# **Use Case Diagram**

A Use Case Diagram is a visual representation of a system's functionality, depicting how different users (actors) interact with the system to achieve specific goals. It is a fundamental part of Unified Modeling Language (UML) and is commonly used in software engineering to model system requirements. The purpose of a Use Case Diagram is to provide a high-level understanding of the system's behavior, making it easier for stakeholders, such as developers, business analysts, and customers, to understand how the system functions.

**Components of a Use Case Diagram**

A Use Case Diagram consists of several key components:

1. Actors: Actors represent external entities (users, systems, or organizations) that interact with the system. There are two types:

* Primary Actors: Directly use the system to achieve their goals.
* Secondary Actors: Support the system but do not directly initiate actions.

1. Use Cases: These are specific functionalities or actions that an actor can perform within the system. Each use case represents a distinct system behavior.
2. Relationships: Different types of relationships exist between use cases and actors, including:

* Association: A direct link between an actor and a use case.
* Include: Represents a mandatory dependency where a use case includes another use case.
* Extend: Represents an optional behavior where a use case extends another use case under specific conditions.
* Generalization: Shows inheritance where one actor or use case is a specialized version of another.

## **Importance of Use Case Diagrams**

Use Case Diagrams play a crucial role in system design and analysis by helping:

* Define system boundaries: Clearly identifying what is part of the system and what is external.
* Enhance communication: Providing a shared understanding for both technical and non-technical stakeholders.
* Improve system documentation: Serving as a reference for developers, testers, and analysts.
* Identify missing requirements: Ensuring all functional needs of the system are captured before development begins.