FARABI UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

Lecture 3.
Use Case Diagrams

Actors, user goals, system scope and UML relationships

Dr. Bazargul Matkerim, Professor-Researcher

Lecture goals and plan

Learning outcomes

Identify actors and their goals.

Define the system boundary.

Write a use-case scenario.

Apply include, extend and generalization correctly.

Lecture plan

1. Concepts and actor discovery
2. Notation and descriptions
3. Relationships and examples
4. Registration and vending cases
5. Modeling practice and review

Use case, scenario and user story

Concept	Meaning	Registration example
Use case	Goal-oriented behavior with related paths	Register for courses
Scenario	One specific path through the behavior	All selected courses are available
User story	A brief statement of a need and value	As a student, I want to choose courses so I can plan my semester

The diagram provides an overview; the description explains the behavior.

Actors are roles outside the system

Who or what can act?

A person playing a role

An external organization

Another software system

An external device or sensor

Roles, not individuals

One person may play several roles.

Many people may play the same role.

An actor can initiate or support an interaction.

Internal components are not actors.

Discover actors and meaningful goals

Ask about the environment

Who starts the interaction?

Who receives a useful result?

Which external services participate?

Who maintains or administers the system?

Turn answers into goals

Student □ Register for courses

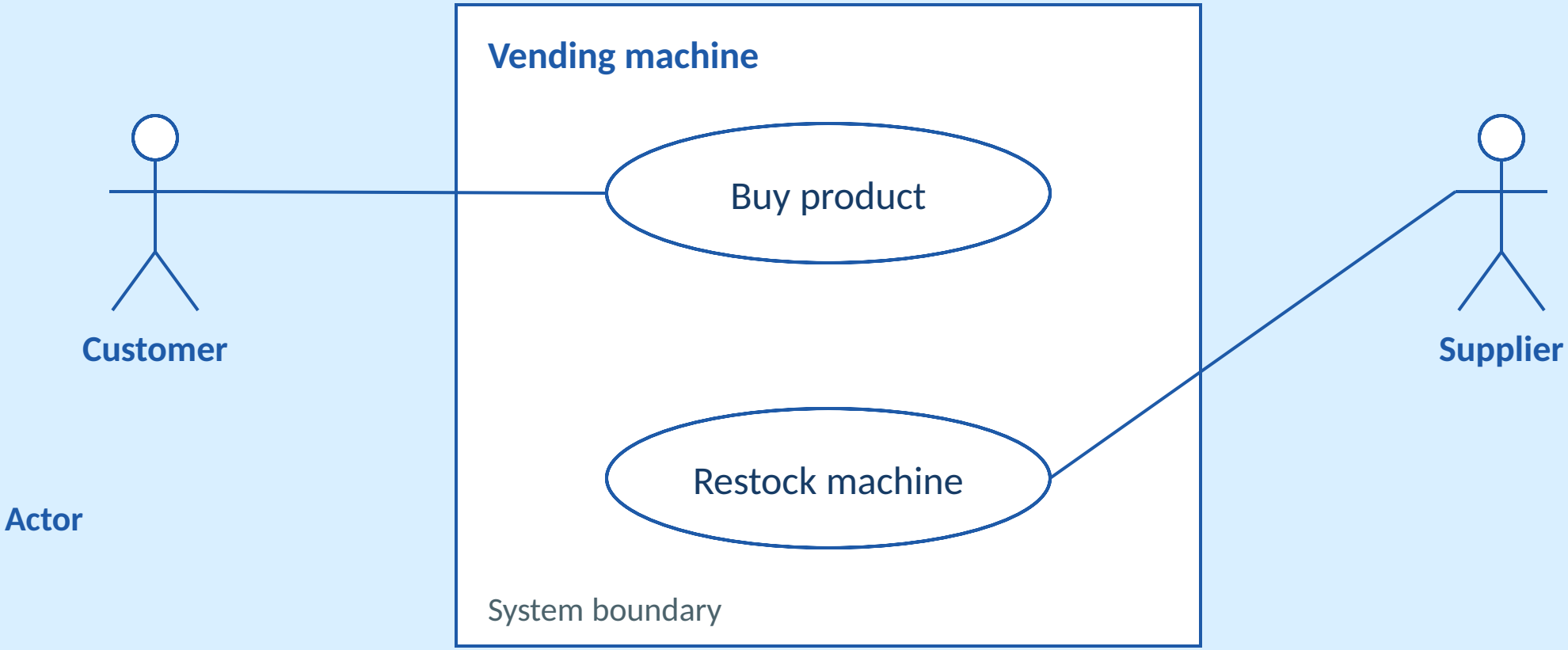
Professor □ View course roster

Supplier □ Restock machine

Customer □ Buy product

Name actors with role nouns and use cases with verb phrases.

Basic notation: a vending machine



Oval = use case. Solid line = participation. Actors remain outside the boundary.

Write a description behind each oval

Field	Example: Buy product
Primary actor / trigger	Customer selects a product
Preconditions	Machine is operational; product selection is available
Success guarantee	Product delivered; payment recorded
Main flow	Select □ check availability □ pay □ dispense □ confirm
Alternative flows	Sold out; insufficient credit; cancellation; dispensing failure

Describe success and alternatives

Main success scenario

1. Customer selects a product.
2. System shows price and stock.
3. Customer provides payment.
4. System accepts payment.
5. System dispenses the product.
6. System confirms the purchase.

Alternative scenarios

2a. Sold out: request another choice.

4a. Insufficient credit: request more payment or cancel.

5a. Dispensing failure: apply the agreed refund/recovery rule.

Keep the model at the right level

User-level goal

Place order

The actor receives a confirmed order.

This provides a useful unit for discussion, requirements and acceptance tests.

