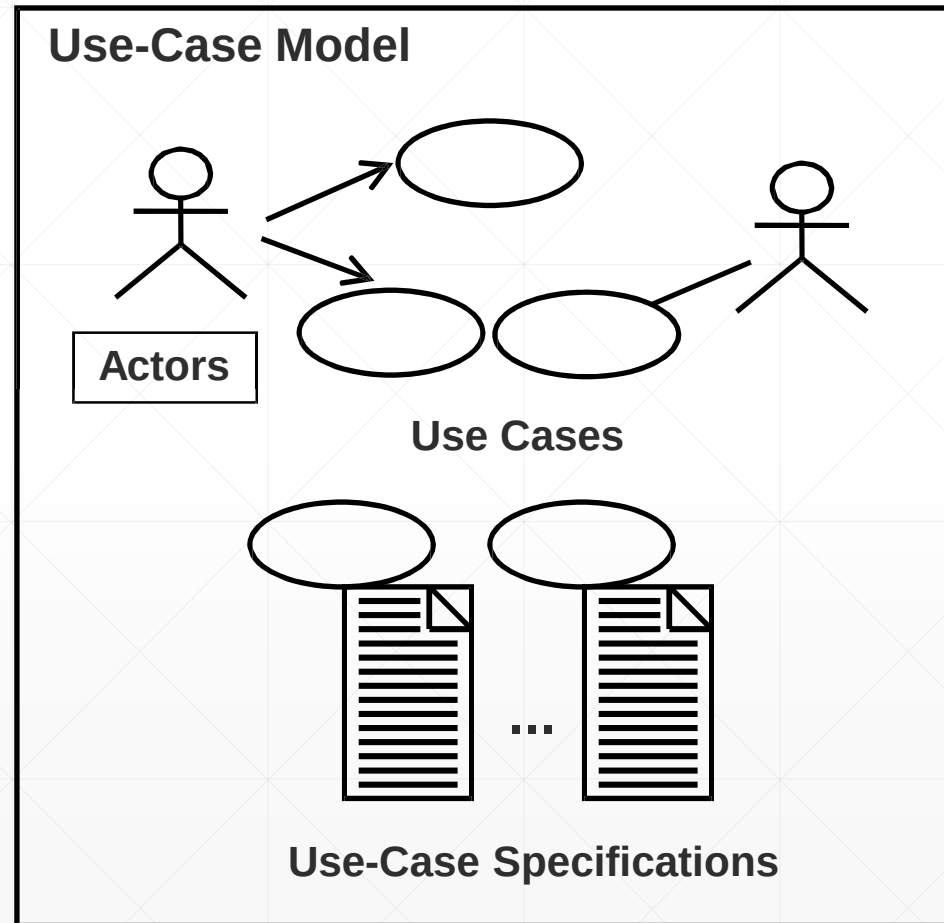


Use Case Model

Use case diagram

Relevant Requirements Artifacts



Glossary



Supplementary
Specification

What Is System Behavior?

System behavior is how a system acts and reacts.

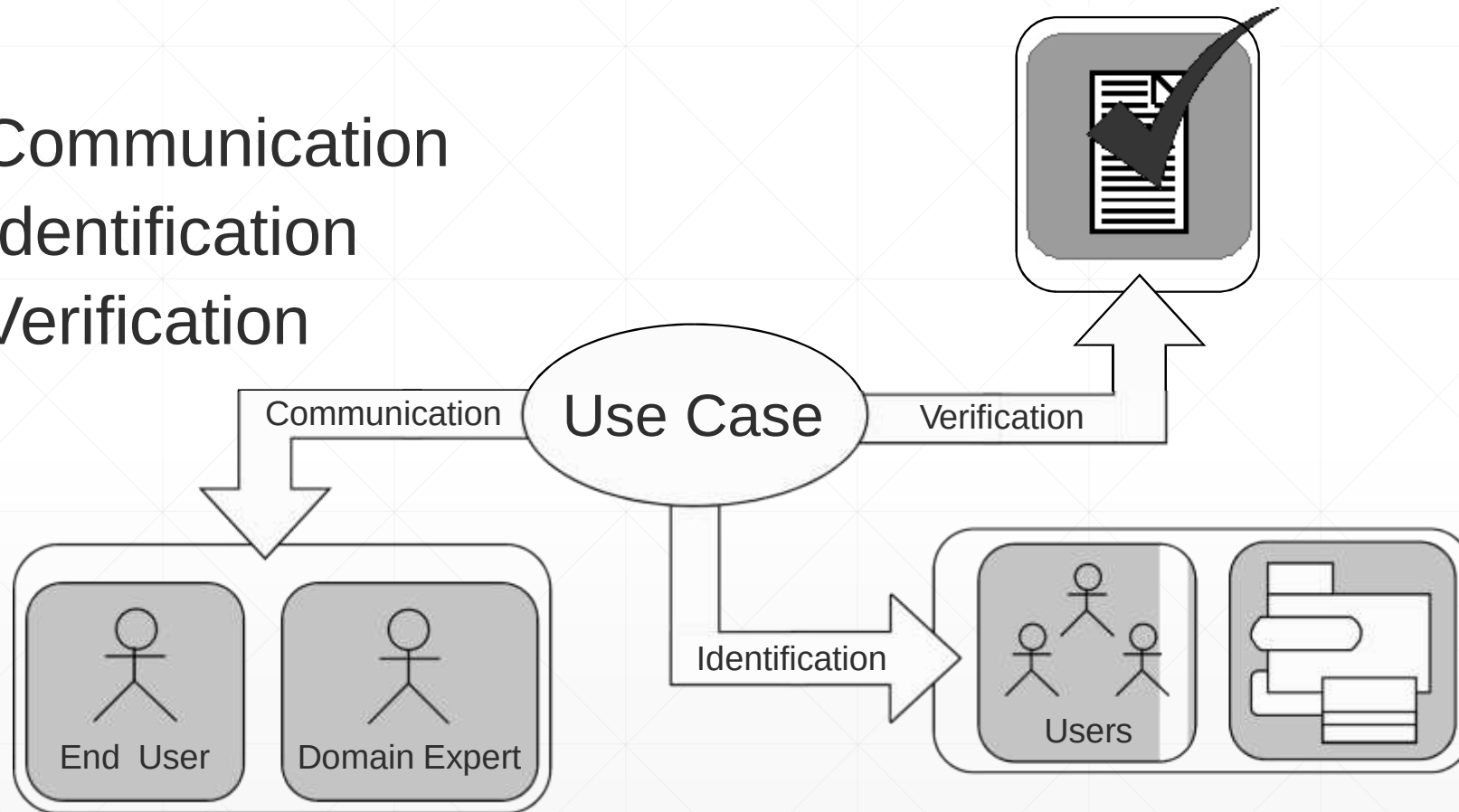
- It is the outwardly visible and testable activity of a system.

System behavior is captured in use cases.

- Use cases describe the system, its environment, and the relationship between the system and its environment.

What Are the Benefits of a Use-Case Model?

