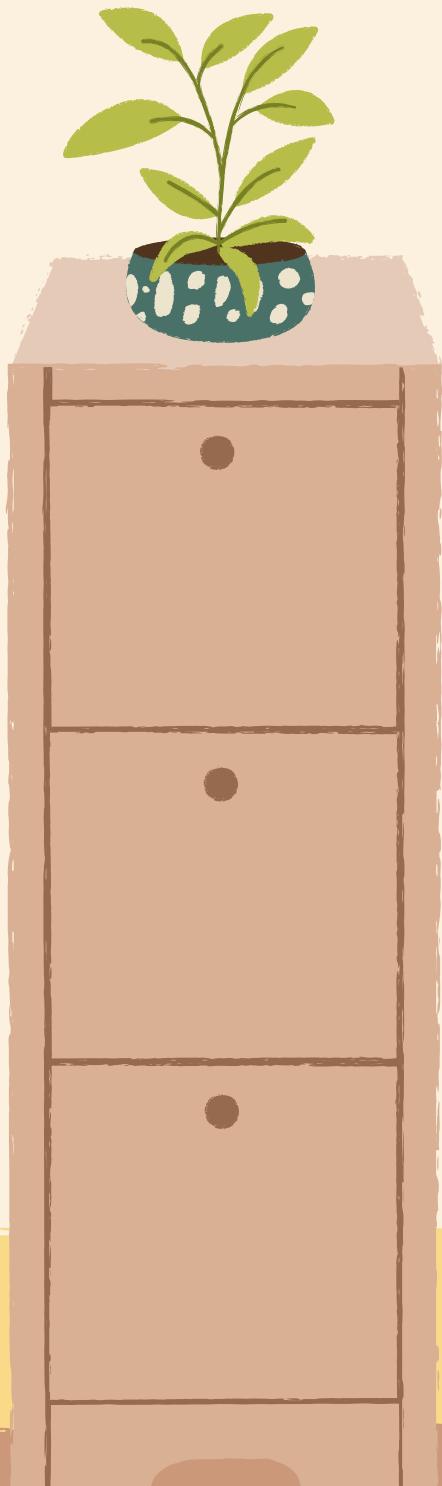


# **PRESENTATION PROJECT**

## **SPRINT 0**



# OUR TEAM



# GENERAL INTRODUCTION

The digital shift in language learning isn't just about teaching—it's about management. Today, an English school's success depends on seamless administration, clear communication, and centralized oversight. That's where a dedicated platform becomes essential.

# INTRODUCTION

This project involves the design and development of an integrated web platform for the complete management of an English language school. The objective is to provide a single digital environment capable of managing the learner lifecycle, from registration to certification, while optimizing the planning of human and material resources.

# PROBLEM STATEMENT

Despite striving for excellence, many language schools face significant operational challenges:

Data Dispersion: The use of disparate files makes tracking payments and absences complex

Scheduling: Difficulty coordinating teacher availability, classrooms, and different levels .

