement defines what a system should do.

2) It specifies "what should the software system do".

3) Functional requirement is specified by user.

4) It is captured in use case.

5) Defined at a component level.

6) Helps you verify the functionality of the software.

7) Usually easy to define.

8) ex: (1) System shuts down in case of cyber attack.

9) Authentication of user.

Whenever he/she logs into system.

Non-Functional Requirement

① A non-functional requirement defines the quality attribute of a Software System.

2) It places constraints on "how should the software system fulfill the functional requirement".

3) Non-functional requirement is specified by Technical people eg: Architect, Software developers.

4) It is captured as quality attribute.

5) Applied to a system as whole.

6) Helps you to verify the performance of software.

7) Usually more difficult to define.

8) ex: (1) emails should be sent with a latency of no greater than 12 hrs from such activity.

9) The processing of each request should be done within 10 seconds.

Ques-9) List & Explain the characteristics of SRS.
(System Software requirements Specification)

Ans!

- (1) Correct
- (2) Unambiguous
- (3) Complete
- (4) Consistent
- (5) Specific
- (6) Traceable

- (1) **Correct** - The SRS should be made up to date when appropriate requirements are identified.
- (2) **Unambiguous** - when the requirements are correctly understood then only it is possible to write an unambiguous SRS.
- (3) **Complete** - To make the SRS complete, it should be specified what a software designer wants to create a software.
- (4) **Consistent** - It should be consistent with reference to functionalities identified.
- (5) **Specific** - The requirements should be mentioned specifically.
- (6) **Traceable** - what is the need for mentioned requirement? This should be correctly identified.

Ques - 8) List the guidelines for writing user requirement

Ans'

- Prepare a standard format and use it for all requirements.
- Apply consistency in language. Use
 - 'shall' for mandatory requirement
 - and 'should' for desirable requirement
- The text which is mentioning the key requirements should be highlighted.
- Avoid the use of Computer jargon (computer terminologies). It should be written in simple language.

{ optional }

For Example: Consider a spell checking and correcting system of a word processor. The user requirement can be given in natural language as

- The system should possess a traditional word dictionary and user supplied dictionary. It shall provide a user-activated facility which checks the spelling of words in the document against spellings in the system dictionary and user-supplied dictionaries.
- When a word is found in the document which is not given in the dictionary, then the system should suggest 10 alternative words. These alternative words should be based on a match between the word found and corresponding words in Dictionary.



when a word is found in the document which is not in any dictionary, the system should propose following options to user

- (1) Ignore the corresponding instance of the word & go to next sentence
- (2) Ignore all instances of word
- (3) Replace the word with a suggested word from the dictionary.
- (4) Edit the word with user-supplied text.
- (5) Ignore this instance and add the word to a specified dictionary.

Ques-7) Difference User Requirement & System Requirement

User Requirement

System Requirements

- | | |
|--|--|
| (1) The focus of this type of requirement is on the problem domain. | (1) The focus of this type of requirement is on solution domain. |
| (2) These requirements describe what effects need to be achieved. | (2) These requirements describe what software the software must do. |
| (3) user requirement tell what application should do to satisfy users needs. | (3) System requirements tell a system should have to be able to run program. |
| (4) The user defines his/her requirements for the system. Hence it is not necessary to incorporate all the user requirements in the system contract. | (4) These requirements must be incorporated into system contract. |
| (5) It is expressed in natural language, often in non-technical terms. | (5) It is expressed in technical terms, using specific measurements and standards. |
| (6) It is evaluated based on user satisfaction, feedback. | (6) It is evaluated based on system performance, reliability and adherence. |
| (7) It may change more frequently due to evolving user needs and market trends. | (7) Tend to be more stable and less prone to frequent changes once established. |

Ques - 6) Classify the following as functional / non-functional requirements for banking system.

- (a) Verifying back balance
- (b) withdrawing money from bank
- (c) Completion of transactions is less than one second.
- (d) Extending the system by providing more tellers for customers.
- (e) Providing 24 hrs service.

Ans: Functional Requirements:

- (b) withdrawing money from the bank
This is the functional requirement as it specifies a specific action or function that the system must perform, which is allowing customers to withdraw money.

- (d) Extending the system by providing more tellers for customers.
This is a functional requirement as it specifies a specific action or function that the system must perform, which is allowing customers to withdraw money.

(d) Extending the system by providing more tellers for customers:

This is functional requirement as it describes a capability that the system should have, which is the ability to extend the system by adding more tellers to serve customer.

* Non-Functional Requirements :

(a) Verifying bank balance:

This can be classified as a non-functional requirement while it does involve a specific function, the emphasis is on how efficiently and reliably the system performs this function, rather than the function itself.

(c) Completion of transaction in less than one second

This is non-functional requirement because it specifies a constraint on the system's performance (response time) rather than a specific function. It focuses on speed at which transactions should be completed.

(e) Providing 24 hrs Service

This is non-functional requirement as it specifies a characteristic of system's availability as it should be available for customer service 24 hrs.

ques - 5) What is requirement elicitation? Briefly describe the various activities performed in requirements elicitation phase with an example of watch system that facilitates to set time and alarm.

Ans:

Requirements elicitation means requirements discovery. Requirement elicitation is very difficult task.

To overcome sometimes customer find it difficult to communicate with the system engineer about their needs. Sometimes customer may have got some conflicting requirements. This ultimately result in specifying ambiguous requirement.

To overcome these problem the requirements gathering must be done very systematically.

Requirement After identifying all possible requirements the analysis on these requirements can be done.

* Various activities performed in requirements elicitation phase of watch system that facilitates to set time & alarm.

(1) Requirements discovery - by effective communication with customers the requirements can be identified.

(2) Requirements classification - All unstructured requirements can be classified systematically.

and can be arranged in functional and non-functional requirements.