- Communication
- Identification
- Verification



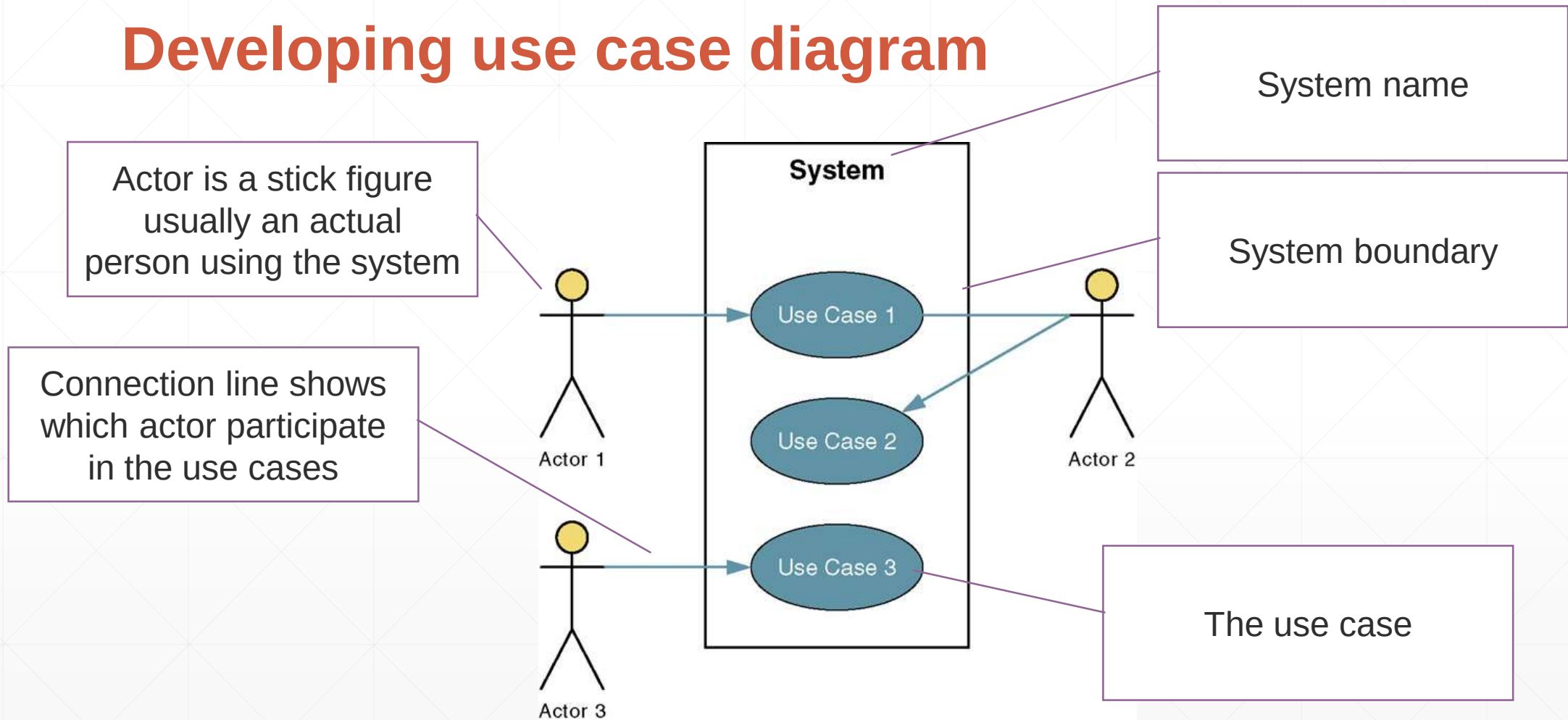
Key points

- The development team, with stakeholders involvement, writes the use cases.
 - Compared to traditional requirement methods, use cases are relatively easy to write and easier to read.
 - Free of technical or implementation details.
-

Use Case Diagrams

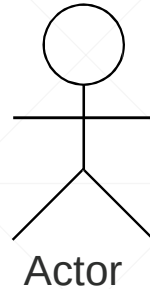
- ❑ A use case diagram depicts the interactions between the system and the external actors.

Developing use case diagram



Major Concepts in Use-Case Modeling

- An actor represents anything that interacts with the system.
- An actor might be:
 - a person
 - a company or organization,
 - a computer program, or a computer system
 - Hardware

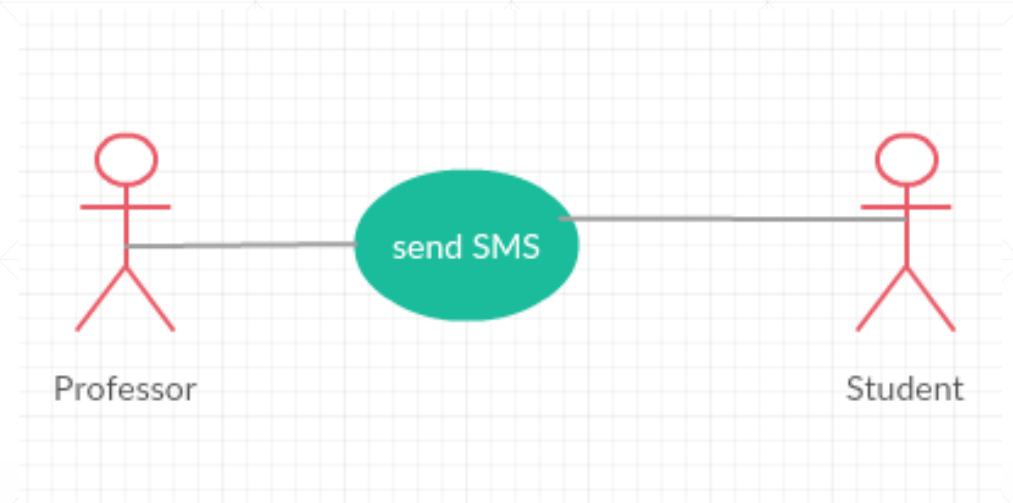


- A use case is a sequence of actions a system performs that yields an observable result of value to a particular actor.
 - Name each use case using Noun-Verb



