School of Science and Technology

EEET2580 – Enterprise Application Development

Lecturer: Thanh Nguyen Ngoc

Due Date: 18th May, 2022

Member list:

Pham Hoang Thien An:	s3818286
Tran Mach So Han:	s3750789
Nguyen Manh Khang:	s3864131
Nguyen Minh Duy Uyen:	s3819342
Mach Dieu Bang:	s3689744
Le Minh Hoang:	

I declare that in submitting all work for this assessment I have read, understood, and agreed to the content and expectations of the Assessment declaration.

Table of contents:

I. Introduction:

Due to the rapid spread of Coronavirus disease, traditional teaching methods have been pushed back. According to the report published by UNESCO, the early announcement of the suspension of educational institutions affected more than 1.5 billion students and 165 countries all over the world [1]. The potential solution for this is shifting to e-learning or on-air lessons through standard technologies, i.e. (Zoom, Google Meet, Microsoft Team, etc.) to make sure students can continue their education at home. However, there are still some limitations to those technologies. In fact, enterprise services like Zoom, which were created for commercial use and embraced by schools out of pandemic necessity, have less certain future in education [2]. Since the graphics, UI, and features are designed for business purposes, it cannot engage interests from kids or keep them focused during the lecture. Therefore, our project will propose an e-learning application with kid-friendly UI, exclusive features, and assurance for both kids and teacher.

II. Business Requirement:

2.1: Overall goal of the project and our objectives

Our goal for Pengu Classroom is to create a fully functional and user-friendly education website for teachers to manage a group of students and operate online teaching within 2 months. In our website, user's account data, and others important data such as the course's information, student's timetable will be store in a database. To access our website, user can join as the teacher or a student, each of them will have different access to the website's function.

2.2: Website function

2.2.1: Signup, login, sign out

To access the website, the user must have created an account and login using that account. An account will be split into 2 types, a teacher account, or a student account. The user can choose which type of account they want to create in the signup screen. The Login feature contains the JWT and Spring Security Configuration, which will authenticate and authorize user account through username, password and JWT token.

After logging in, the user can sign out using a "Logout" button in the homepage screen, and the JWT token will be removed from the session.

2.2.2: Student account

View course.

After the student logged in, the students can find all of the course that they are currently enrolled in. The list of course will display at the student home screen, if a student clicks on one of the courses, it will move the student to the course page and show many details of the course like the course description.

Homework management

Students can access their homework which is assigned to them by the teacher through their course's dashboard. Furthermore, they can view more details of the homework, and from there, students can attach their homework and forward it to the database.

- **View Timetable**

Students can view their timetable through the home page dashboard. The timetable will contain all of the course's class session time that the student is involved in. Students doesn't have any control over the timetable, they can only view it.

Insert course code to join a course

To join a course, students can insert the code of the course that the student wanted to join. The course code will be provided by the teacher. After the user input the right course code, they will automatically be added into the course by the system.\

Edit profile

Students can edit their user profile, they can change their username, profile picture, description. After updating their profile, the database will update the user's detail into the database.

2.2.3: Teacher.

- **Course management.**

Similar to students, teachers can view the course that they are teaching at the moment, but other than just viewing it, teachers can also add and delete courses that they are teaching. To add a course, user will have to input the course's name and its description. Moreover, teacher can also add a student into a course in the course page.

- **View student list.**

As a teacher, user can ask the website to display the list of students that participate in a course that the teacher teaches. From there, user can remove an unwanted student from the course.

- **View and edit timetable.**

Alike students, teachers have a function to view the timetable through the home page dashboard. But as a teacher, user will have the access to edit the timetable. Teachers can add event on the timetable by clicking on a blank time slot, after that, the teacher will have to input the course id of the course to save it into the database.

· **Add, edit homework of a course.**

As a teacher, the user can manage the course’s homework, creating new homework for a course and assign it to students. When creating new homework, user will need to insert the title, description and submission date of that homework. The new created homework

. **Edit profile**

Similar to student, teacher can edit their user profile, they can change their username, profile picture, description. After updating their profile, the database will update the user’s detail into the database.