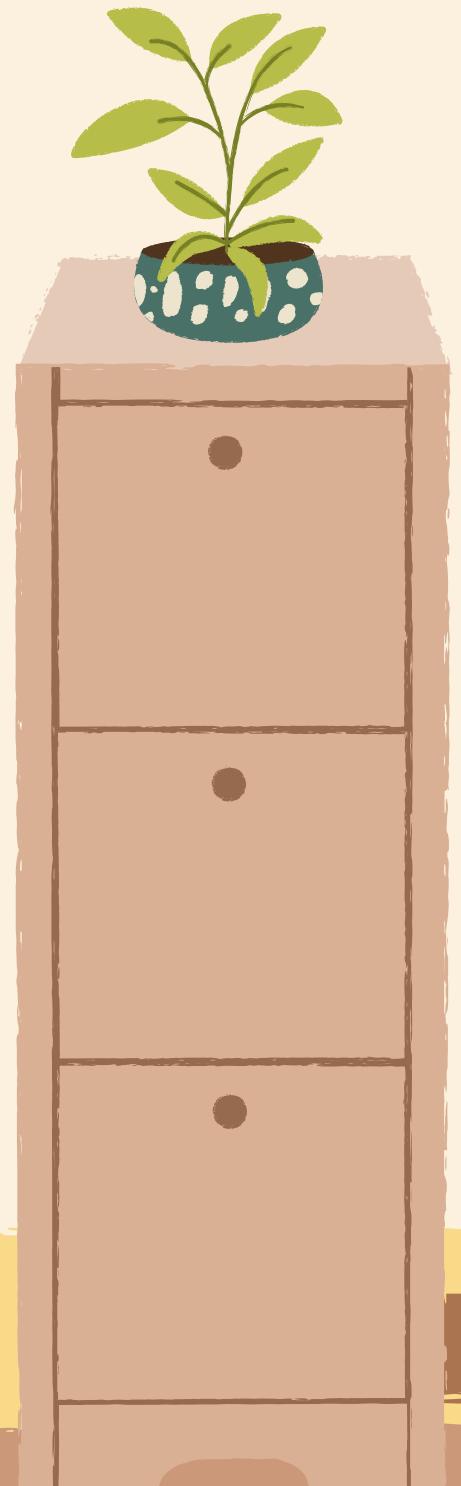
# PROBLEM STATEMENT

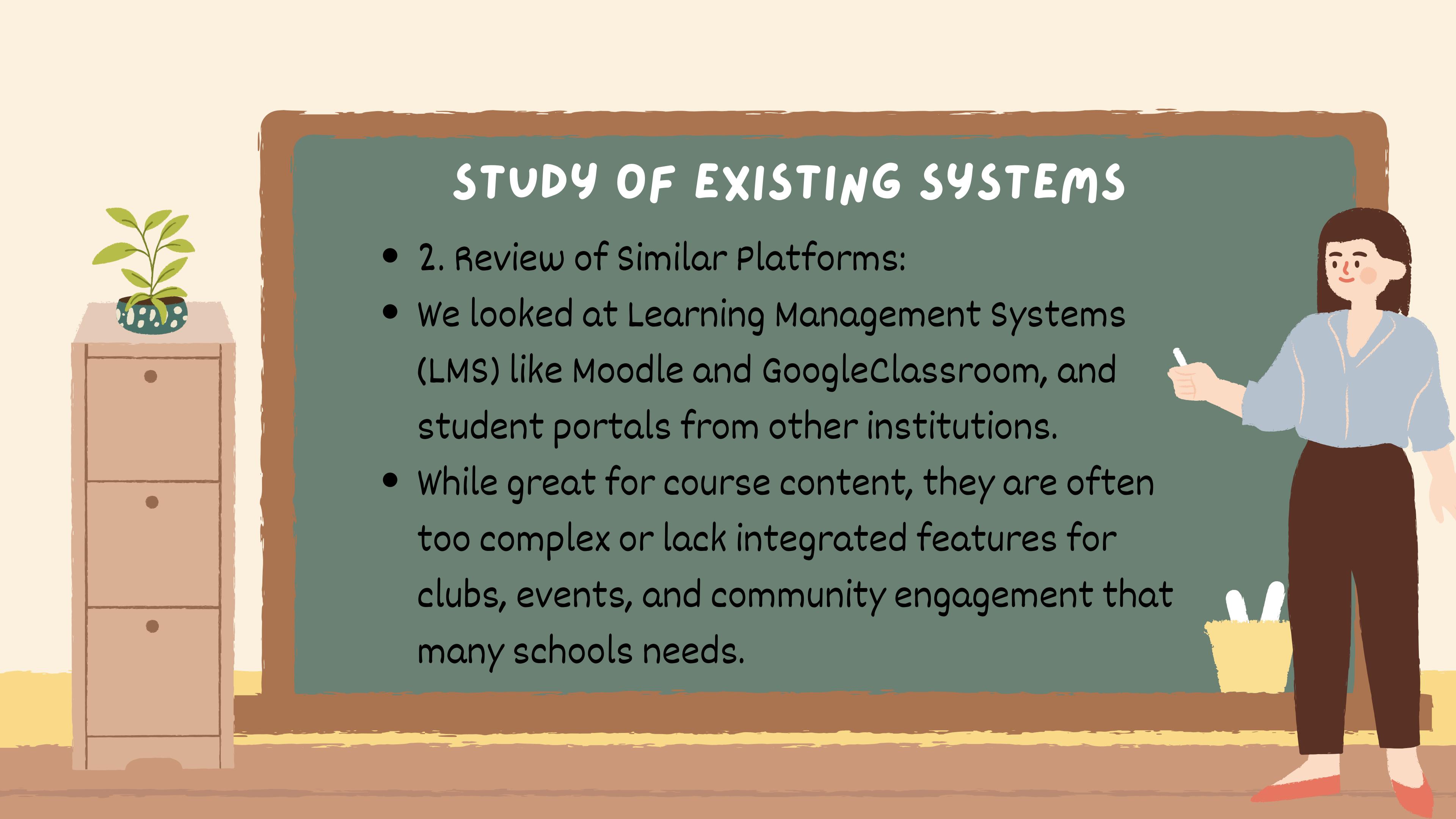
Lack of Communication: The absence of a direct channel for transmitting course materials or results to students in real time.