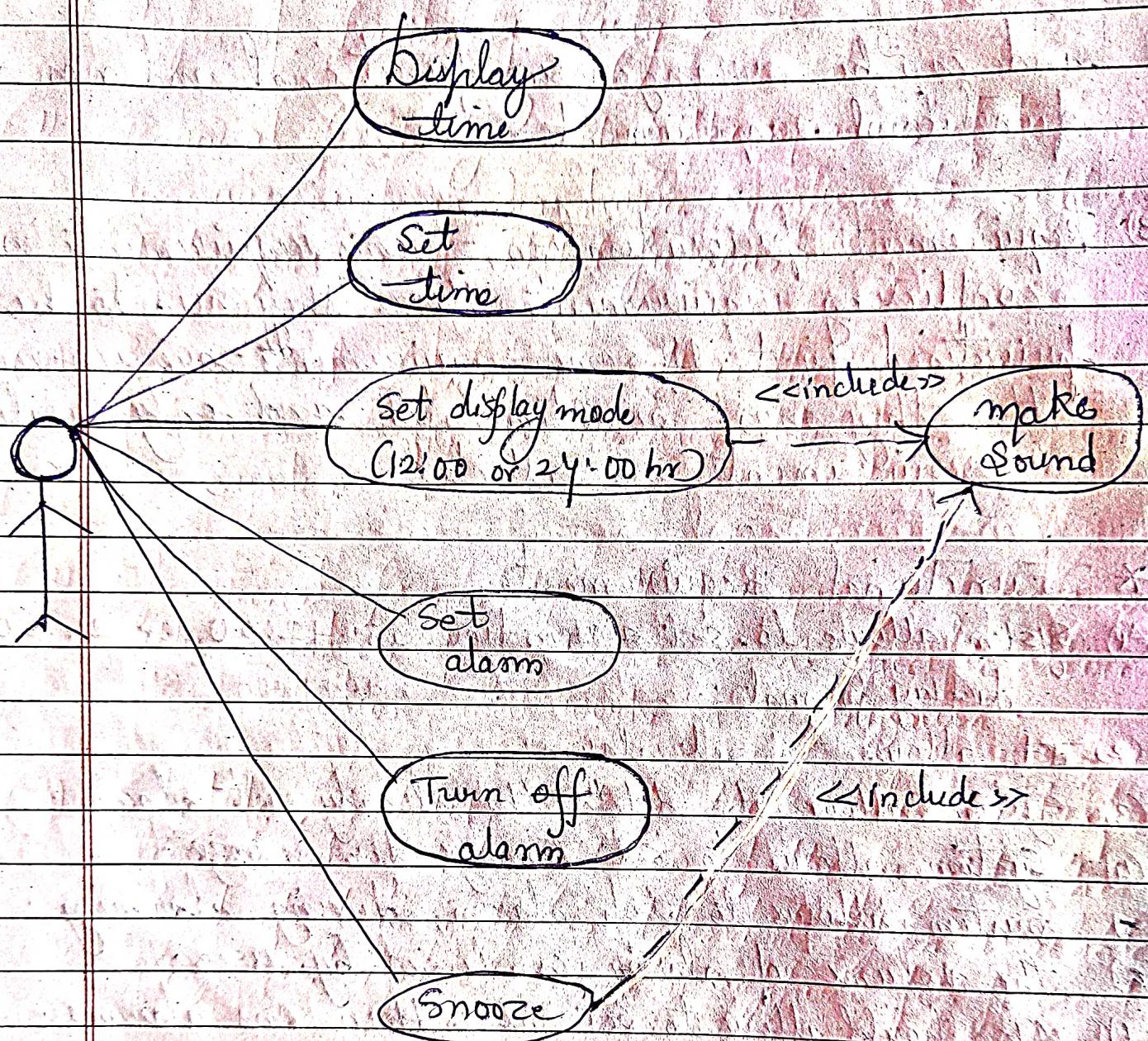
- (3) Requirements prioritisation - Some requirements are conflicting requirements. Hence they need to be prioritised first. If some requirements are un-realistic then they must be eliminated and only realistic requirements are collected.

* Functional Requirements

- (1) System allows to set alarm time in 12:00 or 24:00 hr display format.
- (2) It allows to set alarm.
- (3) System allows to delete the already set alarm.
- (4) System allows to set timings for snooze.
- (5) System allows to stop alarm, during alarming.
- (6) The volume of alarm can be set by user.
- (7) It should allow to set more than one alarms.

* Non-Functional Requirements

- (1) It should indicate battery low message if battery low.
- (2) The system must be reliable and nobody should hack the system.



ques - 1) Explain the Feasibility studies . what are the outcomes ? Does it have implicit or explicit effects on Software requirement collection ?

Ans:

A feasibility study is a study made to decide whether or not the proposed system is worthwhile.

The focus of feasibility study is to check :

- If the system contributes to organizational objectives
- If the system can be engineered using current technology
- If the system is within the given budget
- If the system can be integrated with other useful system

The implementation of feasibility study is based on the information assessment, information collection and report writing

Feasibility study should be done with the help of project managers who is going to handle that particular project , software engineer who are about to develop that system , technical experts and customers who will be using the system . Typically the feasibility study should be completed within two - three weeks .

* Outcomes

While performing the feasibility study, following questionnaires to the people in the organization should be asked.

- (1) What if the system wasn't implemented?
- (2) What are current process problems?
- (3) How will the proposed system help?
- (4) What will be the integration problems?
- (5) Is new technology needed? with what skills?
- (6) If the system can be integrated with other used systems

Types of Feasibility Study →

- (1) Technical feasibility — It is the measure of technical resource such as hardware components, software techniques and skilled persons.
- (2) Economical Feasibility — It is the measure whether finances or funds are available for proposed system.
- (3) Operational feasibility — It is a measure of how well the solution of problems or a specific alternative solution will work in organization.
- (4) Schedule Feasibility — It is the establishment of time limit for completion of the project. This kind of feasibility is dependent upon available manpower and economical support for project.