III. Use Case Diagram

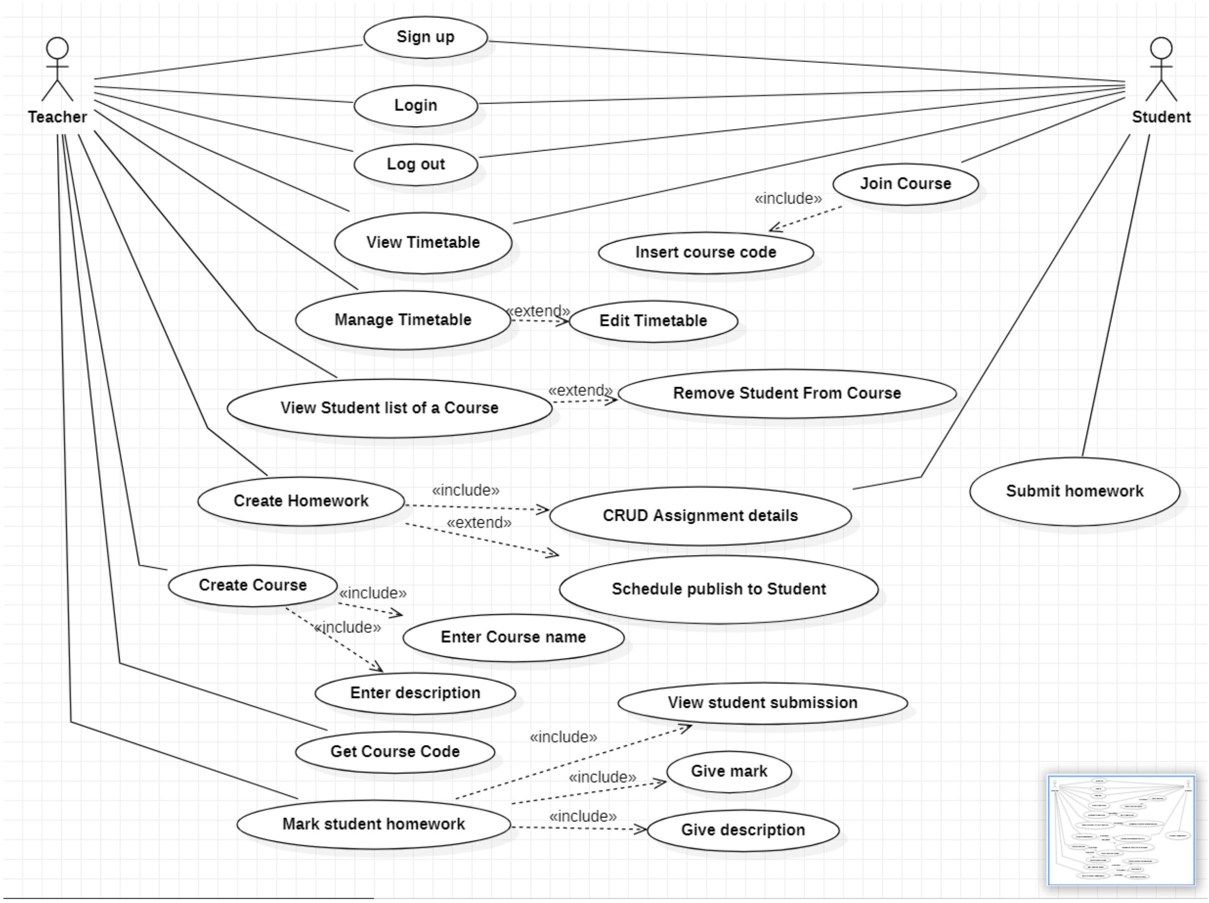


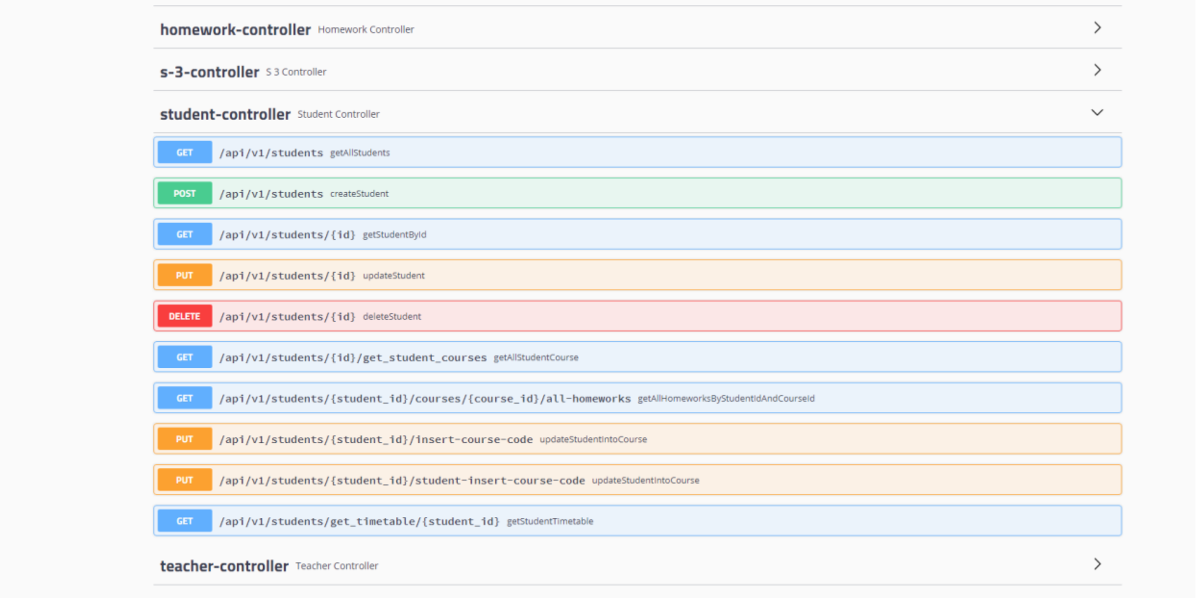
Figure. Use Case Diagram

Our user case diagram contains 2 actors, student and teachers, students and teachers will share some of the basic function, such as login, signup and sign out. Teachers will have other function to manage the courses that they are teaching and others managing function.

IV. Technology

	Technology	Framework	Tool
Frontend	HTML CSS Javascript Python	**Django**:use model-template-view pattern to handle REST api requests. **Bootstrap**: styling the application.	**Figma**: scatter ideas and design a draft UI for application **VSCode:** code editor **Postman:** test REST API request **OpenAPI**: summary API
Backend	Java Postgresql AWS Cloud Storage (S3)	**Spring boot:** open-source framework to build backend using Java Enterprise Edition	**Intellij:** IDE **pgAdmin:** interact with Postgresql database **Postman:** test REST API request **Heroku:** build and configure cloud database for Postgresql

OpenAPI (Swagger): This tool is very useful to organize and demonstrate Restful API to developers. Our team used Swagger to make the work flow between front-end and back-end developers smooth and easy. To be more specific, the picture below displays the swagger UI of our application, all of the API end points, function name, parameter and response of that API are included in each controller in our projects.



homework-controller Homework Controller

s-3-controller S 3 Controller

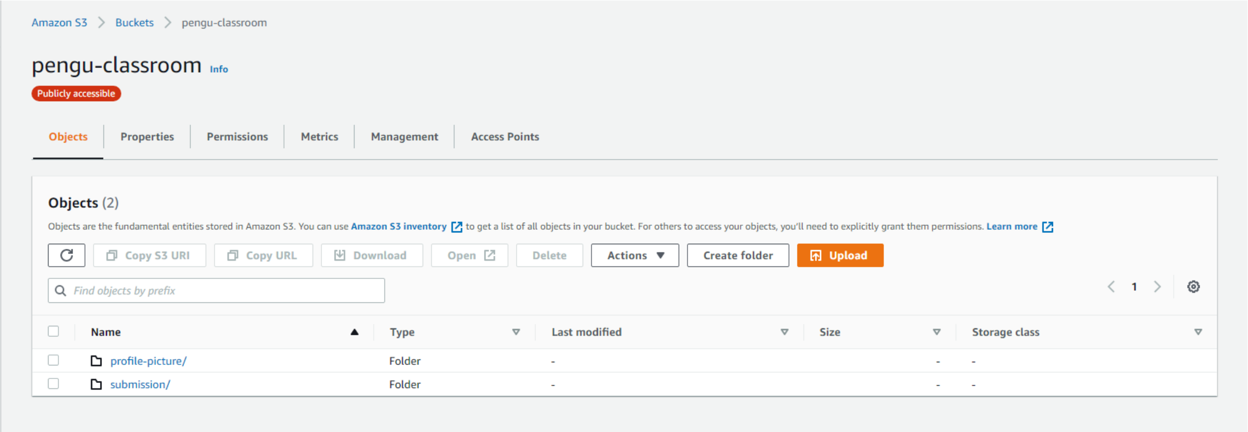
student-controller Student Controller

GET	/api/v1/students	getAllStudents
POST	/api/v1/students	createStudent
GET	/api/v1/students/{id}	getStudentById
PUT	/api/v1/students/{id}	updateStudent
DELETE	/api/v1/students/{id}	deleteStudent
GET	/api/v1/students/{id}/get_student_courses	getAllStudentCourse
GET	/api/v1/students/{student_id}/courses/{course_id}/all-homeworks	getAllHomeworksByStudentIdAndCourseId
PUT	/api/v1/students/{student_id}/insert-course-code	updateStudentIntoCourse
PUT	/api/v1/students/{student_id}/student-insert-course-code	updateStudentIntoCourse
GET	/api/v1/students/get_timetable/{student_id}	getStudentTimetable

teacher-controller Teacher Controller

Figure. Swagger UI

AWS Amazon S3: This is an effective tool for cloud storage. In our application, we will store students ‘submissions and account’s profile picture in S3. In the bucket “pengu_classroom”, there are two folders “profile_picture/” and “submission/”. Currently, all of the access to view, download, and upload to those folders are public to all users.



Amazon S3 > Buckets > pengu-classroom

pengu-classroom Info

Publicly accessible

Objects | Properties | Permissions | Metrics | Management | Access Points

Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Copy S3 URI
Copy URL
Download
Open
Delete
Actions ▼
Create folder
Upload
Find objects by prefix
1

	Name	Type	Last modified	Size	Storage class
□	profile-picture/	Folder	-	-	-
□	submission/	Folder	-	-	-

Figure. Amazon AWS S3 Storage

Version Control System (VCS):

For this project, we decided to use Git and GitHub for the source code management. Since there are two teams in our group, we will create a different GitHub Repo for each team. Each member will finish the own part then push it to each branch then the leader will be responsible of merging the codes to the main branch.

GitHub link: https://github.com/TranMachSoHan/pengu_classroom

To increase the efficiency of the workflow between frontend and backend, our team decide to use Docker which is a very powerful tool to deliver the code in the package called containers. Instead of running Spring Boot project in IntelliJ, frontend Team just need to run the docker file in the container then use it for the implementation and design in VSCode.

V. Architecture

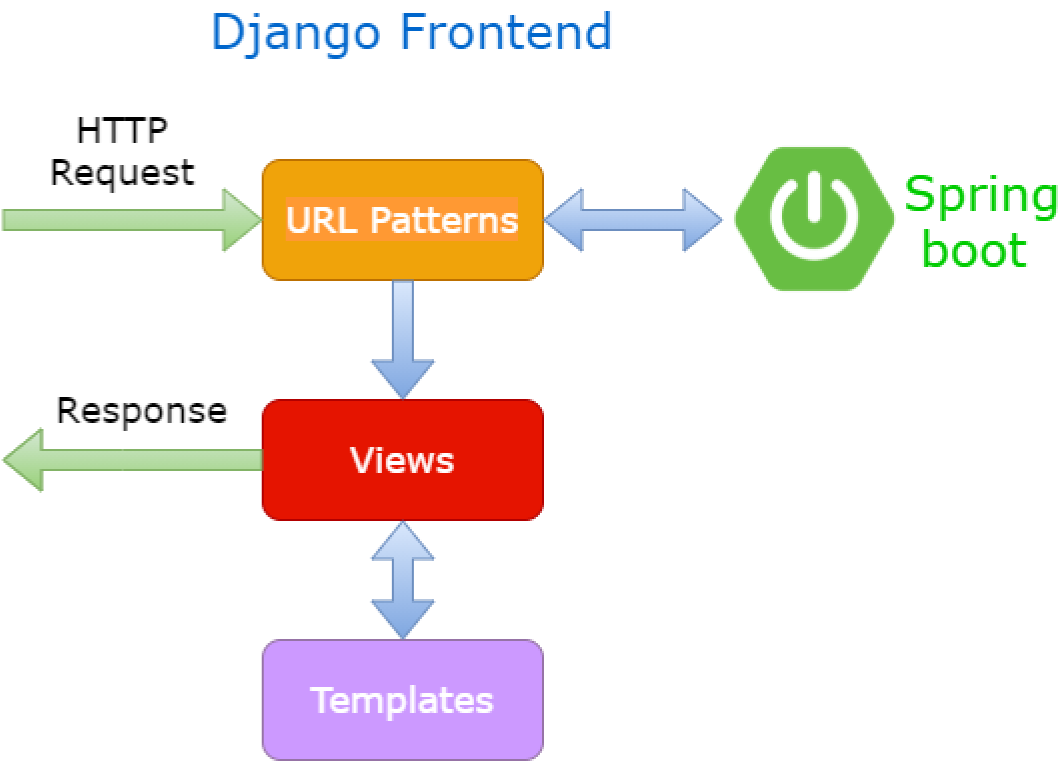


Figure 1. Front-end Architecture using Django

URL Patterns: When users request the http request in browser, if it matches the URL patterns, the request will be sent to Restful API Controller which is the Presentation Layers in Spring Boot backend.

Views: After processing the request in backend, views receive the responses from back end then convert them to HTTP response via a lot of predefined templates.

Since we use Spring boot as a main backend framework, our project will apply Spring boot architecture which consists of four layers.

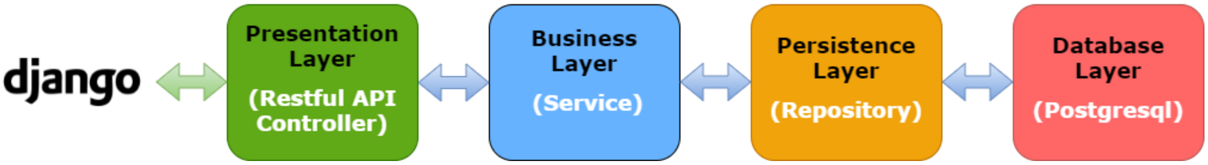


Figure 2. Back-end Architecture using Spring Boot

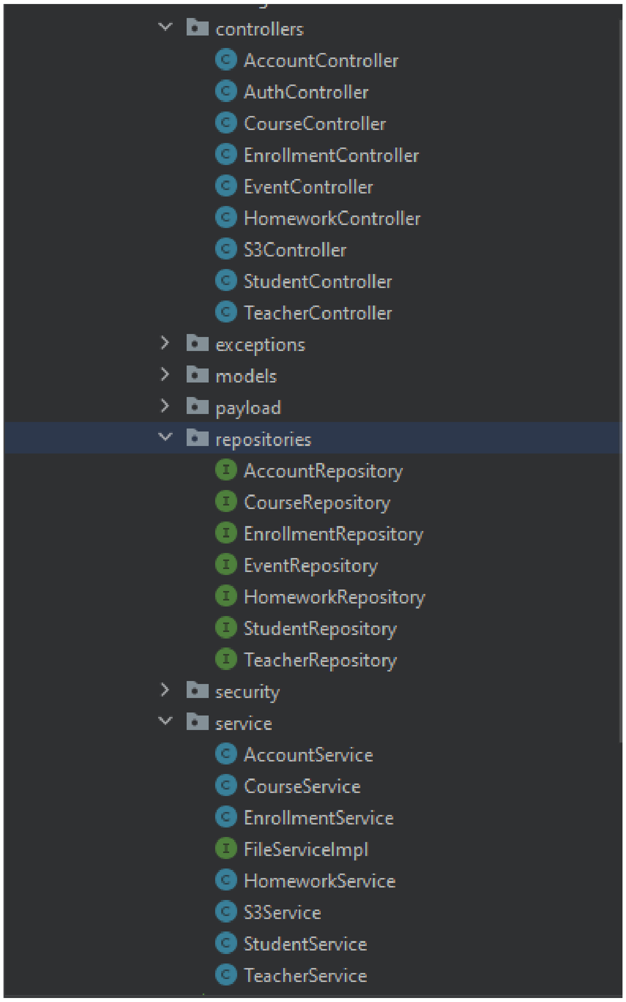
The presentation layer is the connection between the frontend and backend of the application. The objective of this layer is converting JSON data to Java Object and vice-versa, handling HTTP requests, performing authentication and transferring it to the Business Layer the Controller package in our project is relevant to the Presentation Layer.