- Manual Academic Monitoring: The administrative burden of generating progress reports or level certificates.

# STUDY OF EXISTING SYSTEMS

- 1. Current School Process:
- Manual & Fragmented: Heavy use of paper, emails, WhatsApp, and simple spreadsheets.
- Inefficient: Scheduling, tracking attendance, and sharing materials take significant time.
- Disconnected: students, tutors, and administrators lack a central place to communicate and access information.





## STUDY OF EXISTING SYSTEMS

- 2. Review of Similar Platforms:
- We looked at Learning Management Systems (LMS) like Moodle and Google Classroom, and student portals from other institutions.
- While great for course content, they are often too complex or lack integrated features for clubs, events, and community engagement that many schools need.

# OUR SOLUTION “FLUENCY”

FLUENCY is the all-in-one digital platform for English language schools, designed to simplify and connect every part of the learning experience. It provides students, tutors, and administrators with a unified space to manage courses, quizzes, clubs, events, and communication—all in one intuitive, engaging environment.

# "VISUAL IDENTITY"

## OUR LOGO

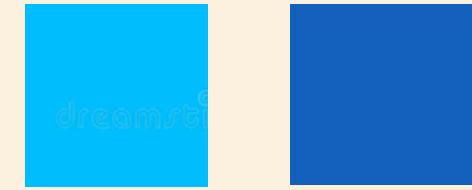


**FLUENCY**

## OUR SLOGAN

Redefining English Learning!

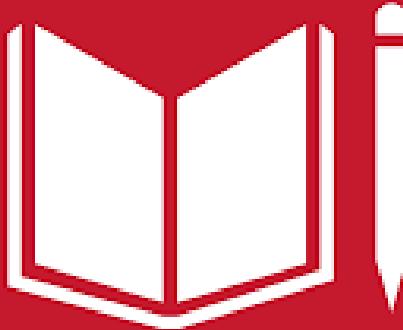
## CHOSEN COLORS



Two variations of blue to signify depth and clarity

# SDQ's THAT MATCH OUR PLATFORM

4 QUALITY EDUCATION



9 INDUSTRY, INNOVATION AND INFRASTRUCTURE



- ✓ ACCESSIBLE ENGLISH LEARNING FOR ALL
- ✓ DIGITAL LITERACY AND LIFELONG LEARNING
- ✓ EQUAL OPPORTUNITIES THROUGH TECHNOLOGY

- ✓ MICROSERVICES ARCHITECTURE INNOVATION
- ✓ SCALABLE CLOUD-BASED INFRASTRUCTURE
- ✓ MODERN TECH STACK (ANGULAR, SPRING BOOT)
- ✓ SUSTAINABLE DIGITAL TRANSFORMATION

# FUNCTIONAL REQUIREMENTS

(WHAT THE SYSTEM DOES)

- User Management: Registration, login, and role-based profiles for Students, Tutors, and Administrators.
- Course & Session Management: Enrollment, booking, scheduling, and attendance tracking.
- Content & Assessment System: Upload and access lessons, articles, and interactive quizzes.

# FUNCTIONAL REQUIREMENTS

(WHAT THE SYSTEM DOES)

- Community Features: Create, join, and manage clubs and events; participate in discussions.
- Notification System: Real-time alerts for bookings, deadlines, announcements, and messages.
- Administrative Dashboard: Manage users, content, activities, and generate reports.