Thus the feasibility study helps in understanding the requirements of system. Hence feasibility study affects the requirement collection implicitly or explicitly.

Ques-3)

what is requirements engineering? Explain in detail the various processes in requirement engineering

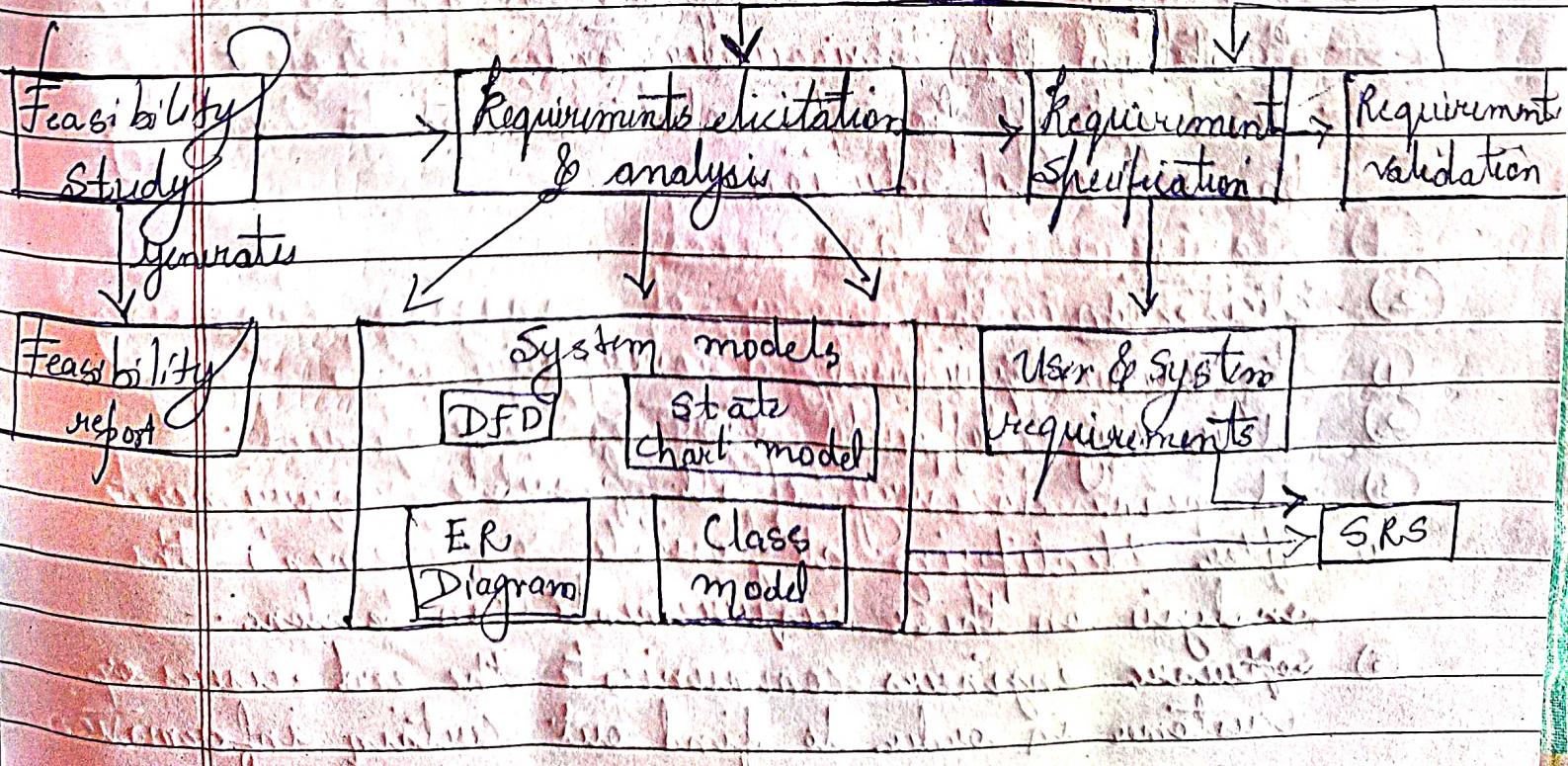
Ans:

A requirement engineering is a process in which various activities such as discovery, analysis and validation of system requirements are done.