In the Business Layer, we will deal with all the business logics. In specific, we receive authentication from the Presentation Layer, then perform validation and authorization. All the Service java classes in our project are equivalent to this layer since they execute logical methods. For example, users can edit their information such as password, profile picture, etc.

After handling the logic in the Business Layer, the Persistence Layer will fetch java objects then convert them to a database in the form of rows and columns. Our project makes use of Repository Interface Package in the layer. We can customize some storage logic by writing queries in those interfaces

Finally, Database Layer is where we perform some basic CRUD operations with the database. In this case, we use only Postgresql database, and Heroku as the cloud service to store the database online.

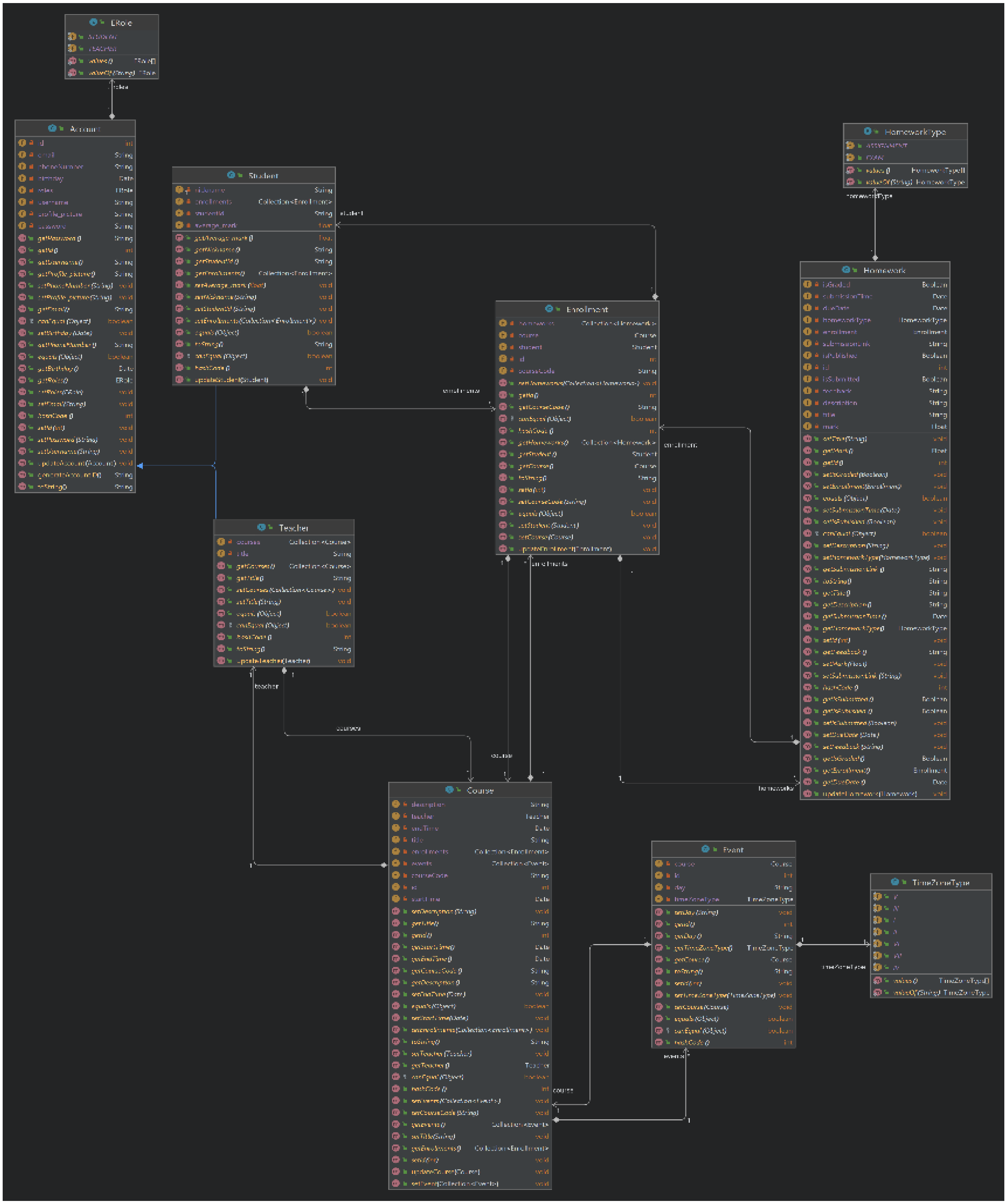
The whole structure will be displayed like this:



```
v controllers
    AccountController
    AuthController
    CourseController
    EnrollmentController
    EventController
    HomeworkController
    S3Controller
    StudentController
    TeacherController
> exceptions
> models
> payload
v repositories
    AccountRepository
    CourseRepository
    EnrollmentRepository
    EventRepository
    HomeworkRepository
    StudentRepository
    TeacherRepository
> security
v service
    AccountService
    CourseService
    EnrollmentService
    FileServiceImpl
    HomeworkService
    S3Service
    StudentService
    TeacherService
```

VI. Class Diagram

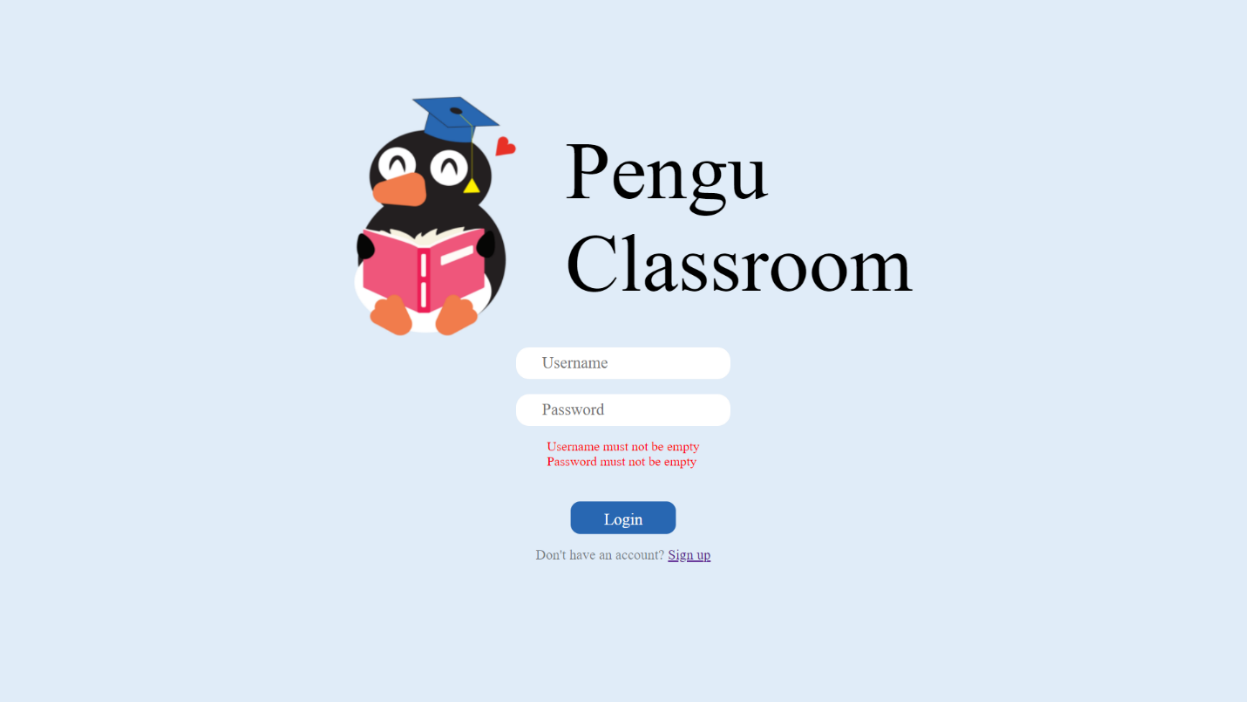
Note: The Class diagram file photo will be included in the submission for better observation