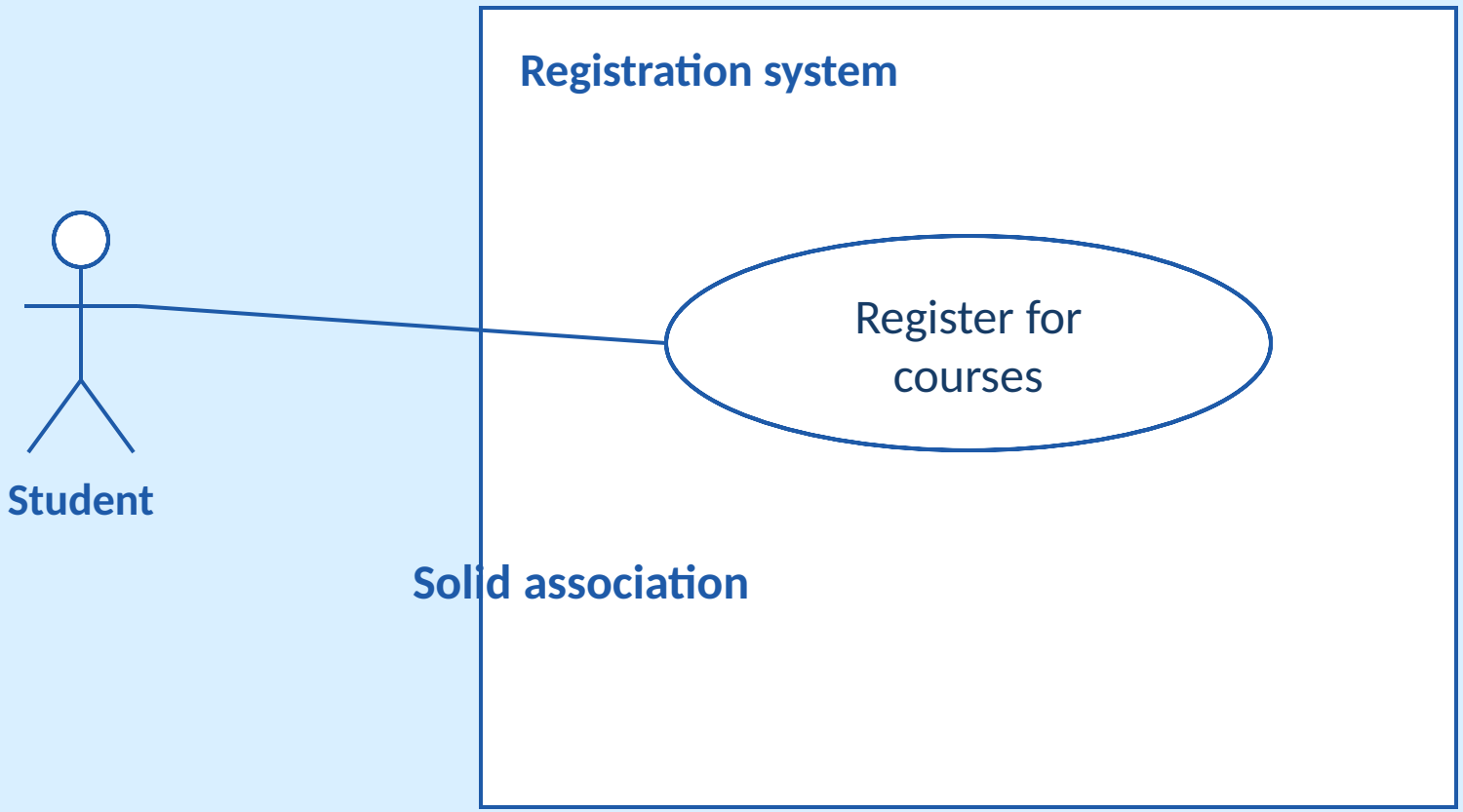
Steps within that goal

Browse catalog
Enter shipping information
Provide payment information
Receive confirmation

