



Module 5

Understanding Requirements and Requirements Modeling

▼ What tasks are involved in requirements engineering?

- Inception: Starting the project. This task is to identify the business need. The team also establishes a basic understanding of the problem, the people who want a solution, and the nature of the solution that is desired.
- Elicitation: Determining the business goal. The team asks the customer, the users, and others what the objectives for the system or product are, what is to be accomplished, how the system or product fits into the needs of the business, and finally, how the system or product is to be used on a day-to-day basis.
- Elaboration: Developing a refined requirements model. User scenarios gathered during elicitation are refined to determine how end users will interact with the system.
- Negotiation: Reconciling potential conflicts. It is normal for customers and users to ask for more than can be achieved with the business resources available. It is not uncommon for different customers or users to propose conflicting requirements, arguing that their version is essential for their special needs. Negotiation addresses all of these issues.
- Specification: Completing a written document, a series of graphical models, a mathematical model, a set of usage scenarios, a prototype, or some combination of these. The format varies with the size and complexity of the project.
- Validation: Examining the specification for quality, making sure that that all software requirements are complete and clearly stated, that any errors have

been corrected, and that the work products meet the standards established for the project. Consistency is an important element of this task.

- Requirements management: Defining a set of activities that help the project team identify, control, and track requirements and changes to requirements at any time as the project proceeds.

▼ What is the goal of requirements gathering?

to identify the problem, propose elements of the solution, negotiate different approaches, and specify a preliminary set of solution requirements

▼ What are use cases or scenarios?

- they provide an understanding of how the software will be used by different classes of end users
- these scenarios need to be completed before moving on to the technical aspect of the project

▼ The final step in requirements gathering involves the elicitation of work products.

What may they include?

- A statement of need and feasibility
- A statement of system or product scope
- A list of customers, users, and other stakeholder who participated in the requirements gathering
- A description of the technical environment of the system
- A list of requirements (ideally organized by function) and domain constraints for each
- A set of usage scenarios to provide an understanding of how the system or product will be used under different operating conditions

▼ What are actors in a use case?

- An actor is anything or anyone that communicates with the system or product that is external to the system or product.

- In other words, actors may represent different roles played by users or by external hardware as they interact with the system. By defining these actors, you are able to rehearse expected interactions.

▼ What questions should be answered by a use case?

1. Who is the primary actor, the secondary actor(s)?
2. What are the actor's goals?
3. What preconditions should exist before the story begins?
4. What main tasks or functions are performed by the actor?
5. What exceptions might be considered as the story is described?
6. What variations in the actor's interaction are possible?
7. What system information will the actor acquire, produce, or change?
8. Will the actor have to inform the system about changes in the external environment?
9. What information does the actor desire from the system?
10. Does the actor wish to be informed about unexpected changes?

▼ What is an analysis model?

An analysis model provides a description of the required informational, functional, and behavioral domains for a computer-based system.

▼ Why is the software modeling process necessary?

to get a complete picture of what the final software product will look like

▼ What are some kinds of requirements models?

- Scenario-based models of requirements from the point of view of various system "actors"
- Class-oriented models that represent object-oriented classes (attributes and operations) and how classes collaborate to achieve system requirements
- Behavioral models that depict how the software reacts to internal or external "events"
- Data models that illustrate the information domain for the problem

- Flow-oriented models that represent the functional elements of the system and how they transform data as they move through the system

▼ What is the UML (unified modeling language) class model?

- Once the development process gets to the implementation phase, classes and their associated objects are designed to represent a system.
- Then, relevant operations are defined. For example, during the first step of deriving a set of operations for an analysis class, you can study a use case and select those operations that reasonably belong to the relevant class.

A UML class model for a floor plan system:

