**Requirements Engineering**

**Team-03**

1.What is requirements engineering?

A: Requirements engineering involves defining the functionalities desired by the customer from a system, as well as the limitations and conditions within which it is both developed and operates.[1]

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

A: We will undergo this process multiple times in an agile/iterative software development setting. It is an iterative process where RE activities are integrated with other software engineering tasks.[1]

3. What are the different requirements engineering activities?  Who takes part in those activities?

A: Requirements Engineering Activities are:

1.Elicitation

2.Requirement Specification or Modeling

3.Requirement Analysis

4.Validation and Verification

5.Requirements Management

People involved in those activities are Stakeholders, Software developers, testers, Product owner, Project Manager, Customer and Business Analysts.[2]

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

A: Stakeholders: People who are involved in project or affected by project.

Product Owner: They prioritize the features and functionalities of the project.

Project Manager: They plan and organize the activities of the development team, ensure that the project objectives are met.

Developers: They are responsible for writing code, build or develop functionalities for project and ensure they work properly.

Testers: They identify defects or bugs in the software and ensure the proper functionality of project through tests.

Business Analyst: Responsible for communication between stakeholders and the development team. They gather and analyze requirements from product owner.

5. 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!

A: Client or Product Owner: Professor Mark Chai

|  |  |
| --- | --- |
| People | Role |
| Divya Bathala | Business Analyst, Stakeholder |
| Lakshmi Manasa Gundala | Backend-end Developer, Stakeholder |
| Parthasarathy Boda | Tester, Stakeholder |
| Srilatha Yadala | Project Manager, Stakeholder |
| Vani Battu | Front-end Developer, Stakeholder |
|  |  |

6. Define functional requirement?

A: Functional requirements define what a system should do. They specify the functions or activities that the system must be able to perform. These requirements are often described in detail, outlining specific actions or tasks that the system needs to accomplish to meet the needs of its users. Functional requirements are crucial for guiding the design, development, and testing of a project to ensure that it meets the desired functionality and user expectations.[2]

7. Define Non-functional requirement?

A: Non-functional requirements are like the quality rules for a system. They don't tell the system what to do, but how well it should do it. They focus on things like how fast it should be, how reliable, easy to use, able to handle lots of users, how secure, and how easy to maintain. While functional requirements say "do this task," non-functional requirements say "do this task well." They're super important for making sure the system not only works but works really well and meets users' needs effectively.

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

A: Project artifacts result from requirements engineering are:

* **Requirements Document**: This document outlines all the functional and non-functional requirements of the project. It includes details like what the system should do, how it should perform, and any constraints it must adhere to. It's like a blueprint for the project.
* **Use Case Diagrams**: These diagrams show interactions between users and the system, depicting various scenarios of how users will interact with the system to achieve their goals.
* **User Scenarios:** These are short descriptions of what users want from the system. They're written from the user's point of view, explaining what they need and why.
* **Prototypes:** These are like sketches or early versions of the system. They give everyone a visual idea of how the final product will look and function.
* **Acceptance Criteria**: These are the conditions the system must meet to be approved by the people who need it. They make sure the final product meets everyone's expectations.

9. Define requirements elicitation?

A: Requirements elicitation is the process of collecting, analyzing, and documenting requirements from stakeholders to understand what a project or system needs to achieve. This includes identifying stakeholders, determining their needs and expectations, and translating them into clear and concise requirements that can guide the development process. Interviews, workshops, surveys, observations, and prototyping are common techniques used to elicit requirements. The goal is to make sure that the product meets the needs of its users and stakeholders.

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

A: Requirements elicitation is a crucial stage in software development where the needs and expectations of stakeholders are gathered and analyzed. Following are the steps:

1. **Identify Stakeholders**: Determine who the key stakeholders are. These could include end-users, customers, project managers, developers, and any other people involved in the project.
2. **Gather Primary Information**: Collect existing documentation, such as user manuals, business process documents, and any other relevant materials that can provide details of the project requirements.
3. **Eliciting Requirements:** This phase is generally done with the help of stake holders using different elicitation techniques to meet the project requirements. Some of the elicitation techniques are Brain Storming, focus groups, interviews, prototyping, conducting workshops, surveys etc.
4. **Documenting Requirements**: There are different tools like specialized software or simple spreadsheets, depending on the project's complexity and stakeholder involvement.
5. **Confirming Findings**: After documenting requirements, they're shared with stakeholders for review and feedback to ensure accuracy. Stakeholders might suggest refinements or new requirements during this process. This ongoing review helps adapt to changes and ensures everyone's on the same page.

The hardest step in requirements elicitation is eliciting the requirements as there will be multiple stakeholders for the project there are many views, requirements, and opinions as well. Analyzing all these and deciding requirements is crucial and probably the hard part.

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

A: When working with a client, it's important to identify the key information to ensure a successful project. Here are some key things to try to identify and questions to ask:

Project Goals and Objectives: What are the client's main goals and objectives for the project? What outcomes are they expecting to achieve?

Scope: What is the scope of the project? What features or functionalities are essential, and what can be considered optional?

Constraints: Are there any time, budget, or resource constraints that need to be considered? Are there any technical or regulatory constraints?

Communication and Collaboration: What are the preferred communication channels and frequency for updates and feedback? Who are the key stakeholders involved in the project?

Previous Experience: Has the client worked on similar projects before? What were the challenges and successes they experienced?

Risk Management: What potential risks or challenges do they foresee in the project? How do they propose to mitigate these risks?

Timeline and Milestones: What is the desired timeline for project completion? Are there any specific milestones that need to be achieved along the way?

**Citation References:**

1. “The Influence of Human Aspects on Requirements Engineering-related Activities: Software Practitioners’ Perspective” by Dulaji Hidellaarachchi , John Grundy , Rashina Hoda , Ingo Mueller. [<https://dl.acm.org/doi/10.1145/3546943>]
2. “Research Directions in Requirements Engineering” by Betty H.C. Cheng and Jo Atlee[<https://www.researchgate.net/publication/4250874_Research_Directions_in_Requirements_Engineering>]