# NON-FUNCTIONAL REQUIREMENTS

(HOW THE SYSTEM PERFORMS)

Scalability and Stability :

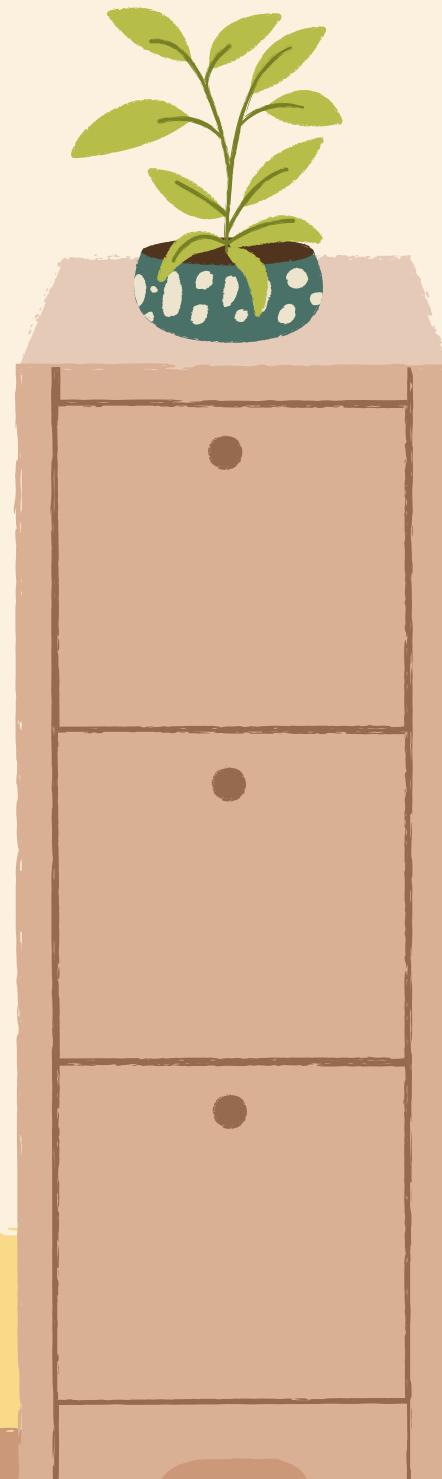
For stable platform during peak usage

(exams, registrations) we are going to use :

- Stateless backend using JWT
- Optimized REST APIs
- Connection pooling in Spring Boot
- Async calls using RxJS (Angular)

Example:

✓ 500 students can take QCMs simultaneously without conflicts



# NON-FUNCTIONAL REQUIREMENTS

(HOW THE SYSTEM PERFORMS)

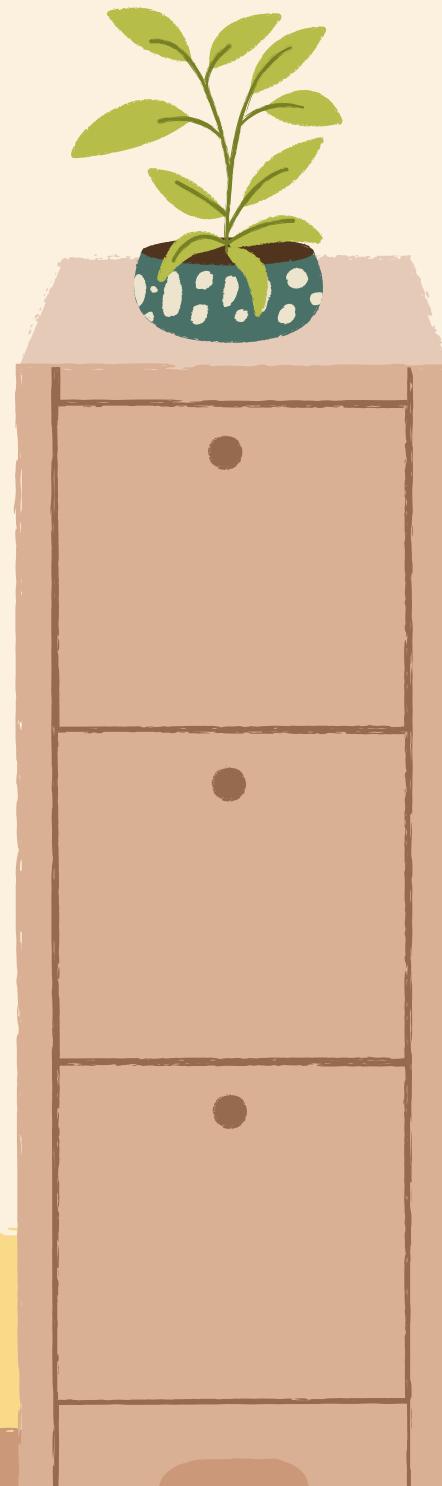
## Security:

To protect users and educational data we are going to use:

- JWT authentication with token expiration
- Role-based access (Student / Tutor / Admin)
- Password encryption using BCrypt
- Secure communication via HTTPS

## Example:

- ✓ Students cannot access admin features
- ✓ Tokens expire after 1 hour



# NON-FUNCTIONAL REQUIREMENTS

(HOW THE SYSTEM PERFORMS)

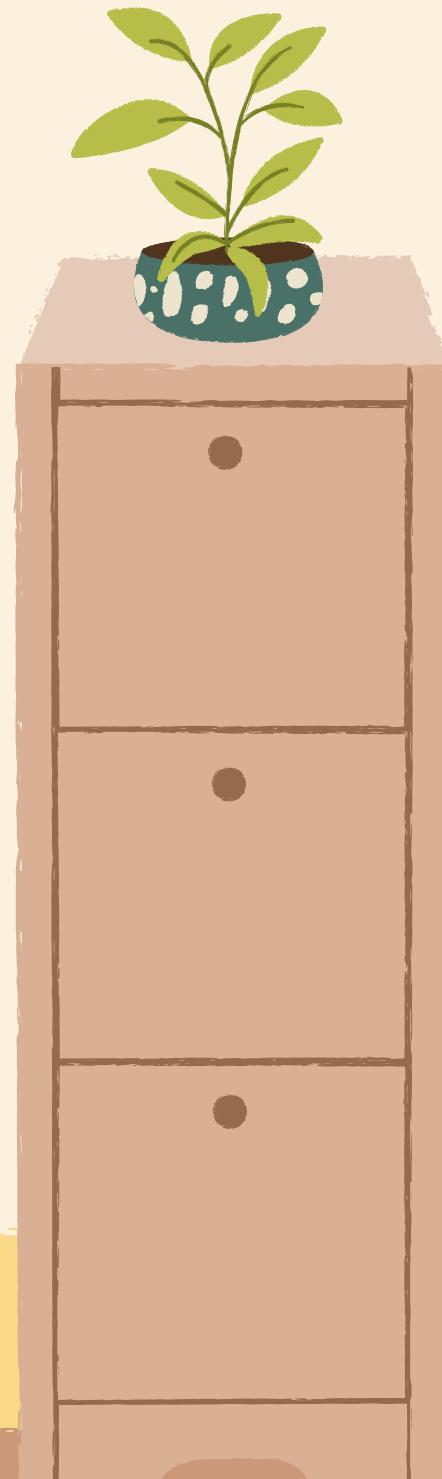
## Usability:

For simple and intuitive user experience we are going to use:

- Angular Material components  
(MatToolbar/Matsidenav )
- Responsive design (Flexbox / Grid)
- Role-based navigation menus