Do not turn every step into a separate user-level use case.

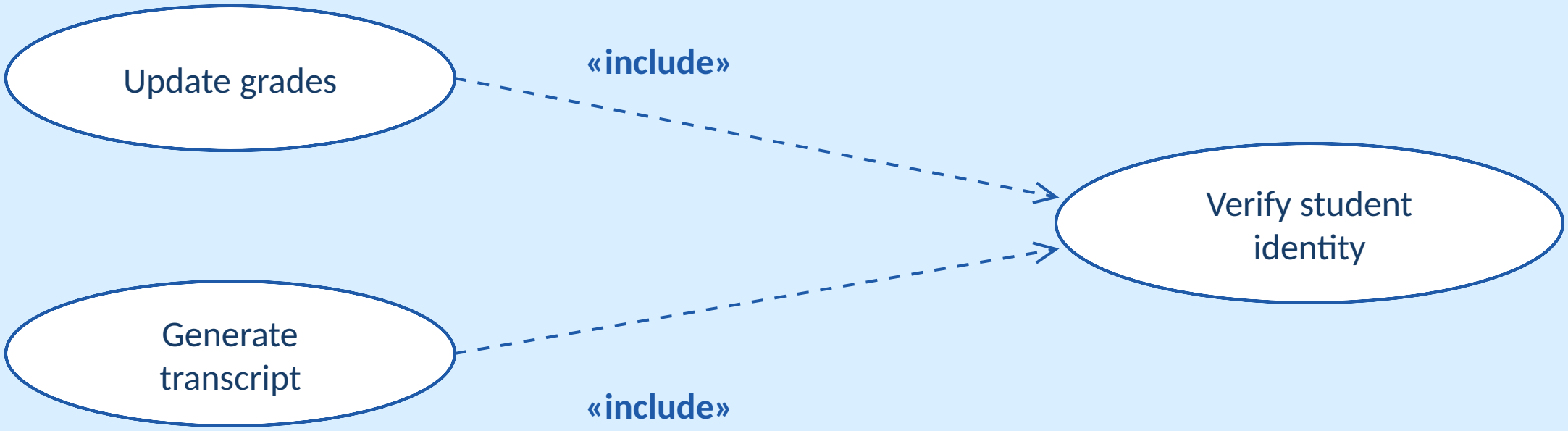
Use an activity or sequence diagram when interaction order is the main question.

