

# Requirements Analysis Exercise

# System Description

A university intends to develop a web-based platform called **UniStudy** to support students in organizing their study activities and preparing for exams.

The system should allow students to create a personal study environment where they can manage the exams they plan to take, organize their study schedule over time, and keep track of their progress while studying different subjects. Students are expected to interact with the system regularly, updating their activities and monitoring their preparation.

# System Description (2)

In addition to student functionalities, the platform should also be accessible to university teachers. Teachers should be able to provide useful information related to their courses, such as sharing teaching materials and communicating important academic dates. These resources should be clearly associated with specific exams so that students can easily find relevant content. The system is expected to support multiple users simultaneously and provide a consistent experience across different devices, including laptops, tablets, and smartphones.

# System Description (3)

Furthermore, the platform should assist students by offering feedback on their study habits, such as summarizing their progress over time or highlighting their level of preparation for each exam. The system may also notify users about planned activities or upcoming deadlines, helping them stay organized throughout the academic term.

The overall goal of UniStudy is to provide an integrated environment that improves students' ability to plan, monitor, and complete their study activities effectively.

# Exercise Overview

We will design the requirements for a software system.

Steps:

1. Understand the problem
2. Identify actors and use cases
3. Define user requirements
4. Derive system requirements
5. Identify non-functional requirements

# Step 1 – Actors

## Question

Who interacts with the system?

# Step 1 – Actors

## Solution

Actors:

- Student
- Teacher

(optional discussion: Admin?)

# Step 2 – Use Cases

## Question

What are the main actions users can perform?

(High-level only)

# Step 2 – Use Cases

Student:

- Register/Login
- Manage exams
- Plan study sessions
- Track progress
- Receive notifications

Teacher:

- Upload materials
- Publish exam dates

# Step 2 – Diagram

## Task

Draw a **Use Case Diagram (overview)**

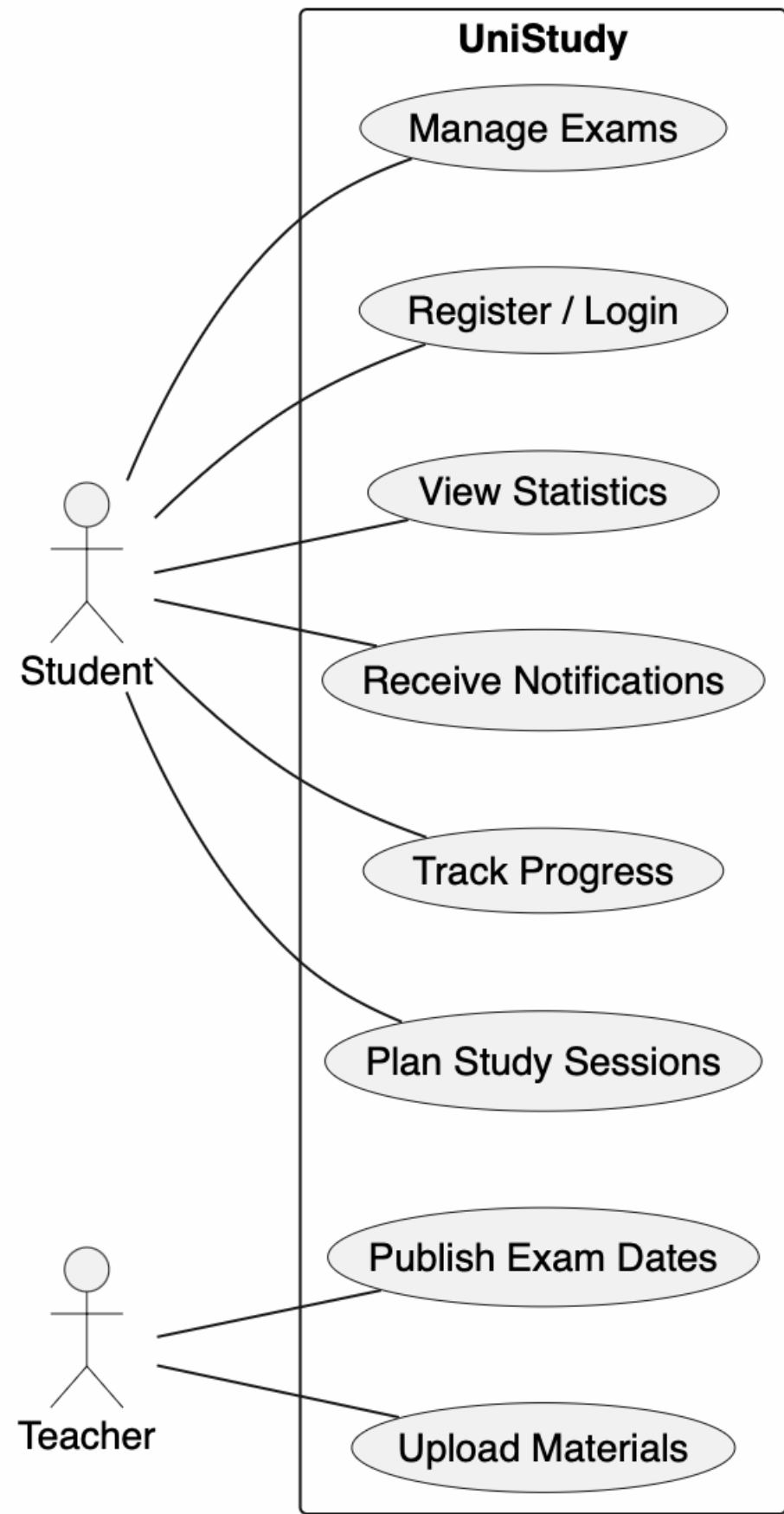
Constraints:

