**Requirements Engineering**

1. **What are the requirements of engineering?**

RE is the process of identifying, documenting, and managing requirements in the engineering design process. It is a typical role in systems engineering and software engineering.

Requirements elicitation: It is the process of acquiring information from stakeholders to understand their needs, wants, and expectations.

Requirements Analysis: After acquiring requirements, they must be examined to ensure they are clear, full, and consistent.

Documentation of requirements: Clear and well-documented requirements are necessary. Use cases, user stories, functional and non-functional requirements, and any other relevant specifications may all be included in the documentation.

Validation of Requirements: Stakeholders review the documented requirements to ensure that they appropriately represent their needs.

Testing, inspection, and other techniques of verification are employed to verify compliance.

Requirements Management: Throughout the project, requirements may vary owing to changing stakeholder needs, external circumstances, or new insights.

Prioritization is necessary since not all criteria are equal in priority. Prioritization assists in determining which criteria are crucial to the project's success and which can be postponed or eliminated.

Communication ensures that all parties understand the project's goals and requirements.

1. **In an agile/iterative software development setting, will you undergo this process once or multiple times?**

In an agile or iterative software development environment, the requirements engineering process is usually repeated numerous times during the project's lifecycle. It is a recurring action that occurs iteratively to guarantee that the project remains aligned with changing stakeholder needs and evolving project goals.

Gathering of Initial Requirements: An initial round of requirements gathering occurs at the start of the project.

Iteration Planning: Before each iteration or sprint, the development team, in consultation with stakeholders and the product owner, selects a subset of needs from the total backlog.

Requirements Refinement occurs during iteration planning or just before.

Development and testing: During the iteration, the determined requirements are implemented, with development and testing activities taking place.

Feedback and Review: Throughout the iteration, there is constant feedback and review of the work in progress.

Retrospectives: At the end of each iteration, the team conducts a retrospective to reflect on what went well and what may be improved in the development process, including how requirements were handled.

Change Management: As new insights emerge, stakeholder priorities shift, or external circumstances impact the project, needs may vary.

Iterations: The project continues to cycle through iterations, each time selecting, implementing, and validating a set of requirements.

Release Planning: The team may engage in release planning on a regular basis, often after multiple iterations, to review overall progress, prioritize remaining work, and prepare for the next software release or increment.

1. **What are the different requirements for engineering activities?  Who takes part in those activities?**

Here are the different requirements engineering activities and the key participants in each:

* Elicitation:
* Participants: Customers, end-users, domain experts, business analysts, product managers, and project managers.
* Activity: Questioning stakeholders to learn about their requirements, expectations, and limitations. Techniques including observations, workshops, questionnaires, and interviews are employed.
* Analysis and Documentation:
* Participants: Business analysts, systems analysts, and subject matter experts.
* Activity: Analyzing and organizing the information gathered into requirements that are precise and short. This entails locating contradictions, disputes, and ambiguities as well as formally recording the requirements.
* Specification:
* Participants: Business analysts, systems analysts, and technical writers.
* Activity: creating requirement documents that are comprehensive, clear, and organized. Textual descriptions, illustrations, use cases, and prototypes can all be found in these documents.
* Validation:
* Participants: Stakeholders, including customers, end-users, and quality assurance teams.
* Activity: Reviewing and validating the requirements to ensure they appropriately represent the stakeholders' needs and expectations. Feedback is gathered to improve the specifications.
* Verification:
* Participants: Developers and quality assurance teams.
* Activity: Ensuring that the implemented system or product conforms to the specified requirements. This is done through testing, inspections, and reviews.
* Management and Traceability:
* Participants: Project managers, configuration managers, and requirements managers.
* Activity: managing requirements modifications, keeping track of their progress, and guaranteeing traceability throughout the development process. For this, a requirements traceability matrix is frequently employed.
* Prioritization:
* Participants: Project managers, product owners, and stakeholders.
* Activity: deciding which requirements should be implemented in the first release because they are the most important. Prioritization aids in resource management and guarantees that the most crucial features are created first.
* Communication:
* Participants: All stakeholders involved in the project.
* Activity: fostering efficient stakeholder communication to notify them on the status of requirements, changes, and their effects on the project.
* Documentation and Maintenance:
* Participants: Technical writers, configuration managers, and project managers.
* Activity: as the project develops, maintaining and updating the requirements papers. It may be required to use version control, and changes and revisions should be thoroughly documented.