Association means participation



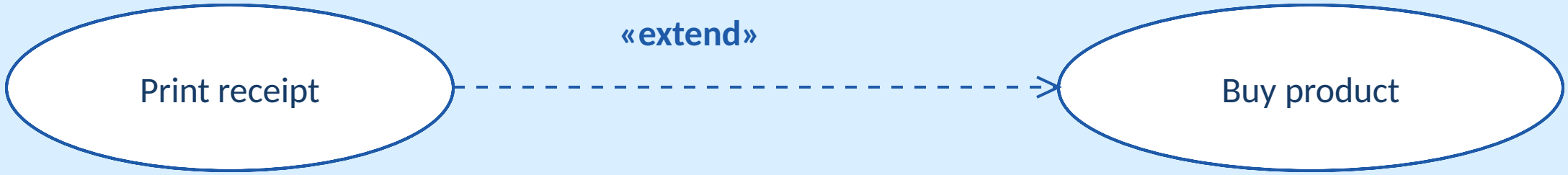
The Student role interacts with this use case.
The line does not describe control flow or timing.

Include: factor shared required behavior



The arrow points from the including use case to the included behavior.

Extend: add behavior at an extension point



Condition

[Customer requests a receipt]

Extension point in Buy product

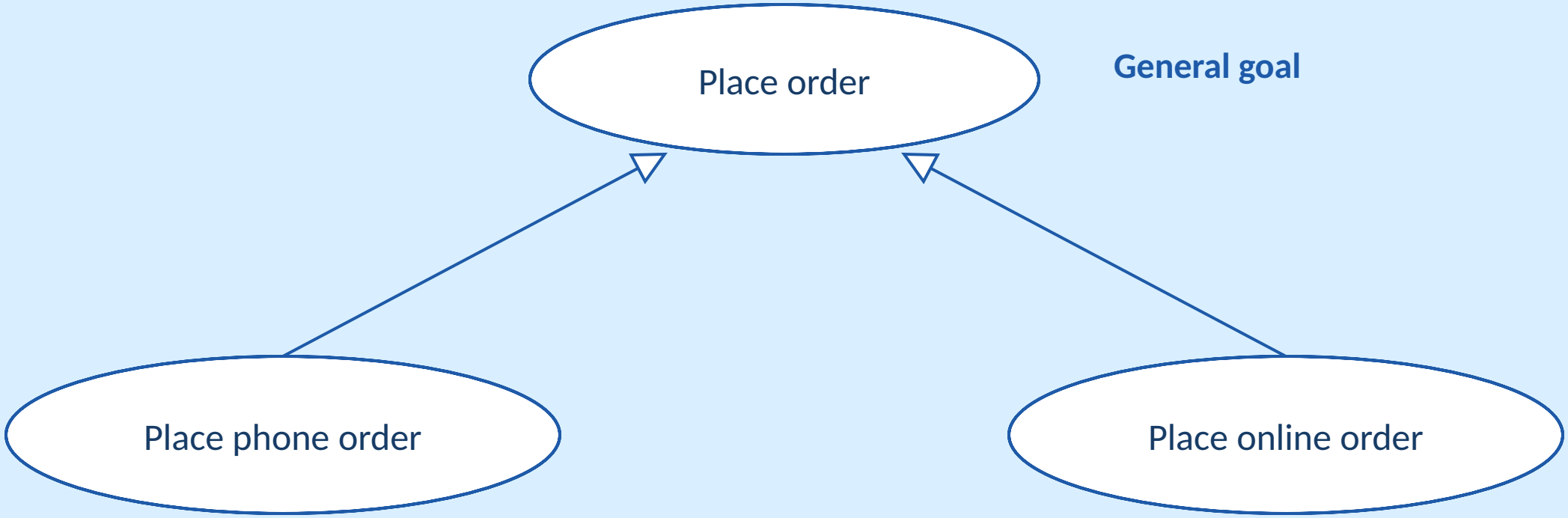
After purchase confirmation

Include and extend: compare the intent

Question	«include»	«extend»
Why use it?	Factor behavior used within another case	Add behavior at an extension point
Which way is the arrow?	Including case □ included case	Extension □ base case
What does the base need?	Included behavior at its inclusion step	A complete goal without the extension
Example	Update grades □ Verify student identity	Print receipt □ Buy product

Use relationships to express meaning—not simply to make the diagram look detailed.

Generalization: specialize a use case



Solid line with a hollow triangle; the triangle points to the more general element.