- One level only
- No include/extend

# **Step 2 – Diagram Solution (Conceptual)**

Actors:

- Student → (Manage Exams, Plan Study, Track Progress, Receive Notifications)
- Teacher → (Upload Materials, Publish Exam Dates)



# **Step 3 – User Requirements**

## **Question**

Define 3 user requirements

# **Step 3 – User Requirements**

## **Solution**

1. The student shall be able to plan study sessions for each exam
2. The student shall be able to track their study progress
3. The teacher shall be able to upload teaching materials

# Step 4 – System Requirements

## Question

For each user requirement:

→ define at least 2 system requirements

# Step 4 – System Requirements

## Solution (UR1)

User Requirement:

"The student shall be able to plan study sessions for each exam"

System Requirements:

- The system shall allow the user to create a study session with date and time
- The system shall store study sessions in a persistent database

# Step 4 – System Requirements

## Solution (UR2)

User Requirement:

"The student shall be able to track their study progress"

System Requirements:

- The system shall allow users to mark topics as completed
- The system shall compute and display progress percentage

# **Step 4 – System Requirements Solution (UR3)**

User Requirement:

"The teacher shall be able to upload teaching materials"

System Requirements:

- The system shall allow file upload (PDF, slides)
- The system shall associate materials with a specific exam

# **Step 5 – Non-Functional Requirements Question**

Identify at least 2 non-functional requirements

# Step 5 – Non-Functional Requirements Solution

Performance:

- The system shall respond within 2 seconds for user actions

Usability:

- The system shall be accessible from mobile devices

# Step 5 – Additional Examples

Security:

- User data shall be protected with authentication

Reliability:

- The system shall guarantee 99% uptime






# Scrum Exercise



**Sprint Planning Exercise –  
UniStudy**

# What we'll do

**25 minutes, in groups of 4-5**

1.  Read the backlog
2.  Assign roles
3.  Estimate and plan your sprint
4.  Simulate a Daily Scrum
5.  Debrief together

*No coding required. Just thinking and negotiating.*

# The Scenario

*You are a small development team working on UniStudy.*

*The university has asked you to deliver a first working version of the platform.*

*You have two weeks (one sprint) to build core features.*

# The Product

The platform should support students in:

- planning study sessions
- tracking their progress
- accessing course materials
- receiving notifications

# Step 1 – Assign Roles

 **2 minutes**

<b>Role</b>	<b>Responsibility</b>
<b>Product Owner</b>	Priorities & business value
<b>Scrum Master</b>	Facilitates, removes blockers
<b>Dev Team (2-3)</b>	Estimates effort, plans work

 *Product Owner = the voice of the university*

# Step 2 – Your Product Backlog

 **5 minutes**

ID	User Story	Priority
US1	Plan study sessions	★★★ High
US2	Track study progress	★★★ High
US3	Upload teaching materials	★★ Medium
US4	Receive notifications	★★ Medium
US5	View study statistics	★ Low
US6	System works on mobile devices	★★ Medium
US7	Login functionality	★ Low
US8	Performance: response < 2s	★★ Medium

# Step 3 – Sprint Planning

 **10 minutes**

**Your team capacity:**

*3 developers × 8 days × 6 hours = 144 hours*

*1 story point ≈ 8 hours → ~18 points available*

# Estimation scale

**Points**

**Effort**

---

1

A few hours

---

3

About one day

---

5

Two days

---

8

More than three days

# Step 3 – Estimate These Stories

User Story	Your estimate
US1 – Plan study sessions	? pts
US2 – Track progress	? pts
US3 – Upload materials	? pts
US4 – Notifications	? pts
US6 – Mobile support	? pts




 *Discuss as a team.*

*What fits in 18 points?*

# Step 4 – Daily Scrum (Simulated)

 **5 minutes**

Each developer answers:

1.  What did I do yesterday?
2.  What will I do today?
3.  Any blockers?

*Synchronisation, not reporting.*

# **Questions?**

**What would you improve in this system?**