**Investigation: Requirements Engineering : Group01**

1. **What is requirements engineering?**
2. Gathering, recording, and organizing stakeholder needs and expectations for a software system in a methodical manner is called requirements engineering. It entails being aware of restrictions, corporate goals, and user needs. This procedure guarantees that the software is designed with the intended usage and benefits the users.

Activities related to requirements engineering involve gathering requirements from relevant parties, evaluating, and ranking them, clearly and understandably documenting them, and verifying that they satisfy the needs of the parties involved. It is an iterative procedure that is frequently reviewed to account for modifications and guarantee compatibility with changing project objectives over the course of the software development lifecycle. Successful requirements engineering necessitates cooperation between stakeholders, including users, developers, and project managers.[1]

1. **In an agile/iterative software development setting, will you undergo this process once or multiple times?**
2. In agile/iterative software development setting we undergo this process through multiple times. As a result, requirements are updated and modified iteratively in agile environments as the program develops and new knowledge is obtained from user input and continuous development. Throughout the development lifespan, the software is guaranteed to match the evolving needs of stakeholders thanks to this iterative approach, which enables ongoing improvement. [3]
3. **What are the different requirements engineering activities? Who takes part in those activities.**
4. The goal of requirements engineering is to collect, record, analyze, and validate the needs for a system or product through several tasks. The main activities are as follows:
5. **Elicitation**: We talk to everyone to understand what's needed.
6. **Documentation**: We write down all that we've learned in a clear document.
7. **Analysis**: We check if everything makes sense and there are no mistakes.
8. **Validation**: We make sure everyone agrees with what we've written.
9. **Verification**: We check if the project matches what we wrote while building it.
10. Management: We keep track of any changes along the way.
11. Communication: We keep talking to each other to make sure we understand everything correctly.

The people who involved in these activities are: Business analysts, Developers, Product owners, project managers, Subject matter experts and people who are involved in the project. **[2]**

1. **Define each of the participants in the above activities (example: developer, stakeholder).**
2. Business Analysts: Understand business needs and translate them into technical plans.
3. Product Owners: Represent stakeholders' interests and prioritize project requirements.
4. Subject Matter Experts (SMEs): Provide domain-specific knowledge and insights for project clarity.
5. Developers: Implement technical solutions based on project requirements.
6. Designers: Create user-friendly and visually appealing designs aligned with project goals.
7. Testers / QA Engineers: Ensure the project meets quality standards and functions correctly.
8. Project Managers: Coordinate and oversee project activities to ensure timely delivery and success.
9. End-users / Customers: Provide feedback and ensure the project meets their needs and expectations. **[3]**
10. **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! [4]**

**Developers**: Harichaithanya Kotapati, Laxminaryana Yadav

**Stakeholders**: Laxminaryana Yadav

**Project Manager**: Teja Kumar Muppala

**QA**: Mani Chandra, Vaishnavi Inturi.

**Business Analyst**: Vaishnavi Inturi.

1. **Define functional requirement.**
2. A functional requirement is a concise statement of the features that a system or product must have. Users' or stakeholders' requests are met by defining specific behaviors, actions, or functions that the system must perform. **[2]**
3. **Define non-functional requirement.**
4. Non-functional requirements describe how the system should behave or perform, focusing on qualities like speed, reliability, and security rather than specific tasks. They're like the "how" of the system, ensuring it works smoothly and effectively. **[3]**

