Parul® University

Software Engineering

Dr. Kaushal A Shah, Assistant Professor
Computer Science and Engineering

Parul® University

DIGITAL LEARNING CONTENT

UNIT-3

Requirements Engineering

Parul®
University

DIGITAL LEARNING CONTENT

What is Requirement?

Image source : Google

Parul® University

DIGITAL LEARNING CONTENT

What is a Stakeholder?

Image source : Google

Parul® University

Requirement Practices: Loopholes

- Not understanding the requirements that we take from the customer.
- Record requirements in a unorganized manner
- Not spending enough time verifying the record
- Do not allow mechanisms to control the change
- Lack of importance given to the software that the user wants

Parul® University

Requirement Engineering Tasks

- **Seven distinct tasks**
  1. Inception
  2. Elicitation
  3. Elaboration
  4. Negotiation
  5. Specification
  6. Validation
  7. Requirements Management

Parul® University

DIGITAL LEARNING CONTENT

Inception

- Defines the scope and nature of the problem.
- Context-free questions are asked by software engineer.
- Establish a basic understanding of the problem, the people, nature of the solution and the effectiveness of primary communication and bridge between the customer and the developer.

Parul® University

The First Set of Questions

These questions look on the clients, stakeholders, goals, and the advantages

- **Who is responsible for the request of the work?**
- **Who will utilize the solution?**
- **What will be the financial advantage of a successful solution?**
- **Is there other source for the solution that you require?**

Parul® University

The Next Set of Questions

These questions make the requirements engineer gain a more understanding of the problem and let the customer to say his or her views about a solution

- How would you characterize "good" output that might be generated by a successful solution?
- What problem(s) will this solution address?
- Are you able to show me (or describe) the business environment during which the answer are going to be used?
- Will special performance issues or constraints affect the way the answer is approached?

Parul® University

DIGITAL LEARNING CONTENT

The Final Set of Questions

These questions look on the effectiveness of the communication itself

- Are you the proper person to answer these questions? Are your answers "official"?
- Are my questions relevant to the matter that you simply have?
- Am I asking too many questions?
- Can anyone else provide additional information?
- Should I be asking you anything else?

Parul® University

Elicitation

- **Take the requirements from the customers.**
- **Eliciting requirements is not easy because of:**
    1. Problems of scope in identifying the boundaries of the system or specifying an excessive amount of technical detail instead of overall system objectives
    2. Problems of understanding what's wanted, what the matter domain is, and what the computing environment can handle (Information that's believed to be "obvious" is usually omitted)
    3. Problems of volatility because the wants change over time

Parul® University

Collaborative Requirements Gathering

- Meetings are conducted and attended by both software engineers, customers, and other interested stakeholders
- Rules for preparation and participation are established
- An agenda is usually recommended that's formal enough to hide all details but informal enough to encourage the free flow of ideas
- A "facilitator" (customer, developer, or outsider) controls the meeting
- A "definition mechanism" is employed like work sheets, flip charts, wall stickers, electronic bulletin board, chat room, or another virtual forum
- The goal is to spot the matter , propose elements of the answer , negotiate different approaches.

Parul® University

Quality Function Deployment

- **This is a technique that converts the requirementss of the customer into technical needs for software**
- **It identifies three types of requirements:**

1. Normal requirements: These requirements are the objectives and goals stated for a product or system during meetings with the customer
2. Expected requirements: These requirements are implicit to the merchandise or system and should be so fundamental that the customer doesn't explicitly state them
3. Exciting requirements: These requirements are for features that transcend the customer's expectations and convince be very satisfying when present

Parul® University

Elaboration

- **During elaboration, the software engineer takes the information obtained during inception and elicitation and begins to expand and refine it.**
- **It is an analysis modeling task:**
    1. Use cases are developed
    2. Domain classes are identified along with their attributes and relationships
    3. State machine diagrams are used to capture the life on an object

Parul® University

Use cases

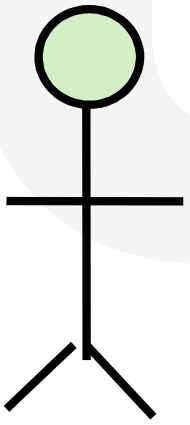
- **Step One – Define the set of actors that will be involved in the story.**
    1. Actors are people, devices, or other systems that use the system or product within the context of the function and behavior that is to be described
    2. Actors are anything that communicate with the system or product and that are external to the system itself
- **Step Two – Develop use cases, where each one answers a set of questions**

