

## Practical - 03.

Subject - Object oriented Modeling and design. (OOPD)

Assignment title: Use case diagram for capturing and representing system requirements.

Semester - VII., BE.

Instructor: Prof. S. R. Bhise.

Submission date: 04/08/2025.

### 1) OBJECTIVE:

- To understand and apply industry-standard UML guidelines for modeling system requirements using use case diagrams.
- To develop professional-quality use case templates documenting detailed scenarios with basic and alternative flows.
- To improve communication of requirements using among technical and non-technical stakeholders through clear standardized artefacts.

### 2) PROBLEM STATEMENT:

Draw one or more use case diagrams to capture and represent requirements of a chosen system. Complement these diagrams with detailed use case templates showing description and step wise flows for various real world user scenarios.

Note: "Smart Home Automation System".

### 3) INTRODUCTION TO USE CASE MODELING.

Use case diagrams are a core UML artefact officially standardized by the Object Management Group (OMG). They define the system boundary by illustrating the interactions between external actors (users or other systems) and system functionalities. These diagrams help clarify requirements, establish scope and serve as a communication bridge among diverse stakeholders including business analysts, developers, and end users.

In professional settings, use case models are part of iterative development cycles and are routinely defined based on stakeholder feedback to ensure alignment with business goals and user needs.

### 4) THEORY AND BEST PRACTICES.

#### ◦ UML Elements:

- **Actors:** External entities interacting with the system (people, other systems, devices)
- **Use Cases:** Specific system functions providing value to actors.
- **Relationships:** Include (reuse common behaviour), Extend (optional/conditional behaviours), Generalization (inheritance among actors or use cases)

#### ◦ Naming and Notation -

- Use case names should start with strong action verbs reflecting clear domain context (eg. "place order" vs. "order")
- Actors are named by user roles or external system names.



- Use standard UML symbols - oval for use cases, stick figures for actors, rectangle for system boundary.

### ◦ Diagram Layout:

- Organize primary use cases on the left/top, extended or supporting use cases on the right/bottom.
- Clearly label actors outside the system boundary for clarity.

### ◦ Use case template Contents:

- Use case name.
- Actors involved.
- Description/purpose.
- Preconditions (system state before execution)
- Basic flow: (detailed main success scenario steps)
- Alternative / Exception flows (variations, error handling)
- Post conditions (system state after execution)
- Business rules or constraints (if applicable)
- Assumptions or Notes.

## 5] ASSIGNMENT WORKFLOW -

### Step 1: System definition and Boundary.

Clearly define system scope, indicating what functionalities and actors are inside versus ~~outside~~ the boundary.

Step 2: Identify Actors - list all external entities interacting with the system. Categorize them by roles or system names.

Step 3: Identify and list use cases. - derive use cases by identifying user goals and core system functions. Prioritize critical functionalities.

Step 4: Draw use case diagrams -

Create one or more use case diagrams with UML compliant symbols and proper labelling.

Step 5: Develop use case templates: select at least two major use cases and document detailed templates including basic and alternative flows.

Step 6: Validate with stakeholders (Hypothetical) - Include notes on feedback or validation approaches that ensure the model aligns with user needs.

## 6] DELIVERABLES :-

- Introduction and overview of use case modelling.
- Neatly drawn, labelled use case diagram.
- Completed use case template documentation.
- Optional stakeholder validation notes (if feasible)

## 7] Evaluation Criteria -

- Correctness and completeness of UML notation
- Clarity, consistency and professionalism in naming and diagram layout.
- Depth and comprehensiveness including flows, business rules.
- Overall presentation and timely submission.

## 8] RECOMMENDED TOOLS :-



- UML tools recognized in industry such as Draw.io, Microsoft Visio, Lucidchart, Astah or open source UML tools.
- Collaborative platforms like Miro, Confluence for review and iterative refining (Optional)

## 9] Systems for modelling -

### ◦ Smart home Automation System:

- A smart home automation system is an integrated platform that allows residents to monitor, control and automate various home appliances and systems (like lighting, climate control, security cameras, smart locks, entertainment systems, etc) through a centralized interface such as a smartphone application.
- The system integrates IoT devices, sensors, actuators to provide real time monitoring, energy efficiency, security and convenience.
- Users can schedule tasks, receive alerts and remotely operate devices, ensuring seamless interaction between humans and smart devices within defined home boundary.

## 10] Requirements for the assignment -

- System Boundary - Clearly defined and Confined to a manageable scope.
- Actors - Comprehensive identification and categorization based on system interaction.

- Use cases: Capturing major business user goals linking to system functionality.
- UML diagrams - Use case diagrams must be standard, precise and clearly labelled.
- Templates - Detailed use case description with flows, pre-post conditions, business rules and exceptions.
- Stakeholder validation - Include sample feedback.
- Documentation - Clear, consistent, professionally presented.

### 11] Sample feedback from stake holder -

- The system should have a clear onboarding process for device installation and setup. From a maintenance perspective, a diagnostic dashboard for identifying faulty devices and connection issues would be very useful.
- Security protocols need to be robust. Ensure multifactor authentication for remote access.
- Prefer a diagnostic and alert systems for proactive maintenance and issue resolution.

### 12] CONCLUSION:

Use case modelling is a pivotal activity bridging requirements gathering and system design. Creating standardized diagrams alongside detailed scenario templates ensures a comprehensive understanding of system functionality and user interactions, facilitating seamless communication among all project stakeholders. This assignment equips students with practical skills essential for real world software development and professional collaboration.