1. **Define each of the participants in the above activities (example: developer, stakeholder)**

detailed definition of each participant involved in the various requirements engineering activities:

* Customers: Customers are the people or businesses who will use the finished product or system, according to the definition. They are essential in contributing information about their demands, expectations, and needs from the system.
  + Function: Customers provide the system with high-level goals, business objectives, and user needs. They are the main stakeholders, and the success of the project depends on their satisfaction.
* End-Users: End-users are defined as those who, after the system is installed, will utilize and interact with it directly. They have needs in terms of performance, functionality, and usability.
  + Function: End users provide perceptions into the operational facets of system utilization. User interfaces, user experience (UX) design, and user-specific features are shaped by their input.
* Domain Experts:
  + Definition: People who have specialized knowledge and skill in a particular field, industry, or subject area relevant to the system being developed are known as domain experts.
  + Function: During the requirements engineering process, domain specialists offer helpful insights into industry-specific laws, best practices, and domain-specific requirements.
* Business Analysts:
  + Definition: Business analysts are professionals who specialize in analyzing business needs and translating them into functional and non-functional requirements. They act as intermediaries between stakeholders and development teams.
  + Role: Business analysts facilitate the elicitation, analysis, and documentation of requirements. They ensure that requirements are well-defined, feasible, and aligned with business objectives.
* Product Managers:
  + Definition: Product managers are responsible for defining the overall vision and strategy for a product. They prioritize features and requirements based on business goals and market demands.
  + Role: Product managers guide the requirements prioritization process, ensuring that the most valuable features are developed first and aligning the product with the company's strategic direction.
* Project Managers:
  + According to the definition, project managers are in charge of organizing, carrying out, and overseeing the project. They are in charge of the project's money, resources, and timetable.
  + Function: Project managers oversee responsibilities pertaining to requirements, such as monitoring modifications, upholding timetables, and guaranteeing that requirements are completed on time and within budget.
* Quality Assurance Teams:
  + Teams in charge of quality assurance (QA) are in charge of ensuring that the finished product complies with all requirements and standards for quality. They carry out activities such as testing and validation.
  + Function: The role of QA teams is to make sure that the implemented system conforms with the specifications and runs properly. They assist in locating and disclosing errors or requirements deviations.
* Developers: Developers, also referred to as software engineers or programmers, are in charge of implementing and creating the software or system in accordance with the given specifications.
  + Function: Developers translate specifications into useful software components. They make that the code complies with the design criteria and requirements for the system.
* Configuration Managers:
  + Definition: Throughout the whole development lifecycle, configuration managers oversee controlling the versioning and configuration of requirements documents and software components.
  + Function: By monitoring changes, keeping track of document revisions, and ensuring that requirements are correctly documented and regulated, configuration managers support the integrity and traceability of requirements.

1. **For each of the roles in the above two questions, identify who fills those roles (for example, your team is the developer(s), etc).  Remember that one person/group can play multiple roles!**

* Customers: Sandhya Paladugu
* End-Users: Aparna Marepally
* Domain Experts: Praveen Babu Narni
* Business Analysts: Manoj Vamanaguntla
* Product Managers: Manoj Vamanaguntla
* Project Managers: Praveen Babu Narni
* Quality Assurance Teams: Poojasri Ramineni
* Developers: Aparna Marepally
* Configuration Managers: Chandra Venkata Vijaya Gopal Raju

1. **Define functional requirement.**

A functional requirement is a precise and thorough description of how a system must behave or function to satisfy a certain business need or user demand. A system's design, construction, and testing can all be guided by its functional requirements, which are a crucial component of software and system development processes. These specifications often outline what the system should be able to perform, as well as its features, capabilities, and responses to various inputs and stimuli.

Specificity: There should be no ambiguity in the functional requirements, which should be specific and obvious. They need to be specific when describing the system's behavior.

Measurability: Functional requirements should be testable, which means they can be examined to see whether they have been carried out successfully.

Completeness: They must include all crucial system functionalities, leaving no crucial details unclear.

Consistency: The objectives of the overall project should be aligned with the functional needs, which should not conflict with one another.

Traceability: To be sure that a functional demand is addressing a real need, it should be possible to link it to a specific business need or user requirement.

Prioritization: To meet the needs of the project or the user, functional requirements may be ranked in order of significance. This is beneficial while working on software in iterative cycles or under resource restrictions.

Requirement specifications are frequently used to record functional requirements, which serve as the basis for the processes of system design, development, and testing. They assist in making sure the finished system achieves the desired goals and successfully completes the necessary duties.

1. **Define non-functional requirement.**

The NFR specification describes the characteristics that a system or product need to have rather than what it should be able to achieve. NFRs explain how well the system performs and complies with quality standards, as compared to functional requirements, which describe features.

* Portability
* Security
* Maintainability
* Reliability
* Scalability
* Performance
* Reusability
* Flexibility

1. **What project artifacts (documents, etc) should result from requirements engineering?**

Requirements engineering in software development generates several key documents to ensure everyone understands what the system or product should do. These documents include:

Requirements Document: A central document listing all project requirements.

Use Cases or User Stories: Descriptions of how users interact with the system.

Functional Requirements: Details about specific system features.

Non-Functional Requirements (NFRs): Specifications for system qualities like performance and security.

Traceability Matrix: A map connecting requirements to their sources.

UI/UX Design Documents: Visual design plans if needed.

Prototypes or Mockups: Visual representations of the system.

Data Models: Diagrams showing data structure.

Test Plans and Cases: Documentation for testing system functionality.

Change Control: Procedures for managing requirement changes.

Acceptance Criteria: Criteria for requirement satisfaction.

Risk Assessment: Identification of potential risks.

Dependencies and Constraints: Documentation of external factors.

Quality Assurance Plan: A plan for ensuring system quality.

Requirements Traceability Matrix: A tool to track relationships between project elements.