Types of actor

- Primary actor initiates the use case.
- Secondary actor is used to complete the use case.
- E.g. the professor send SMS to the students. The professor considered as primary actor, whereas students considered as secondary actor



A Step-by-Step Guide to Building the Use-Case Model

- **Step 1: Identify and Describe the Actors**
 - Who uses the system?
 - Who gets information from this system?
 - Who provides information to the system?
 - Where in the company is the system used?
 - Who supports and maintains the system?
 - What other systems use this system?

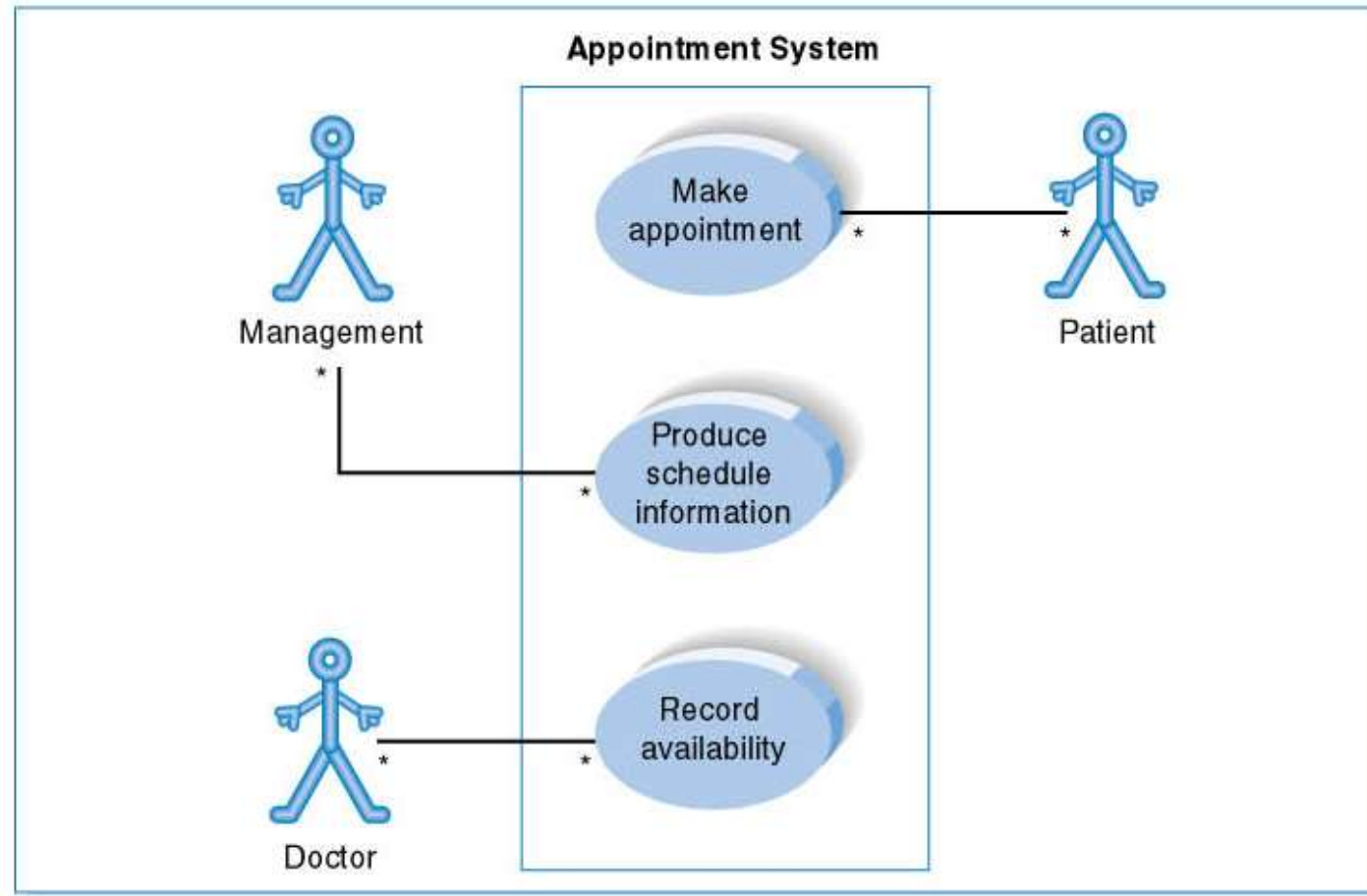
A Step-by-Step Guide to Building the Use-Case Model

