Requirement Engineering

By - Er. Suraj Awal

Feasibility Study

- A study that is done in earlier stage of requirement engineering
- Study of whether the requirements are practically possible
- Determine whether system development is feasible or not in terms of technology and economy
- Should be cheap, quick and focused
- Determination of whether the user requirements can be satisfied using current technologies

Feasibility Study

Technical Feasibility

- Determine whether the current technology can handle the problem statement
- Determine whether the organizational expertise is enough
- Evaluation of current hardware and software to meet the system requirements
- Selection of best technical approach

Legal Feasibility

- Evaluation of whether the current legislature allows the production of the proposed system

Feasibility Study

Operational Feasibility

- Evaluation of how well the system will solve the elicited problem or requirements
- How well the proposed system fits into the business environment
- Focus on development schedule, delivery date, existing business process and so on

Financial Feasibility

- Evaluation of cost incurred and whether the cost lies within the limit
- Judged using parameters like total estimated cost, existing investor, cash flow and profitability, estimated cost benefit ratio, payback period

Requirement Elicitation and Analysis

- Performed after feasibility study
- Iterations : Requirement discovery or gathering, Requirement classification, Requirement prioritization and Requirement specification

Requirement Gathering Process

- Requirement gathering is the first step of any software development process
- All the requirements of the software system are gathered via various methods to generalize what the system will do
- Provides the foundation on what the software is about and what features are required by the software to provide user satisfaction

Requirement Gathering Techniques

The various techniques to gather requirements of a software system are as follows:

1. Interview
2. Questionnaire
3. Task Analysis
4. Domain Analysis
5. Brainstorming
6. Prototyping
7. Observation

Problems in Requirement Gathering

- The customers are unknown about technical capabilities and may demand unrealistic requirements which are not technically feasible
- The customers generally express their requirements in generalized and natural form, which is difficult to understand and difficult to realize technically
- Expression of same requirement by different customers may be different and are difficult to co-relate

Problems in Requirement Gathering

- Requirements of customer changes over time, which makes it difficult to accommodate those changes within the software system

Functional and Non-functional Requirements

Functional Requirements

- Requirements that satisfies the functionality of the software
- Indicates user satisfaction towards the software system
- Eg : Actual intended functionality of the system software

Non-functional Requirements

- Requirements that satisfies the software properties
- Indicates various software performance metrics
- Eg : Reliability, security, response time and so on of the software product

User and System Requirements

User Requirements

- Describe what the system is intended to do in terms of functional and non-functional requirements
- Understandable by the general user
- Represented in natural language, tables, graphs

System Requirements

- Extended user requirements used by the software engineers for system design
- Represented in technical way
- Understandable by engineers
- Includes how user requirements can be implemented in the system
- Defined using system models

Software Requirement Specification

- A document or an official statement describing what the system developers should implement
- Includes both user and system requirements
- Serves as an agreement between developers and customers on what the software system will do
- It acts as an output of the requirement phase of the software development process
- Provides foundation for other software development phases

Software Requirement Specification (Purposes)

1. Help in proper design of the software system
2. Collaboration between engineer and user to understand the system properly
3. Decomposing the system into modules and components
4. To validate all the requirements are correct and complete
5. Lay a foundation for system design phase
6. Enable to generate rough estimate about time and cost

Software Requirement Specification (Characteristics)

- Complete (Requirements clearly defines what the software is intended to do)
- Consistent (Subset of individual requirement defined should not conflict)
- Correct (All user requirements are stated)
- Modifiable (Changes in user requirements should accommodate easily)
- Ranked (Each requirement should be clearly identified and their importance should be mentioned)

Software Requirement Specification (Characteristics)

- Unambigious (Every stated requirements has only one interpretation)
- Traceable (Source of each requirement is clear and facilitates the reference of each requirement in future)
- Verifiable (Specified requirements can be verified to check whether the final software meets those requirements)

Software Requirement Specification (Components)

1. Introduction : It provides the overview of the entire information described in SRS. It should indicate the purpose and scope of SRS and function performed by the system.
2. Functional Requirement : It provides all the requirements related to the functionality of the system.
3. Non-functional Requirement (Design Constraints) : It provides all the requirements related to the performance of the system.

Software Requirement Specification (Components)

4. Standard Compliance : It provides standards that must be met

5. Hardware Limitation

6. Security

7. External Interface Requirement : It provides all the possible interactions of the software with the external entities including people, hardware and other softwares.

Requirement Validation

- Validate whether the requirements meet the standard of practicability, consistency and completeness
- Errors and bugs in the requirements are discovered and corrected in appropriate way

Thank You !