## Example:

- ✓ Student: Courses, QCMs, Clubs
- ✓ Tutor: Course & QCM Management



# ACTORS



ADMINISTRATOR

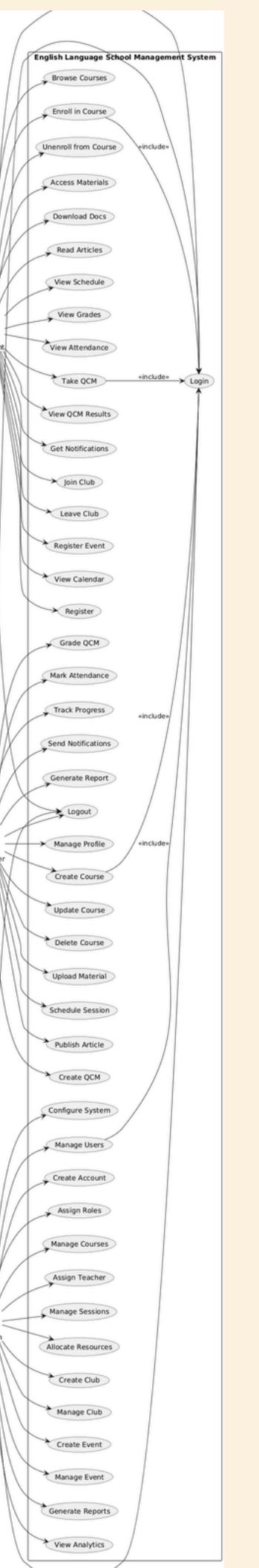


STUDENT

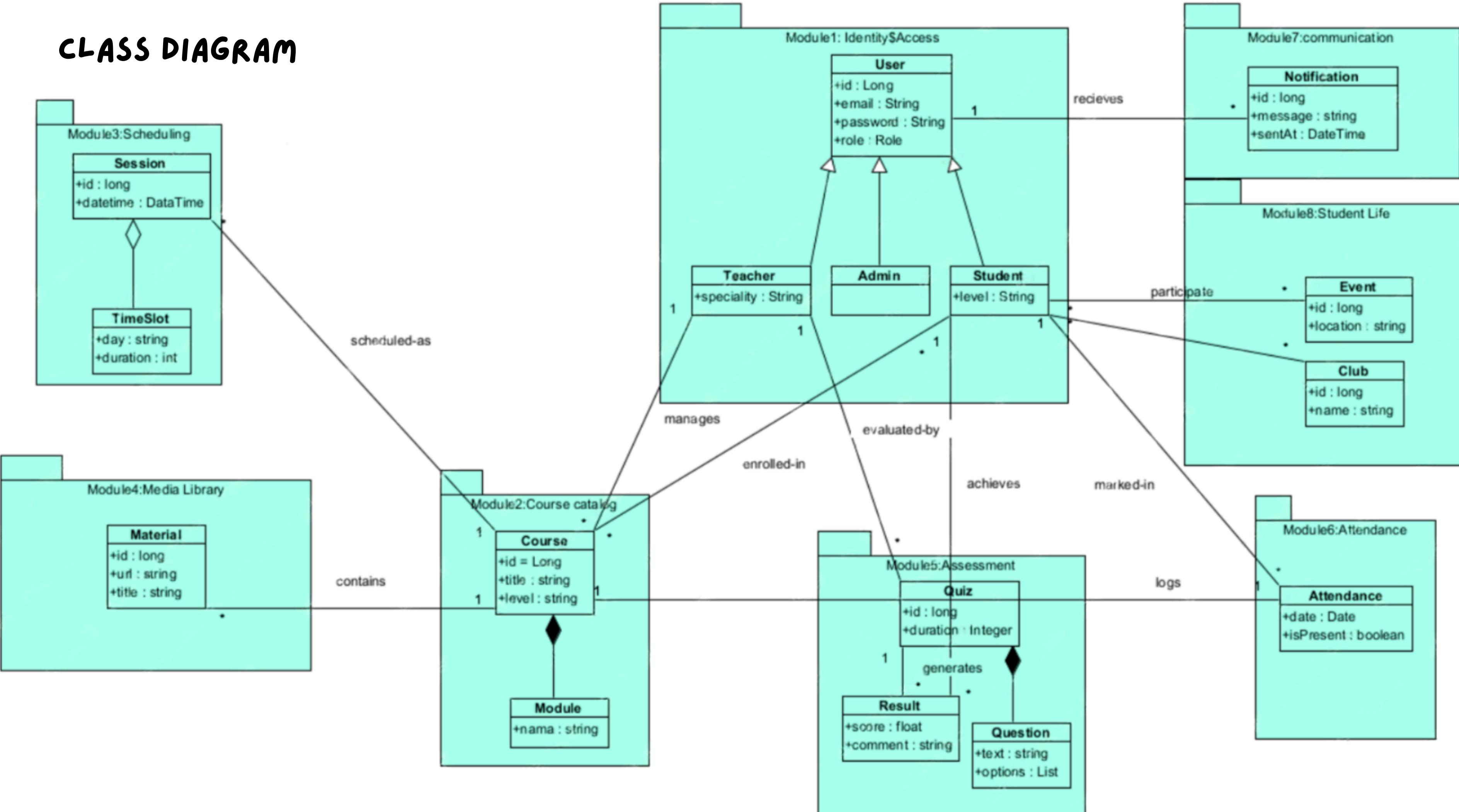


TEACHER

# USE CASE



# CLASS DIAGRAM



# PROTOTYPE



Fluency

Courses

Quizzes

Clubs

Events

Articles



JS

Nidhal Fahem



Beginner

## English for Beginners

Start your English learning journey from the basics

By David Lee

10 weeks 428 students

\$249

[View Course](#)