- **Step 2: Identify the Use Cases and Write a Brief Description**
 - What will the actor use the system for?
 - Will the actor create, store, change, remove, or read data in the system?
 - Will the actor need to inform the system about external events or changes?
 - Will the actor need to be informed about certain occurrences in the system?

A Step-by-Step Guide to Building the Use-Case Model

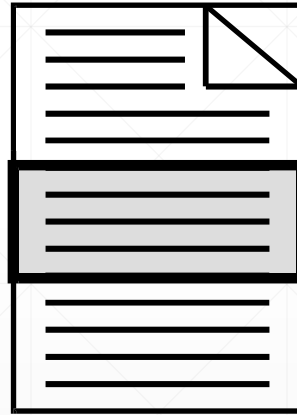
- **Step 3: Identify the Actor and Use-Case Relationships**
 - Only one actor can initiate a use case
 - Many use cases may involve the participation of multiple actors.
 - Each use case is analyzed to see what actors interact with it

Use case diagram



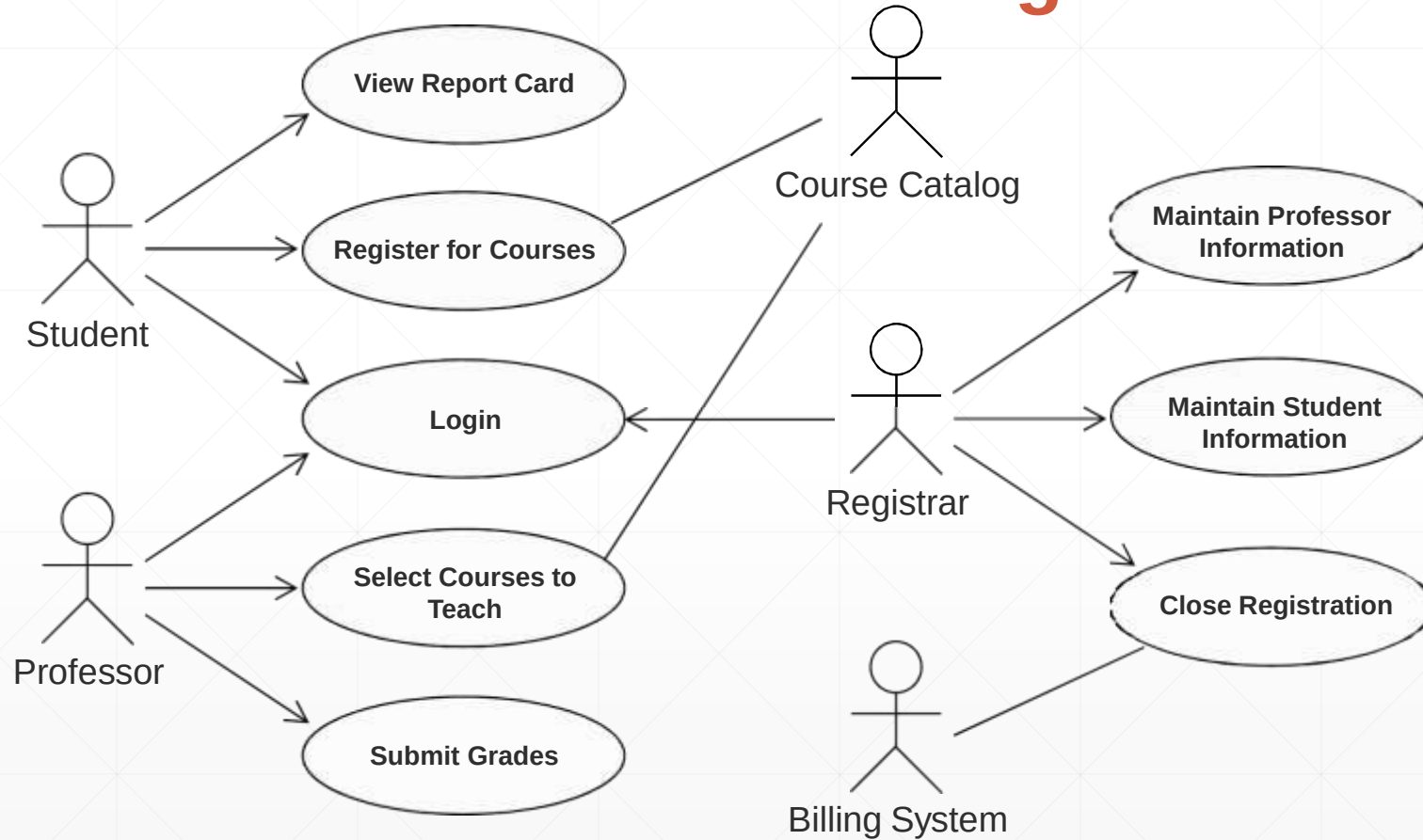
Case Study: Course Registration Problem Statement