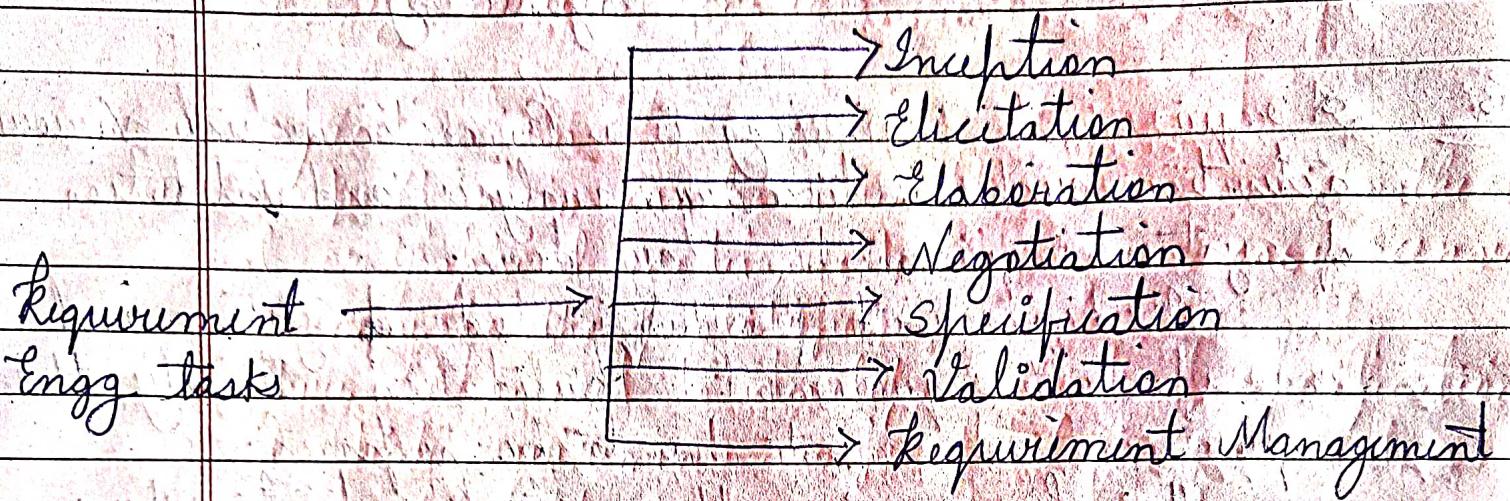
It begins with feasibility study of system and ends up with requirement validation. Finally the requirement document has to be prepared.

This process is a three stage activity where the activities are arranged in the iterative manner.

In the early stage of this process most of the time is spent on understanding the system by understanding the high-level non-functional requirements and user requirements.



Requirement Engg Process performs following Seven distinct functions -



(1) Inception

- (1) It means specifying the beginning of Software Project
- (2) Most of the Software project gets started due to business requirements
- (3) The purpose of the Inception is to
 - Establish the basic understanding of Project
 - Find out all possible solutions and to identify the nature of Solution

(2) Elicitation

- (1) It means requirement discovery
- (2) It is very difficult task
- (3) It means discovery of all possible requirements
- (4) After identifying all possible requirements the analysis on these requirements can be done.
- (5) Software engineers communicate the end-users or customer in order to find out certain information

such as application domain, expected services from the system, the expected performance level of the system.