Compliance Documentation: If needed, documents for regulatory compliance.

These artifacts help guide successful project development and communication with stakeholders.

1. **Define requirements elicitation.**

The process of obtaining and specifying the needs, expectations, and specifications of a system or product that is being developed or modified is known as requirements elicitation. Interacting with stakeholders such as clients, users, subject matter experts, and other relevant parties to extract and document the project's core features, functions, limitations, and objectives is a critical phase in the software development or project management process. The purpose of requirements elicitation is to ensure a clear and thorough understanding of what the system or product must do, which acts as the foundation for the succeeding design, development, and testing phases. During this process, many techniques such as interviews, surveys, seminars, observation, and prototype demonstrations may be used to collect and define requirements.

1. **What is the sequence of steps one should take during requirements elicitation?  Which step is potentially the hardest?**

Requirements Elicitation is a multi-step process. While the particular procedures may differ based on the technique and circumstance.

Identify Stakeholders: Determine all important stakeholders with an interest or position in the project.

Plan Elicitation Activities: Create an elicitation plan including the goals, scope, schedule, and resources for the requirements elicitation process.

Conduct background research: Gather existing paperwork, reports, and any relevant information about the project or system to understand its context and history.

Elicitation Techniques to Consider: Select the right techniques for gathering requirements.

Conduct Elicitation Sessions: Engage with stakeholders using specific strategies to acquire information about their requirements, goals, and expectations.

record Requirements: Carefully record the requirements acquired, ensuring they are clear, complete, and well-structured.

Validate Requirements: Review and confirm the requirements with stakeholders to ensure accuracy and completeness.

Prioritize requirements: Prioritize needs depending on their importance and impact on the project's aims.

Manage Changes: Be prepared for changes in needs as the project continues. Establish a change management process to effectively review and manage any suggested modifications.

Obtain Sign-off: Obtain formal approval from stakeholders, usually through requirements review or sign-off process, to certify their agreement of the specified requirements.

Maintain Traceability: To guarantee alignment and consistency across the project lifecycle, establish traceability between requirements and other project artifacts such as design documents and test cases.

The most difficult step may differ based on the project and the stakeholders.

Understanding the Needs of Stakeholders

Management of Scope

Requirements Management Changes

Requirements Documentation and Communication

Keeping Traceability

1. **What are key things you should try to identify when working with the client?  What kinds of questions should you ask?**

When working with a client during requirements elicitation, it's important to identify key information that will help us to understand their needs, objectives, constraints, and expectations. Here are some key things to try to identify and questions to ask:

Business Goals and Objectives:

* What are the overarching goals of your organization or project?
* What specific objectives are you trying to achieve through this project or system?

Stakeholders:

* Who are the primary stakeholders and decision-makers for this project?
* Are there any secondary or less obvious stakeholders who should be considered?

User Requirements:

* Who are the end-users of the system or product?
* What are their specific needs, preferences, and pain points?
* How will the system improve their work or lives?

Functional Requirements:

* What specific functionalities or features do you envision for the system?
* Are there any must-have features or nice-to-have features?

Non-Functional Requirements:

* What performance expectations do you have (e.g., response time, scalability)?
* What security and compliance requirements need to be met?
* Are there any usability or accessibility standards that must be followed?

Budget and Resource Constraints:

* What is the budget allocated for this project?
* Are there any resource limitations, such as available staff or technology?

Timeline and Milestones:

* What is the desired project timeline or delivery schedule?
* Are there any critical milestones or deadlines that need to be met?

Integration with Existing Systems:

* Does the new system need to integrate with existing software or systems?
* Are there any data migration or compatibility requirements?

Scalability and Future Growth:

* Do you anticipate the need for scalability as your organization or user base grows?
* How do you envision the system evolving in the future?

Regulatory and Compliance Requirements:

* Are there any industry-specific regulations or compliance standards that must be adhered to?
* Do you have data privacy or security requirements to consider?

User Experience and Design Preferences:

* Do you have any design preferences or brand guidelines that should be followed?
* What user experience (UX) considerations are important to you?

Testing and Quality Assurance:

* What are your expectations regarding testing, quality assurance, and bug fixing?
* How will you measure the success and quality of the final product?

Change Management and Training:

* How do you plan to manage the transition to the new system or product?
* Will end-users require training, and if so, what kind?

Risk Factors:

* Are there any potential risks or challenges that you foresee in this project?
* How do you plan to mitigate or address these risks?

Success Criteria:

* How will you measure the success of the project once it's completed?
* What key performance indicators (KPIs) will be used to evaluate its impact?

Feedback and Communication:

* What is the preferred method and frequency of communication throughout the project?
* How open are you to receiving feedback and adjusting during development?

These questions help us understand what the client needs and wants. This way, we can create clear requirements that match their goals. It's important to keep talking with the client during the project to make sure the requirements still work for them as things change.

**Citation**:

<https://chat.openai.com/> (Content from chatgpt)

[Requirements engineering - Wikipedia](https://en.wikipedia.org/wiki/Requirements_engineering)

[Functional vs Non Functional Requirements - GeeksforGeeks](https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/)