Advanced

## Advanced Academic Writing

Develop professional academic writing skills

By Dr. Lisa Anderson

8 weeks 156 students

\$349

4.9

[View Course](#)



All Levels

## English Pronunciation Workshop

Perfect your English pronunciation and accent

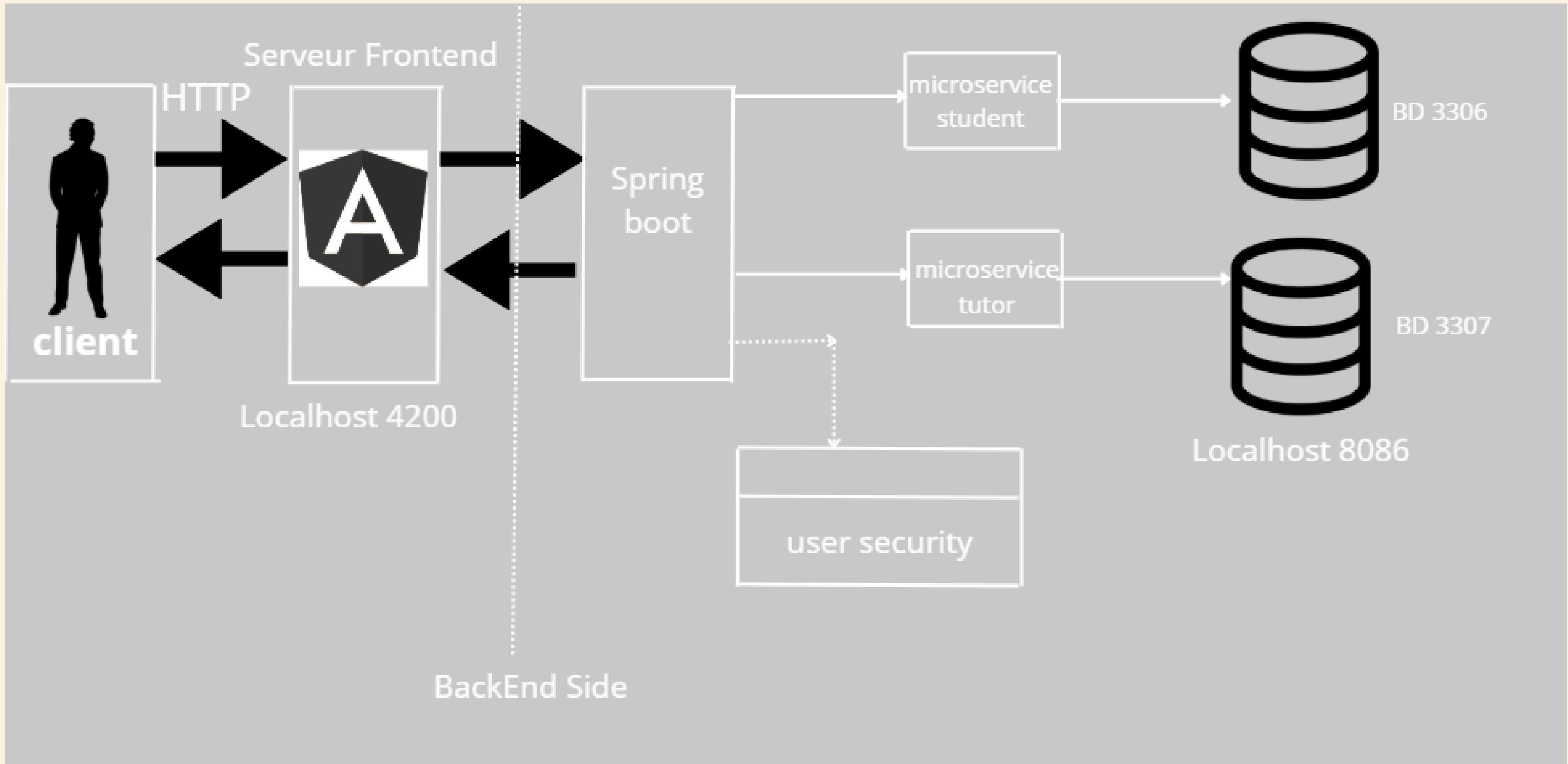
By Rachel Martinez

4 weeks 267 students