1. **What project artifacts (documents, etc) should result from requirements engineering?**
2. The project artifacts the results form requirements engineering are as follows:
3. **Requirement Specifications**: These are detailed documents that list what the project needs to do, including both functional things and non-functional things. **[**1]
4. **User Stories**: These are like little tales that explain how users will interact with the system, demonstrating their actions and the system's reactions.
5. **Requirements Traceability Matrix (RTM):** To ensure that nothing is overlooked or lost, this elegant table links each necessity to its origin and destination.[3]
6. **System or Software Requirements Document (SRD):** This comprehensive document serves as a development and testing guide and contains all the specifications for the system, including what it should look and feel like.
7. **Prototypes:** They let everyone visualize the product and identify any problems early on. These are similar to sketches or models of how the system will look and function.
8. **Change Control Documentation:** This ensures that everyone is in agreement by recording any modifications to the requirements along the way, along with the reason for the change and the person who approved it.
9. **Requirement Validation Reports:** These reports help address issues before they become critical by indicating whether the requirements are good or if there are flaws, based on things like reviews or tests.
10. **Requirement Management Plan:** This plan outlines who is in charge of what during the project, how requirements will be managed, and how communication between team members will occur. **[**1]
11. **Define requirements elicitation.**
12. The process of "requirements elicitation" is like piecing together a mystery. It's the process of getting to know individuals and finding out what they need or desire from a project through conversation and questioning. We gather information from various sources to determine what the project should accomplish, just way detectives gather evidence to comprehend a case. **[4]**
13. **What is the sequence of steps one should take during requirements elicitation? Which step is potentially the hardest?**
14. The steps involved during requirements elicitation are as follows:
15. **Preparation**: Prepare by learning the objectives of the project, identifying the key players, and selecting the most effective elicitation methods. **[4]**
16. **Stakeholder Identification**: Figure out who all the people are who have a stake in the project or who will be affected by it.
17. **Information Gathering**: Engage stakeholders through questionnaires, workshops, or interviews to learn about their requirements, goals, and limitations.
18. **Documentation**: Write down all the information collected in a clear and organized way so that everyone can understand it.
19. **Analysis**: Examine and evaluate the data acquired in order to spot recurring themes, inconsistencies, or gaps.
20. **Validation**: Identify any gaps or reoccurring patterns by carefully reviewing and assessing the collected data.
21. **Prioritization**: Decide which requirements are most important and need to be addressed first. **[4]**

From the above steps **Analysis** is the hardest step This is because it requires analyzing and making sense of the data that has been acquired, which can occasionally be confusing and complex. It can be difficult to recognize patterns and reconcile contradictory requirements; doing so may call for a thorough comprehension of the needs of the stakeholders as well as the project area.[3]

1. **What are key things you should try to identify when working with the client? What kinds of questions should you ask?**
2. When working with a client, it's important to identify key information to understand their needs and expectations. Some of the key things are as follows:
3. **Goals and Objectives**: What are the client's main goals for the project? What do they hope to achieve?
4. **Scope**: What are the boundaries of the project? What should be included and what should be excluded?
5. **Requirements**: What specific features or functionalities does the client want in the final product?
6. **Constraints**: Are there any limitations or restrictions that need to be considered, such as budget, timeline, or technical constraints?
7. **Stakeholders**: Who are the key people involved in the project, and what roles do they play?
8. **Expectations**: In terms of deliverables, updates, and communication, what does the client anticipate? **[4]**

The questions we ask to the client are:

1. What goals do you have for this project?
2. Which features are essential for the finished product?
3. Are there any dates or financial restrictions that we should be mindful of?
4. Which project priorities are most important to you?
5. How frequently would you like to be informed on the status of the project? [3]

**References:**

1. IEEE Computer Society. (1998). IEEE Recommended Practice for Software Requirements Specifications. IEEE Std 830-1998. <http://www.cse.uaa.alaska.edu/~afkjm/csce401/IEEE830.pdf>

2. International Institute of Business Analysis (IIBA). (n.d.). Guide to the Business Analysis Body of Knowledge (BABOK Guide). Retrieved from <https://www.iiba.org/>

3. Software Engineering Institute (SEI). (n.d.). Requirements Engineering Framework. Retrieved from <https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=534004>

4. International Requirements Engineering Board (IREB). (n.d.). Certified Professional for Requirements Engineering (CPRE). Retrieved from <https://www.ireb.org/en/>