Requirement Elicitation & Analysis Process

- (1) Requirement Discovery
- (2) Requirements Classification & Discovery
- (3) Requirement Prioritization
- (4) Requirement Documentation

(3) Elaboration

- (1) Elaboration is an activity in which the information about the requirements is expanded and refined
- (2) The goal of elaboration activity is to prepare a technical model of software functions, features & constraints
- (3) Elaboration consists of several modelling & refinement tasks. In this process several user scenarios are created and refined.
- (4) During elaboration, each user scenario is parsed and various classes are identified. These classes are nothing but the business entities that are visible to end user.

(4) Negotiation

- (1) Sometimes customer may demand for more than that is achieved or there are certain situations in which customer demands for something which cannot be achieved in limited business resources.

(2) To handle such situations requirement engineer must convince the customers or end users by solving various conflicts.

⑤ Specification

- (1) A specification can be a written document, mathematical or graphical model, collection of use case scenarios or may be the prototypes.
- (2) There is a need to develop a standard specification in which requirements are presented in consistent and understandable manner.

⑥ Requirement Validation

- (1) Requirement validation is an activity in which requirement specification is analyzed in order to ensure that the requirements are specified unambiguously.
- (2) If any inconsistencies, omissions and errors are identified then those are corrected or modified during validation.

⑦ Requirement Management

It is the process of managing changing requirements during the requirements Engg. process and system development.

Ques - 1

List the stakeholders and all types of requirements for an online train reservation system.

* Stakeholders :

- (1) Passengers.
- (2) Railway operators.
- (3) Administrators.
- (4) Payment Gateway Providers.
- (5) Regulatory Authorities.
- (6) Customer Support and Helpdesk.
- (7) System Developers and IT Team.
- (8) Marketing Team.
- (9) Accessibility Advocates.
- (10) Third-party Service Providers.
- (11) Financial Analysts.
- (12) Business Executives.
- (13) Competitors.
- (14) Environmental Authorities.

* Functional Requirements :

- (1) Ability to search for train schedules.
- (2) Ticket booking functionality.
- (3) Payment processing features.
- (4) Reservation confirmation and ticket generation.
- (5) Seat availability & selection.
- (6) User account management.
- (7) Administrative tool for managing schedules and pricing.

- (8) Integration with third-party services
(e.g. hotel, transportation)
- (9) Customer support and helpdesk features

* Non-Functional Requirements :

- (1) Usability requirements for user-friendly interface
- (2) Performance requirements for quick response time
- (3) Scalability to handle a large number of concurrent users
- (4) Security measures to protect user data
- (5) Compliance with data privacy regulations
- (6) Accessibility standards to accommodate individuals with disabilities
- (7) Environmental Sustainability Considerations
- (8) System maintenance & updates
- (9) Backup

* Business Requirements:

- (1) Market share targets
- (2) Revenue growth objective
- (3) Branding & marketing requirements
- (4) Competitive analysis & feature alignment

Ques - 2)

Show the template of IEEE standard software requirements documents.

Ans:

Document Title
Author(s)
Affiliation
Address
Date
Document Version

(1) Introduction

- * Purpose of this document → Describes the purpose of the document
- * Scope of this document → Describes the scope of this requirement definition effort. This section also details any constraints that were placed upon the requirement elicitation process, such as schedules, cost
- * Overview - Provides a brief overview of the product defined as a result of the requirement elicitation process

(2) General Description

Describes the general functionality of the product such as similar system information, User characteristics, user objectives, general constraints placed on design team.

③ Functional Requirements

- (1) Description
- (2) Criticality
- (3) Technical issues
- (4) Cost and Schedule
- (5) Risks
- (6) Dependencies with other requirement

④ Interface Requirements

- * User Interface
- * Hardware Interface
- * Communication Interface
- * Software Interface

⑤ Performance Requirements

⑥ Design Constraints

⑦ Other Non-Functional Attributes

- * Security
- * Binary Compatibility
- * Reliability
- * Maintainability
- * Portability
- * Extensibility
- * Usability