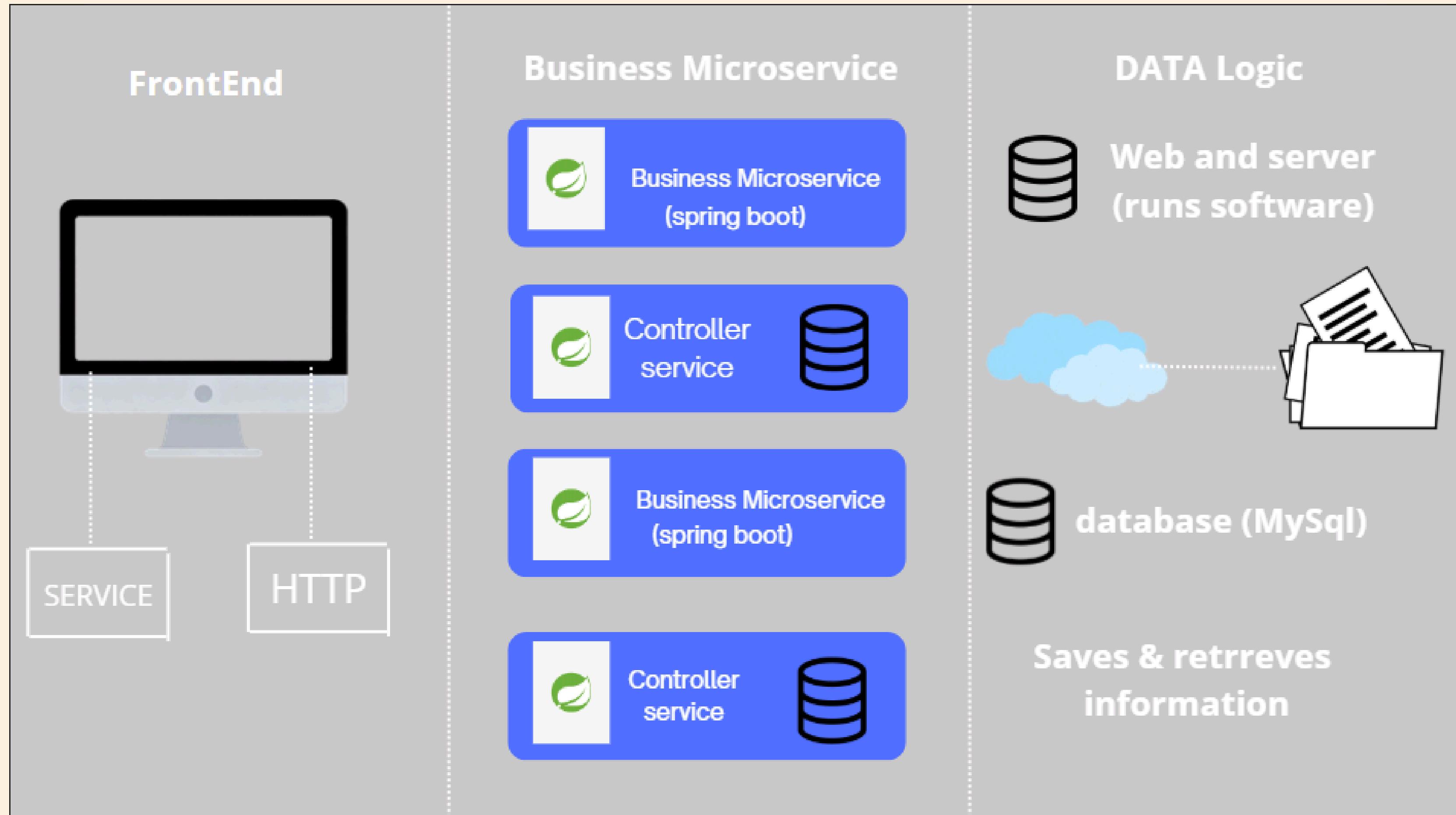
\$149

[View Course](#)

# PHYSICAL LAYER



# LOGICAL LAYER



# CONCLUSION

This analysis phase forms the essential foundation upon which the subsequent phase of detailed design and implementation will be based, in order to translate these needs into concrete and efficient software functionalities.



**THANK YOU FOR YOUR ATTENTION**