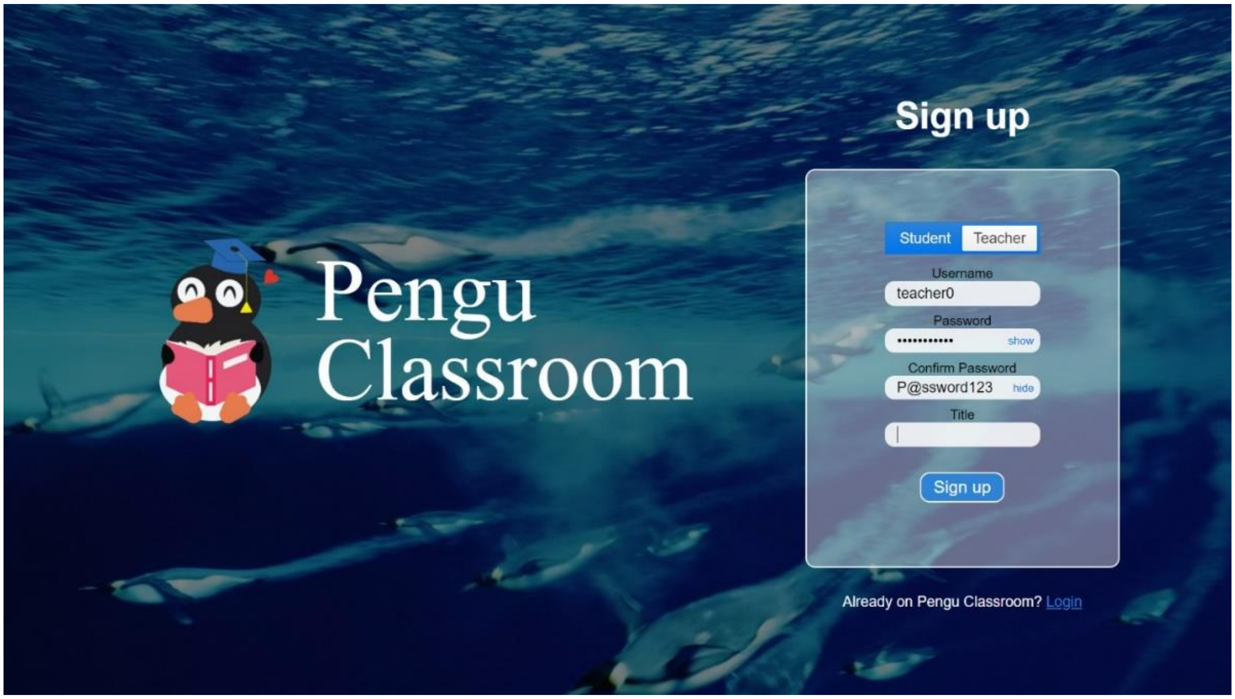
The Class Diagram above show the attributes as well as methods of each model class. Moreover, the relationships among them are also displayed, including the bio-directional and unidirectional ways. Our application comes up with 7 classes: Account, Student and Teacher are children classes of Account, Enrollment, Course, Event and Homework. There are also some enum classes which will be the attribute of some main classes.