Review the problem statement provided in the Course Registration Requirements Document.



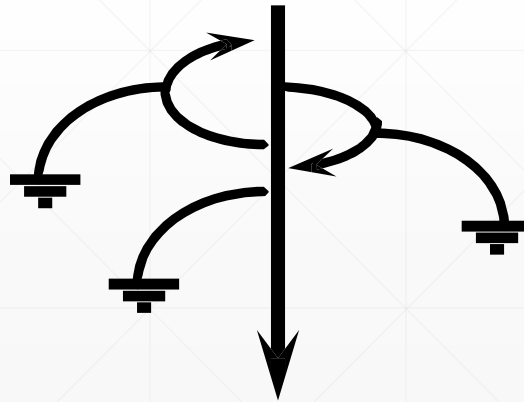
Course Registration
Requirements Document

How Would You Read This Diagram?



Use-Case Flow of Events

- Has one normal, *basic flow*
- Several *alternative flows*
 - Regular variants
 - Odd cases
 - Exceptional flows for handling error situations



Relationships between Use Cases

1. Include - use cases that are included as parts of other use cases. Enable to factor common behavior.
 2. Extend - use cases that extend the behavior of other core use cases. Enable to factor variants.
-

Includes and Extends

- Includes

- You have a piece of behavior that is similar across many use cases
- Break this out as a separate use-case and let the other ones “include” it
- Examples include
 - Valuation
 - Validate user interaction
 - Sanity check on sensor inputs
 - Check for proper authorization



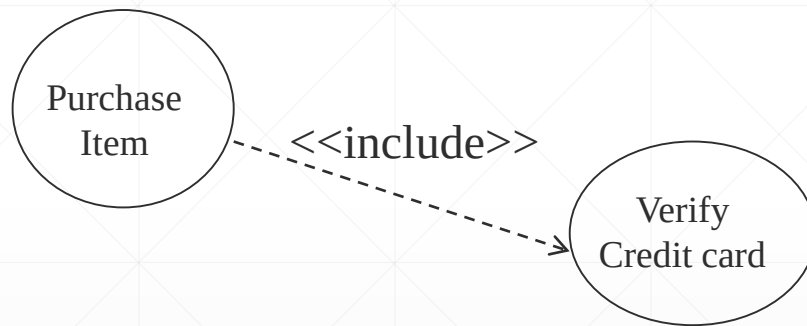
- Extends

- A use-case is similar to another one but does a little bit more
- Put the normal behavior in one use-case and the exceptional behavior somewhere else
 - Capture the normal behavior
 - Try to figure out what can go wrong in each step
 - Capture the exceptional cases in separate use-cases
- Makes it a **lot** easier to understand



Examples

- Include



- Extend