Case study: course registration

Actors and goals

Student: select, add or drop courses

Professor: state teaching preferences and view rosters

Registrar: maintain course information

Billing system: receive enrollment data

Rules from the source case

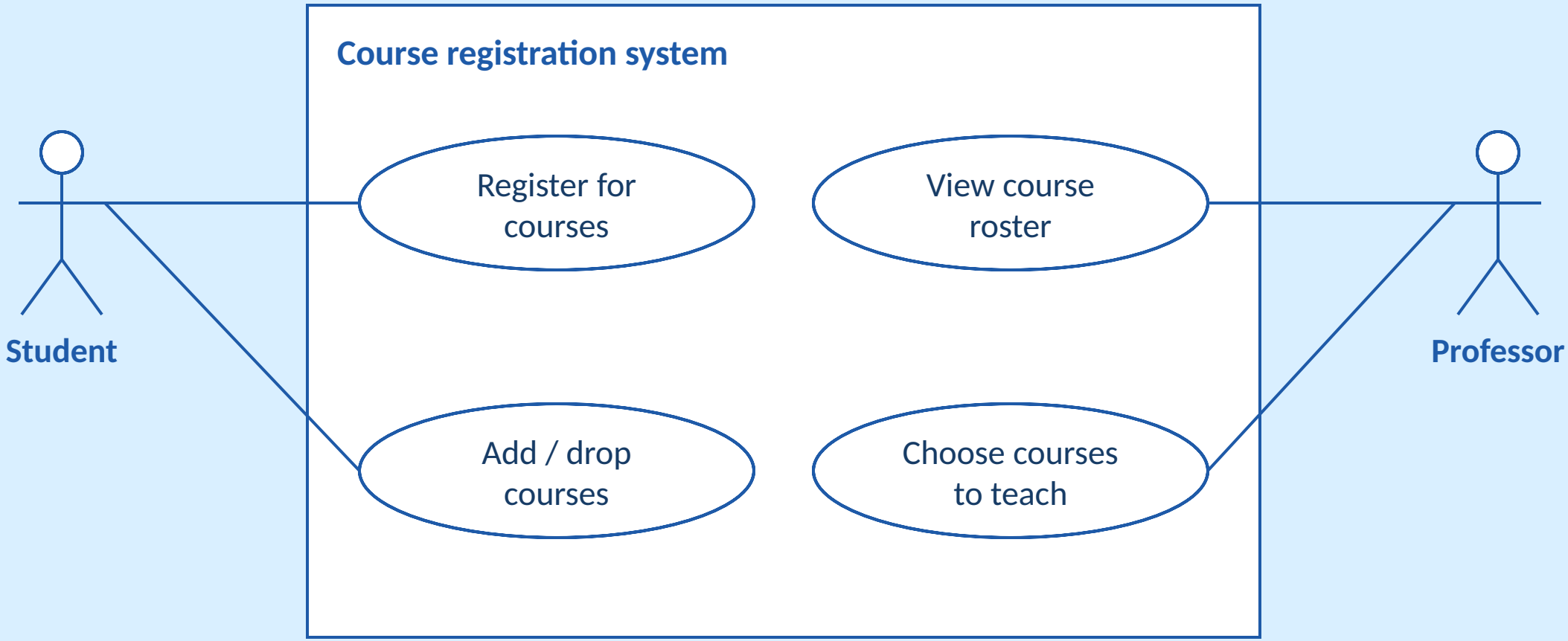
Students select four courses.

A course runs with 3–10 students.

Completed registration sends information to billing.