VII. Implementation Result

7.1.1 User need to sign up for each teacher or student account before login to the PenguClassroom



Username must not be empty
Password must not be empty



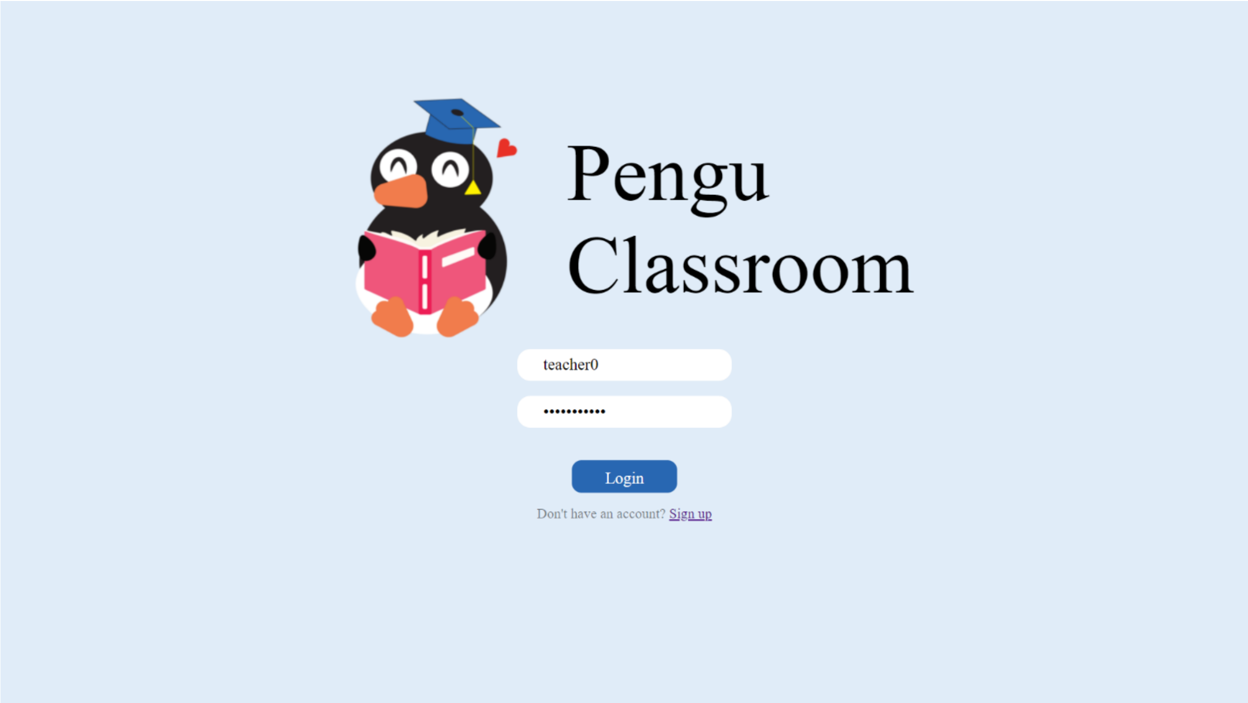
Username
teacher0

Password
•••••••••••

Confirm Password
P@ssword123

Title

Already on Pengu Classroom? Login

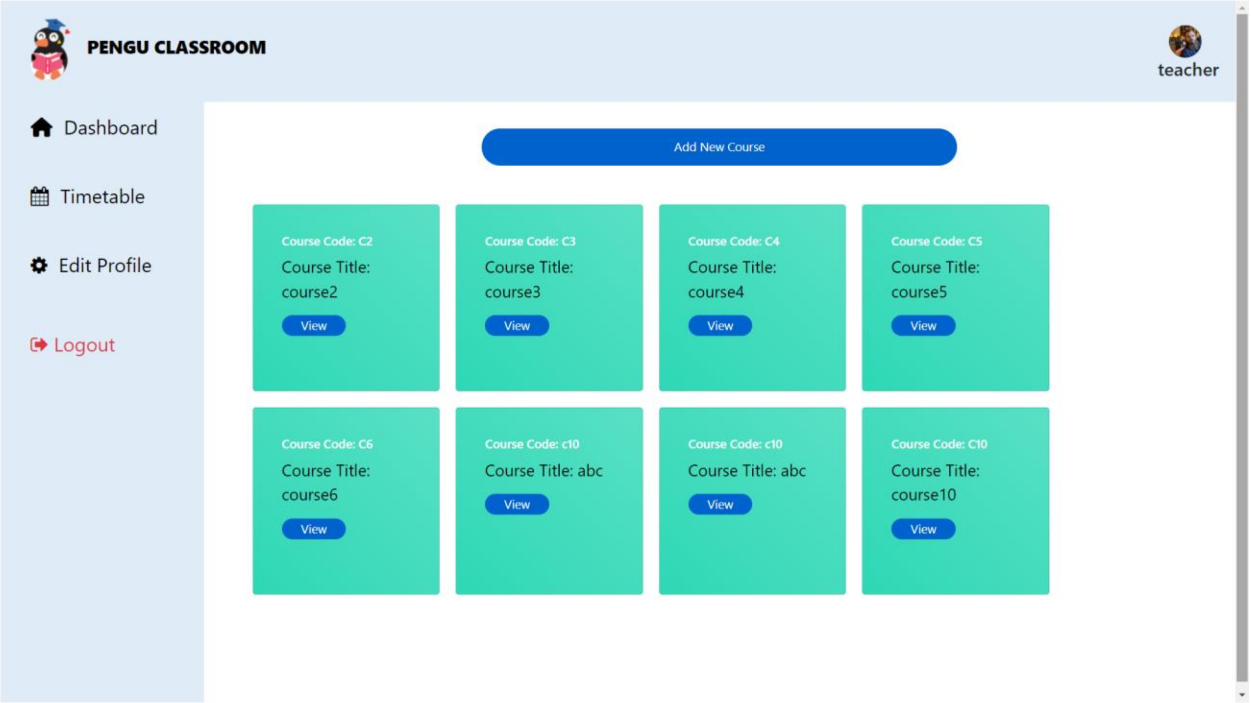


teacher0

•••••••••••

Don't have an account? Sign up

7.1.2 After login user will go to dashboard (list of course) and have a menu bar on the left side of the screen to view timetable and edit profile.



Course Code: C2
Course Title: course2

Course Code: C3
Course Title: course3

Course Code: C4
Course Title: course4

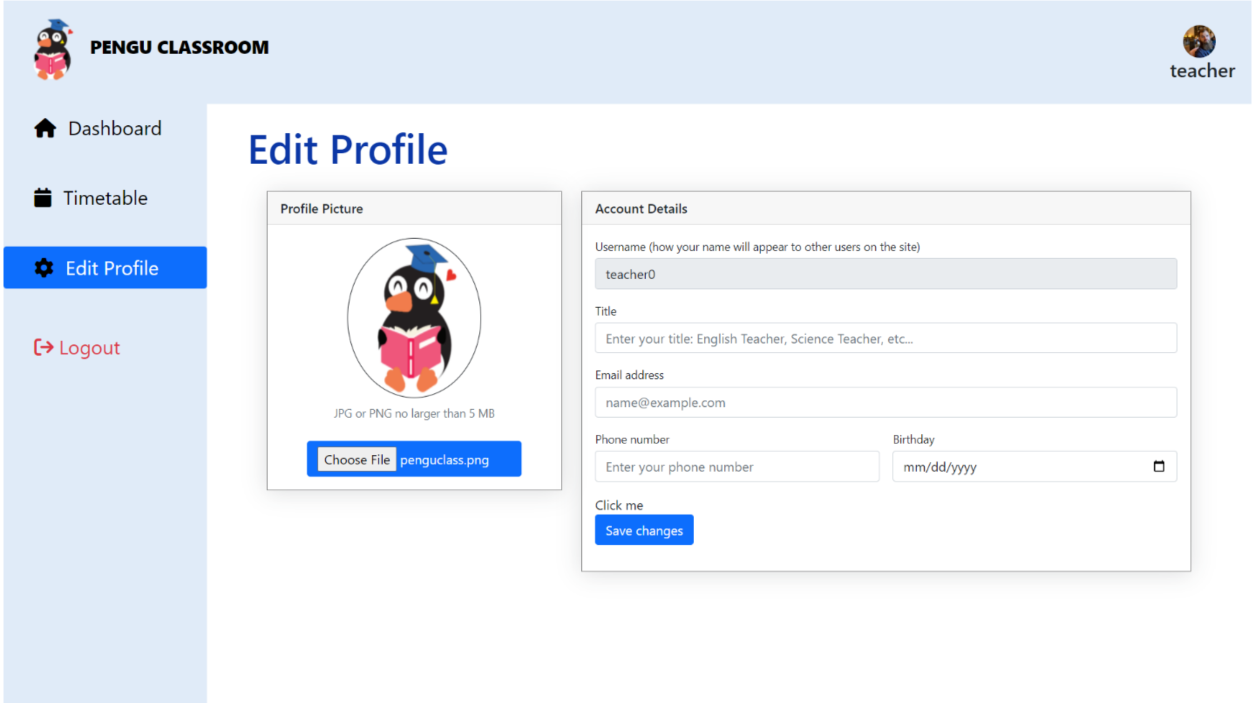
Course Code: C5
Course Title: course5

Course Code: C6
Course Title: course6

Course Code: c10
Course Title: abc

Course Code: c10
Course Title: abc

Course Code: C10
Course Title: course10



Edit Profile

Profile Picture

JPG or PNG no larger than 5 MB

penguclass.png

Account Details

Username (how your name will appear to other users on the site)

teacher0

Title

Enter your title: English Teacher, Science Teacher, etc...