Parul® University

DIGITAL LEARNING CONTENT

Use cases - Actors

- **An Actor is outside or external the system. It can be a:**
  1. Human
  2. Peripheral Device (Hardware)
  3. External System or Subsystem
  4. Time or time-based event
- **Represented by the following figure:**



Parul® University

DIGITAL LEARNING CONTENT

Use cases - Relationships

- Represent communication between actor and use case
- Depicted by line or double-headed arrow line
- Also called association relationship

Parul® University

Negotiation

- During negotiation, the programmer reconciles the conflicts between what the customer wants and what are often achieved given limited business resources.
- Requirements are ranked (i.e., prioritized) by the purchasers , users, and other stakeholders
- Risks related to each requirement are identified and analyzed
- Using an iterative approach, requirements are eliminated, combined and/or modified in order that each party achieves some measure of satisfaction

Parul® University

DIGITAL LEARNING CONTENT

The Art of Negotiation

- Recognize that it is not competition
- Map out a strategy
- Listen actively
- Focus on the other party's interests
- Don't let it get personal
- Be creative
- Be ready to commit

Parul® University

Specification

- A specification is that the final work product produced by the wants engineer
- it's normally within the sort of a software requirements specification
- It is the inspiration for subsequent software engineering activities
- It formalizes the informational, functional, and behavioral requirements of the proposed software in both a graphical and textual format
- It describes the function and performance of a computer-based system and therefore the constraints which will govern its development

Parul® University

Validation

- **During validation, the products generated as a result of requirements engineering are examined for quality.**
- **The specification is examined to ensure that**
    1. all software requirements are stated unambiguously
    2. inconsistencies, omissions, and errors are detected and corrected
    3. the work products conform to the standards established for the method , the project, and therefore the product
- **The formal technical review is the basic requirements validation mechanism**

Parul®
University

Requirements Management

- During requirements management, the project team performs a group of activities to spot , control, and track requirements and changes to the wants at any time because the project proceeds
- Each requirement is assigned a singular identifier
- The wants are then placed into one or more traceability tables
- These tables could also be stored during a database that relate features, sources, dependencies, subsystems, and interfaces to the wants
- A requirements traceability table is additionally placed at the top of the software requirements specification

Parul® University

Requirements Engineering Process

- A Requirement Engineering may be a process during which various activities like discovery, analysis and validation of system requirements are done.
- It begins with feasibility study of the system and finishes up with requirement validation.
- This process may be a three stage activity where the activities are arranged within the iterative manner
- within 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.

Parul® University

Software Requirement Specification - SRS

- The software requirements specification document enlists all necessary requirements that are required for the project development.
- To derive the wants we'd like to possess clear and thorough understanding of the products to be developed.
- this is often prepared after detailed communications with the project team and customer.

Parul® University

DIGITAL LEARNING CONTENT

An Example of SRS

Document Title
Author(s)
Affiliation
Address
Date
Document Version

1. Introduction
   1.1. Purpose of this document
   1.2. Scope of this document
   1.3. overview
2. General Description
3. Functional Requirements
   3.1. Description
   3.2. Criticality
   3.3. Technical issues
   3.4. Cost and Schedule
   3.5. Risks
   3.6. Dependencies with other requirements
   3.7. Any other appropriate.
4. Interface Requirements
   4.1. User Interface
   4.1.1. GUI
   4.1.2. CLI
   4.1.3. API
   4.2. Hardware interfaces
   4.3. Communications interfaces
   4.4. Software interfaces.
5. Performance Requirements
6. Design Constraints
7. Other Non-Functionality Attributes
   7.1. Security
   7.2. Binary compatibility
   7.3. Reliability
   7.4. Maintainability
   7.5. Portability
   7.6. Extensibility
   7.7. Reusability
   7.8. Application compatibility
   7.9. Resource utilization
   7.9.1. Serviceability
8. Operational Scenarios
9. Preliminary Schedule
10. Preliminary Budget
11. Appendices

Parul® University

Requirements Validation

- When any model of software is made at that point it's examined for inconsistency, ambiguity , Error etc.
- the wants are often prioritized by the stake holders.
- It grouped with in requirement package which will be implemented by software packages.
- it's one process during which will check about gathered requirements whether it represents an equivalent system or not.
- Here any quite generated errors are often fixed because fixing an requirement errors after delivery may cost up to 100.

