REQUIREMENTS ELICITATION

Requirements elicitation is the process of gathering and defining the requirements for a software system. The goal of requirements elicitation is to ensure that the software development process is based on a clear and comprehensive understanding of the customer's needs and requirements. Requirements elicitation involves the identification, collection, analysis, and refinement of the requirements for a software system. It is a critical part of the software development life cycle and is typically performed at the beginning of the project. Requirements elicitation involves stakeholders from different areas of the organization, including business owners, end-users, and technical experts. The output of the requirements elicitation process is a set of clear, concise, and well-defined requirements that serve as the basis for the design and development of the software system.

Requirements elicitation is perhaps the most difficult, most error-prone, and most communication-intensive software development. It can be successful only through an effective customer-developer partnership. It is needed to know what the users really need.

Requirements elicitation Activities: Requirements elicitation includes the subsequent activities. Few of them are listed below –

- Knowledge of the overall area where the systems is applied.
- The details of the precise customer problem where the system is going to be applied must be understood.
- Interaction of system with external requirements.
- Detailed investigation of user needs.
- Define the constraints for system development.

Requirements elicitation Methods:

There are a number of requirements elicitation methods. Few of them are listed below –

1. Interviews
2. Brainstorming Sessions
3. Facilitated Application Specification Technique (FAST)
4. Quality Function Deployment (QFD)
5. Use Case Approach

The success of an elicitation technique used depends on the maturity of the analyst, developers, users, and the customer involved.

1. Interviews:

Objective of conducting an interview is to understand the customer's expectations from the software.
It is impossible to interview every stakeholder hence representatives from groups are selected based on their expertise and credibility.

Interviews maybe be open-ended or structured.

1. In open-ended interviews there is no pre-set agenda. Context free questions may be asked to understand the problem.
2. In structured interview, agenda of fairly open questions is prepared. Sometimes a proper questionnaire is designed for the interview.

2. Brainstorming Sessions:

- It is a group technique
- It is intended to generate lots of new ideas hence providing a platform to share views
- A highly trained facilitator is required to handle group bias and group conflicts.
- Every idea is documented so that everyone can see it.
- Finally, a document is prepared which consists of the list of requirements and their priority if possible.

3. Facilitated Application Specification Technique:

Its objective is to bridge the expectation gap – the difference between what the developers think they are supposed to build and what customers think they are going to get. A team-oriented approach is developed for requirements gathering. Each attendee is asked to make a list of objects that are-

1. Part of the environment that surrounds the system
2. Produced by the system
3. Used by the system

Each participant prepares his/her list, different lists are then combined, redundant entries are eliminated, team is divided into smaller sub-teams to develop mini-specifications and finally a draft of specifications is written down using all the inputs from the meeting.

4. Quality Function Deployment:

In this technique customer satisfaction is of prime concern, hence it emphasizes on the requirements which are valuable to the customer.
3 types of requirements are identified –

- **Normal requirements –** In this the objective and goals of the proposed software are discussed with the customer. Example – normal requirements for a result management system may be entry of marks, calculation of results, etc
- **Expected requirements –** These requirements are so obvious that the customer need not explicitly state them. Example – protection from unauthorized access.
- **Exciting requirements –** It includes features that are beyond customer's expectations and prove to be very satisfying when present. Example – when unauthorized access is detected, it should backup and shutdown all processes.

The major steps involved in this procedure are –

1. Identify all the stakeholders, eg. Users, developers, customers etc
2. List out all requirements from customer.
3. A value indicating degree of importance is assigned to each requirement.
4. In the end the final list of requirements is categorized as –
   - It is possible to achieve
   - It should be deferred and the reason for it
   - It is impossible to achieve and should be dropped off

5. Use Case Approach: This technique combines text and pictures to provide a better understanding of the requirements. The use cases describe the 'what', of a system and not 'how'. Hence, they only give a functional view of the system. The components of the use case design includes three major things – Actor, Use cases, use case diagram.

1. **Actor** –
   It is the external agent that lies outside the system but interacts with it in some way. An actor maybe a person, machine etc. It is represented as a stick figure. Actors can be primary actors or secondary actors.
   - Primary actors – It requires assistance from the system to achieve a goal.
   - Secondary actor – It is an actor from which the system needs assistance.
2. **Use cases –** They describe the sequence of interactions between actors and the system. They capture who(actors) do

what(interaction) with the system. A complete set of use cases specifies all possible ways to use the system.

Use case diagram – use case diagram graphically represents what happens when an actor interacts with a system. It captures the functional aspect of the system.

- A stick figure is used to represent an actor.
- An oval is used to represent a use case.
- A line is used to represent a relationship between an actor and a use case.

Features of requirements elicitation:

1. **Stakeholder engagement:** Requirements elicitation involves engaging with stakeholders such as customers, end-users, project sponsors, and subject matter experts to understand their needs and requirements.
2. **Gathering information:** Requirements elicitation involves gathering information about the system to be developed, the business processes it will support, and the end-users who will be using it.
3. **Requirement prioritization:** Requirements elicitation involves prioritizing requirements based on their importance to the project's success.
4. **Requirements documentation:** Requirements elicitation involves documenting the requirements in a clear and concise manner so that they can be easily understood and communicated to the development team.
5. **Validation and verification:** Requirements elicitation involves validating and verifying the requirements with the stakeholders to ensure that they accurately represent their needs and requirements.
6. **Iterative process:** Requirements elicitation is an iterative process that involves continuously refining and updating the requirements based on feedback from stakeholders.
7. **Communication and collaboration:** Requirements elicitation involves effective communication and collaboration with stakeholders, project team members, and other relevant parties to ensure that the requirements are clearly understood and implemented.

8. **Flexibility:** Requirements elicitation requires flexibility to adapt to changing requirements, stakeholder needs, and project constraints.

Advantages of Requirements Elicitation:

- Helps to clarify and refine customer requirements.
- Improves communication and collaboration between stakeholders.
- Increases the chances of developing a software system that meets customer needs.
- Avoids misunderstandings and helps to manage expectations.
- Supports the identification of potential risks and problems early in the development cycle.
- Facilitates the development of a comprehensive and accurate project plan.
- Increases user and stakeholder confidence in the software development process.
- Supports the identification of new business opportunities and revenue streams.

Disadvantages of Requirements Elicitation:

- Can be time-consuming and expensive.
- Requires specialized skills and expertise.
- May be impacted by changing business needs and requirements.
- Can be impacted by political and organizational factors.
- Can result in a lack of buy-in and commitment from stakeholders.
- Can be impacted by conflicting priorities and competing interests.
- May result in incomplete or inaccurate requirements if not properly managed.
- Can lead to increased development costs and decreased efficiency if requirements are not well-defined.