Email address

name@example.com

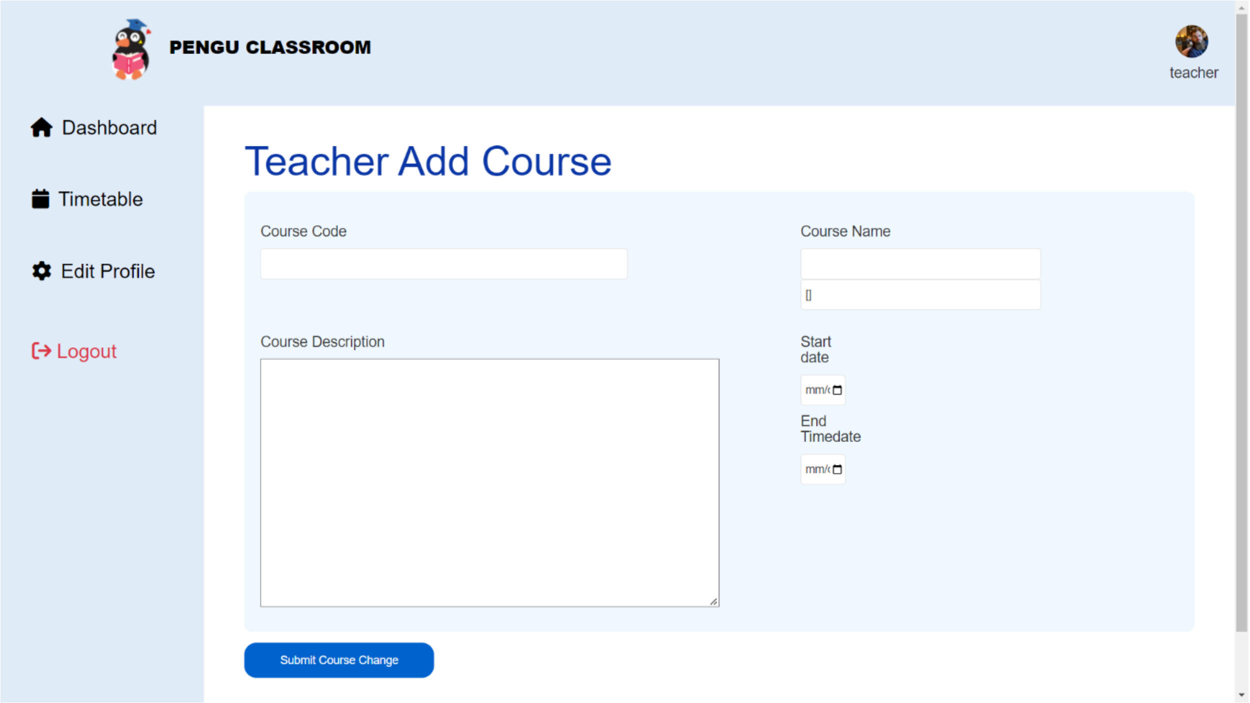
Phone number

Enter your phone number

Birthday

mm/dd/yyyy

7.1.3 In dashboard there is a button to add new course. If you are a student, you cannot see this button



Teacher Add Course

Course Code

Course Name

[]