Parul® University

DIGITAL LEARNING CONTENT

Requirements Validation (Contd.)

- **Requirement checking can be done in following manner:**

1. **Validity Checks**
2. **Consistency Checks**
3. **Completeness Checks**

Parul® University

Validity Checks

- A user might imagine that a system is required to perform certain functions.
- In validity check analysis may identify additional or different functions that are required.
- Systems have diverse stakeholders with distinct needs, and any set of requirements is inevitably a compromise across the stakeholder community.

Parul® University

Consistency Checks

- **Requirements within the document shouldn't conflict.**
- **There should be no contradictory constraints or descriptions of an equivalent system function.**

Parul® University

Completeness Checks

- **Realism checks-**
    1. Using knowledge of existing technology, the wants should be checked to make sure that they might actually be implemented.
    2. These checks should also appreciate of the budget and schedule for the system development.
- **Verifiability-**
    1. To reduce the potential for dispute between customer and contractor
    2. System requirements should even be written in order that you ought to be ready to write a group of tests which will demonstrate that the delivered system meets each specified requirement.

Parul® University

Requirement Validation Techniques

- **Requirements Reviews-**

1. both the customer and contractor staff should be involved in reviews.
2. Reviews could also be formal (with completed documents) or informal.
3. Good communications should happen between developers, customers and users such a healthy communication helps to resolve problems at an early stage.

- **Prototyping-**

1. The requirements can be examined through executable model of system

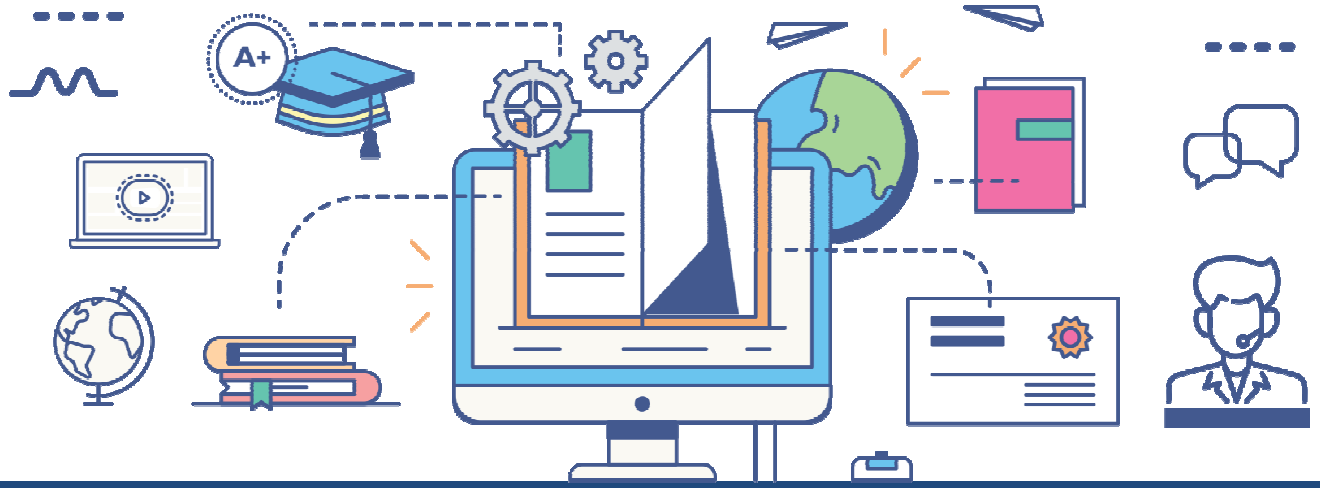
Parul® University

Requirement Validation Techniques (Contd.)

- **Test Case Generation-**

1. If the tests for the wants are devised as a part of the validation process, this often reveals requirements problems.
2. If a test is difficult or impossible to style , this usually means the wants are going to be difficult to implement and will be reconsidered.
3. Developing tests from the user requirements before any code is written is an integral a part of extreme programming.

DIGITAL LEARNING CONTENT



Parul® University

www.paruluniversity.ac.in