Requirements Engineering Summary

Topics Covered

1. Functional and non-functional requirements
2. Requirements engineering processes
3. Requirements elicitation
4. Requirements specification
5. Requirements validation
6. Requirements change

Requirements Engineering

Requirements Engineering (RE) is the process of defining the services a system should provide and the constraints under which it operates and is developed. It involves finding, analyzing, documenting, and checking these services and constraints.

What is a Requirement?

Requirements range from high-level abstract statements to detailed mathematical functional specifications. They can serve as the basis for a contract bid or the contract itself.

Types of Requirements

1. **User Requirements**: Statements in natural language plus diagrams of the services the system provides and its operational constraints, intended for customers.
2. **System Requirements**: A detailed document setting out the system's functions, services, and operational constraints, used as a part of the contract between client and contractor.

Functional and Non-Functional Requirements

- **Functional Requirements**: Describe the services the system should provide, how it should react to inputs, and how it should behave in situations.
- **Non-Functional Requirements**: Constraints on the system such as timing, standards, and quality attributes (e.g., reliability, response time).

Examples of Requirements in the Mentcare System

- **Functional Requirements**: A user shall be able to search the appointment lists for all clinics.
- **Non-Functional Requirements**: The system shall be available to all clinics during working hours, with a downtime not exceeding five seconds per day.

Requirements Engineering Processes

1. **Requirements Elicitation**: Involves working with stakeholders to discover the application domain, services, and constraints.
2. **Requirements Analysis**: Interpreting and structuring gathered information.
3. **Requirements Specification**: Writing down the user and system requirements.
4. **Requirements Validation**: Ensuring the requirements define the system the customer wants.
5. **Requirements Management**: Managing and controlling changes to the requirements.

Requirements Elicitation Techniques

1. **Interviews**: Formal and informal interviews with stakeholders to gather requirements.
2. **Ethnography**: Observing and analyzing how people actually work.
3. **Scenarios and User Stories**: Real-life examples of system usage to help stakeholders relate and comment.

Requirements Specification

The process of documenting the user and system requirements. Specifications can be written in various notations including natural language, structured language, design description languages, graphical notations, and mathematical specifications.

Requirements Validation Techniques

1. **Requirements Reviews**: Systematic manual analysis of requirements.
2. **Prototyping**: Using an executable model to check requirements.
3. **Test-Case Generation**: Developing tests for requirements to check their testability.

Example Scenarios

Scenario 1: Mentcare System Functional Requirement

Requirement: A user shall be able to search the appointments lists for all clinics.

- **Starting Situation**: A user logs into the system.
- **Normal Flow**: The user enters a search query and retrieves the appointment list.
- **What Can Go Wrong**: The system fails to retrieve the list due to a server error.
- **Concurrent Activities**: Other users are also searching the database.
- **End State**: The user views the appointment list or receives an error message.

Scenario 2: Insulin Pump System Non-Functional Requirement

Requirement: The system shall run a self-test routine every minute.

- **Starting Situation**: The system is operating normally.
- **Normal Flow**: The system runs a self-test and confirms all conditions are met.
- **What Can Go Wrong**: The self-test detects a hardware failure.

- **Concurrent Activities**: The system continues to monitor blood sugar levels.
- **End State**: The system alerts the user if an issue is detected, or continues operating normally.

Key Points

- Requirements define what the system should do and its constraints.
- Functional requirements describe system services; non-functional requirements describe system constraints.
- The RE process is iterative, involving elicitation, specification, and validation.
- Good RE practices include clear documentation, stakeholder involvement, and thorough validation.