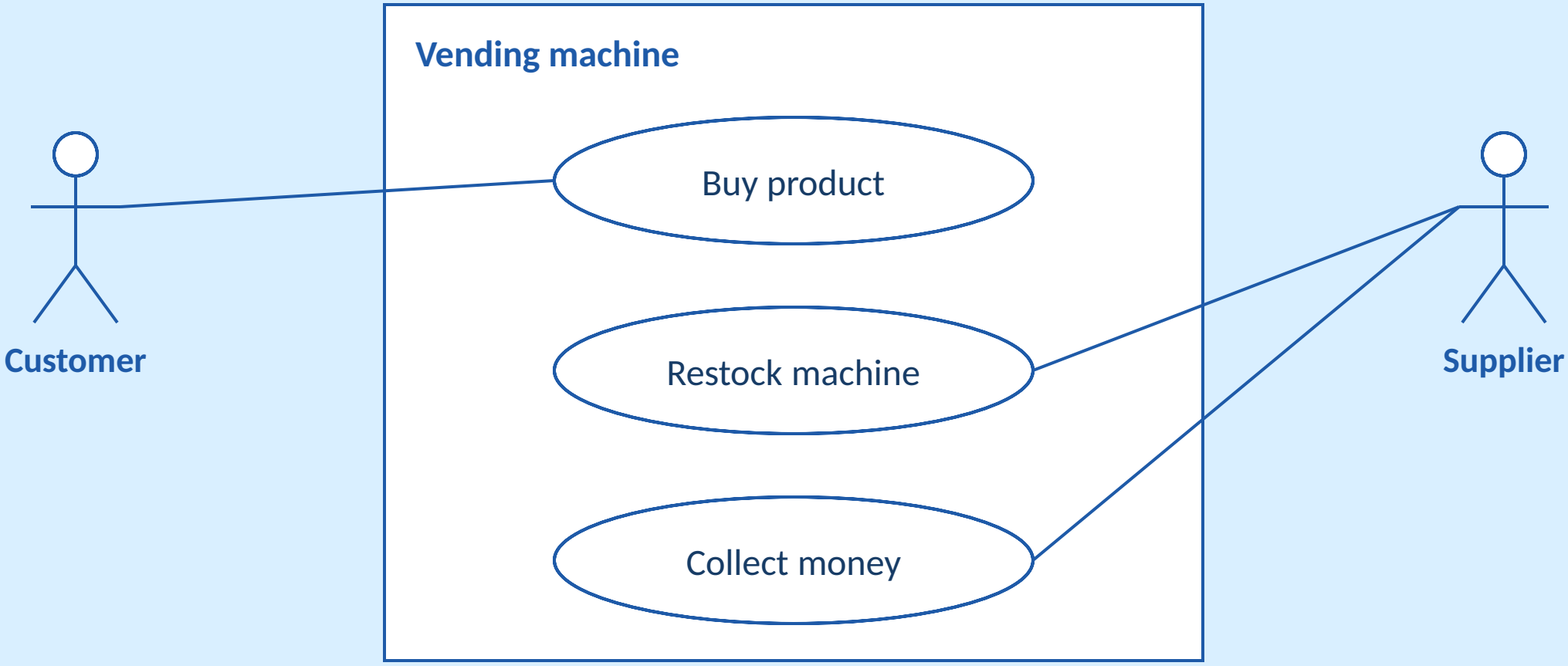
Timing and add/drop rules need clarification.

Registration: a focused use-case diagram



Focused view: student and professor goals. Administrative and billing views can be separate.

Vending machine: roles and constraints



Assumption: the supplier also plays the money-collector role. Validate this with the client.

A repeatable modeling workflow

Build the model

1. Define scope and system boundary.
2. Identify external roles and goals.
3. Write a success scenario.
4. Add alternatives and constraints.

Refine with stakeholders

5. Draw actors, goals and associations.
6. Add only justified relationships.
7. Review names, scope and direction.
8. Derive tests from the scenarios.

Review and classroom practice

Explain your model

What is inside the boundary?

Can a supporting actor start nothing?

Which direction does extend point?

When is a text alternative enough?

Practice: appointment booking

Identify Patient, Receptionist and any external service.

Write a successful booking scenario and two alternatives.

Draw 3–5 user-level use cases.

Justify each relationship.

Reading and references

Core reading

Dennis, A., Wixom, B. H., & Tegarden, D. (2021).
Systems Analysis and Design: An Object-Oriented Approach
with UML, 6th ed. Wiley. ISBN 9781119559917.

UML notation reference

Object Management Group (2017). Unified Modeling Language,
Version 2.5.1. Chapter 18: UseCases.

Full references and source-page mapping are included in the speaker notes.