Course Description

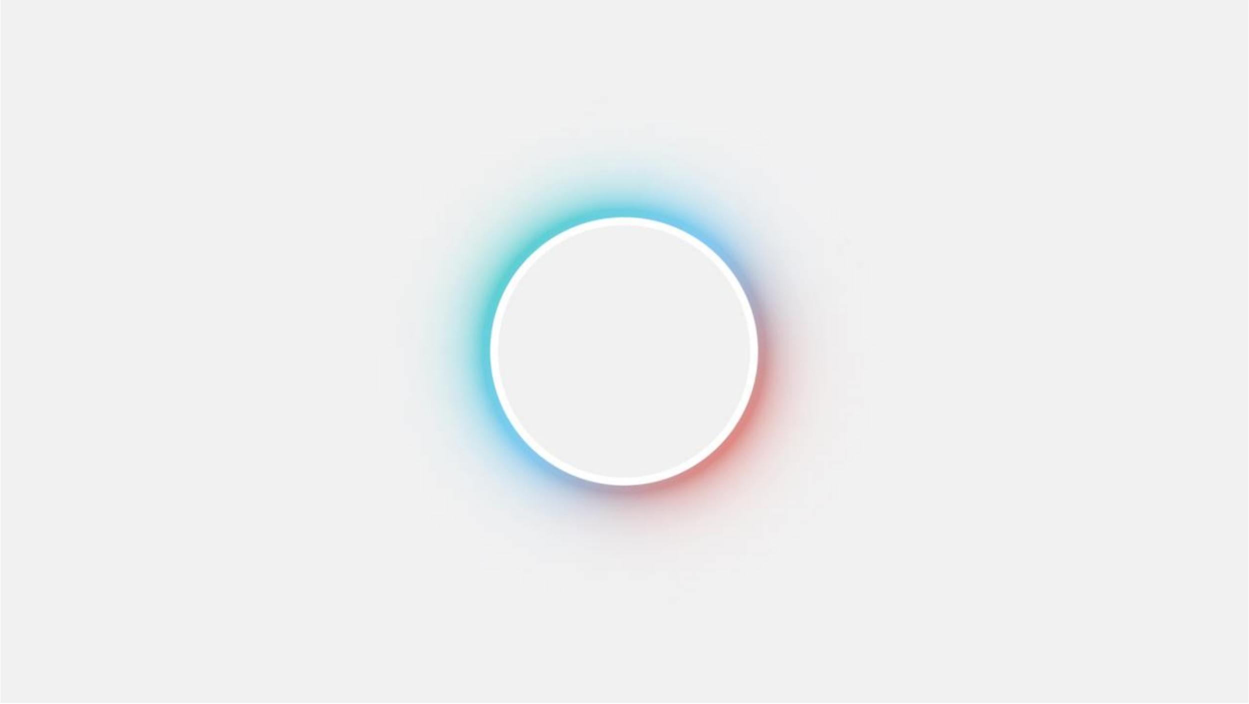
Start date

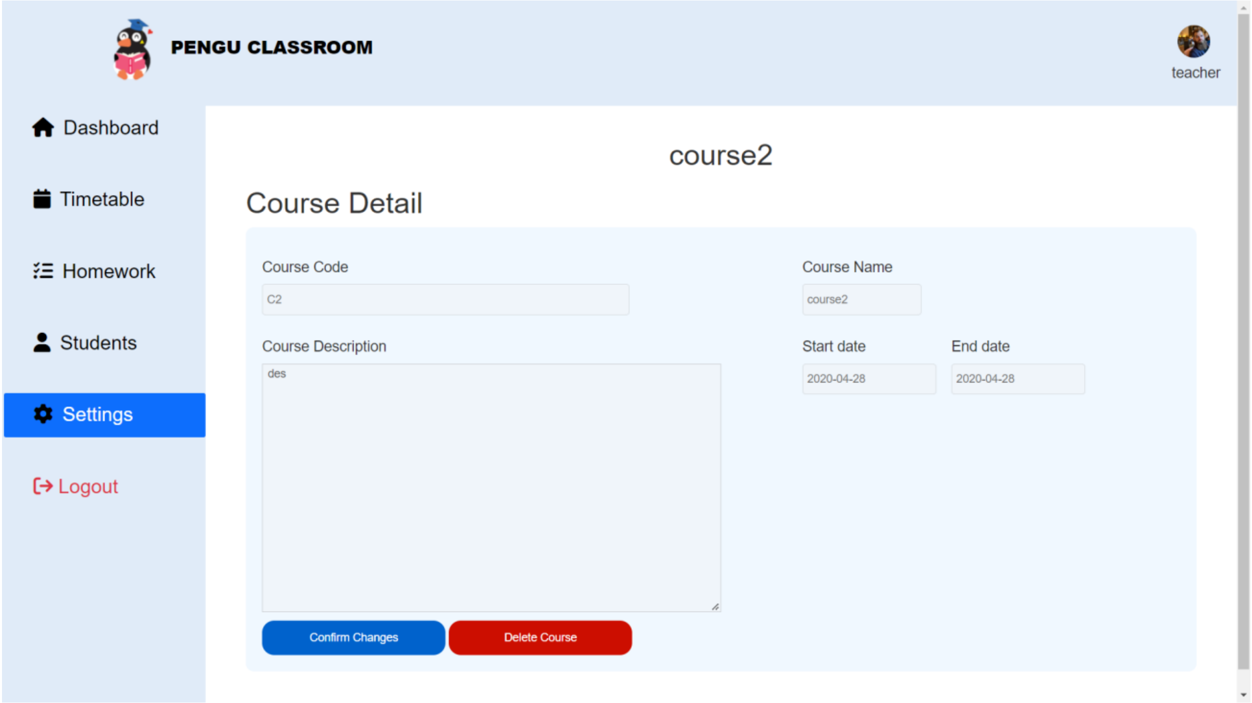
mm/d

End Timedate

mm/d

7.1.4 After click on ‘View’ under the course card, user will be sent to the loading page and then sent to course detail. In here, teacher can edit course detail while student can only view it.





course2

Course Detail

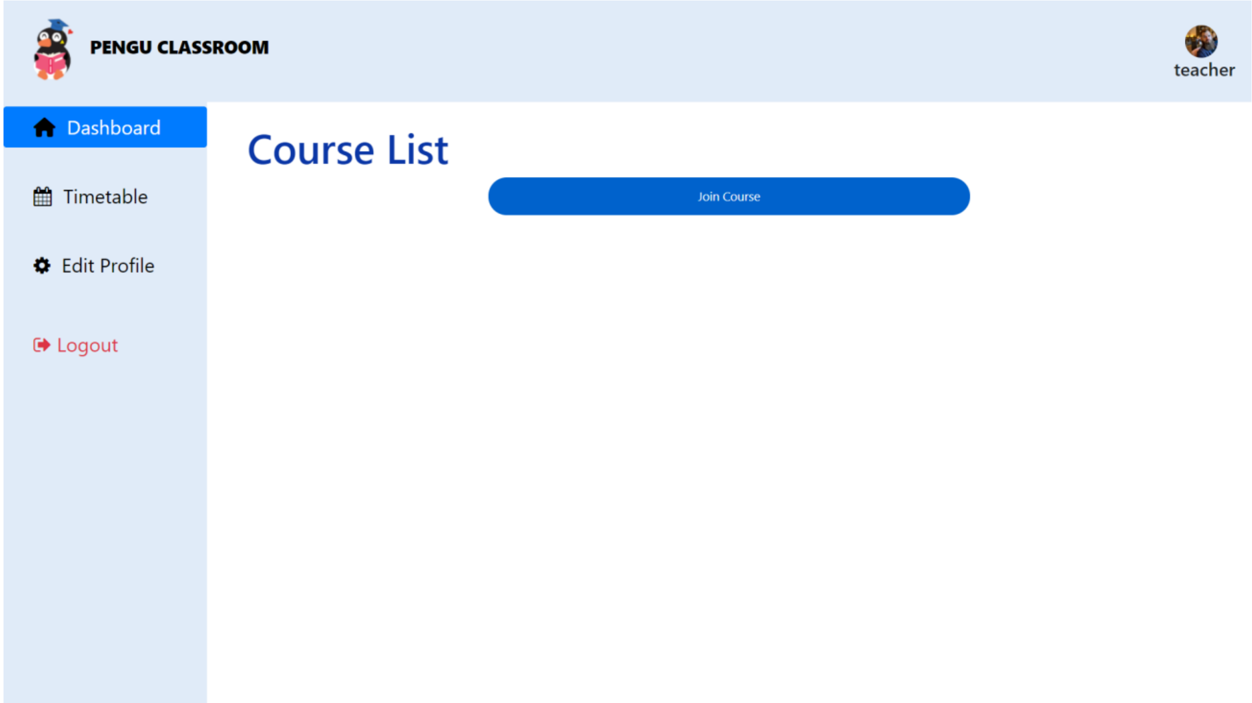
Course Code: C2

Course Name: course2

Course Description: des

Start date: 2020-04-28

End date: 2020-04-28



Course List

VIII. Limitation, Known bugs

8.1: Password recovery unfinished.

In our project proposal, we included a password recovery feature for our application which will let the user change their account's password if they somehow have forgotten their account's password. Due to lack of time, we couldn't implement the function in time, therefore, user should write down their password in case they forgotten it and can't access their account.

8.2: Check Attendance of Student

At first, we plan to make the Attendance feature which will automatically check attendance of students when they join the class through the event. Moreover, teachers can access to those attendances and give mark for student's behavior. Since we don't have enough time for this feature, we will leave it as the future development feature.

8.3: Marks for student's behavior

As mentioned in the previous part, the student's behavior will be marked by teachers based on the attendance report and punctuality of students. Since our group did not finish the attendance feature, we will combine student's behavior feature and attendance together for the future development.

8.4: JWT Token Refresh

In our application, we configure Spring security with JWT with the fixed expiration time of token. After that time, the JWT token will be expired then user cannot log in anymore with that account. That's why out Pengu_classroom app needs the JWT Token refresh features. However, this feature is a little far out of the scope of this course, we will try to research and develop this feature in the future.

IX. References

[1] S. Fiş Erümit, "The distance education process in K–12 schools during the pandemic period: evaluation of implementations in Turkey from the student perspective", Technology, Pedagogy and Education, vol. 30, no. 1, pp. 75-94, 2020. Available: 10.1080/1475939x.2020.1856178 [Accessed 23 March 2022].

[2] N. Singer, "Learning apps have boomed in the pandemic. now comes the real test.," The New York Times, 17-Mar-2021. [Online]. Available: https://www.nytimes.com/2021/03/17/technology/learning-apps-students.html. [Accessed: 24-Mar-2022].