Project Report

Software Engineering

Course: CSE327

Section: 6

Faculty: Sarnali Basak (SLB)

Submission Date: 10th June 2023

Group Information:

Group Members:

Member Name	ID Number
Md. Nafiz Al Mahmud	2012241042
Asif Rahman	2012691042
Mia Mahmudul Hoque Shaon	2014314042
MD Nazmul Islam Srabon	2021227042

Project Information:

Project Name: জীবন যুদ্ধ , an E-Learning Platform

GitHub Link: https://github.com/asifrahman18/cse327Project

Updates Based on Faculty's Suggestions

No.	Faculty Suggestion	Updates	Additional Info	Page No
1	**Add buttons** in place of input boxes	Completed		16, 19, 23, 24, 26, 28, 29. 32, 33, 34
2	Add a **reject button** for admin during teacher verification	Completed	When the admin rejects any verification application, that teacher account gets deleted from the database. However, the teacher will be able to apply again	16
3	Add a **list of enrolled students** for teacher to see	Completed		20, 23, 24
4	Add a **remove student** button for teacher for their courses	Completed		24
5	Add **upload assignment** option for teacher	Completed	Teachers can select the type of material like lecture or assignment before uploading	21
6	Add **view assignment** option for student	Completed	Students can see the type of material form course profile and view assignment from there	29
7	Add user **registration time**	Not Completed		
8	Add student **enrollment time**	Not Completed		

We were suggested to make some updates to our project. The above table contains the list of those changes. We tried to do all the tasks but we couldn't finish a couple of them. In the above table, we have mentioned the updates which we completed along with the ones which we couldn't.

Introduction:

Our project is a web-based E-learning platform called জীবন যুদ্ধ. It is designed to meet the needs of educational institutions, teachers, and students alike. Whether you are a teacher, or a student, our platform offers a range of features to streamline the management and learning process.

There are three types of users: Admin, Teacher, and Student.

For teachers, our website provides tools for creating and organizing courses, and managing assignments and assessments. Teachers can easily track student enrollment.

Students benefit from a user-friendly interface where they can access their course materials and assignments. Our website fosters a dynamic and interactive learning environment, enabling students to engage with course content and enhance their educational experience.

Admin have access to a comprehensive users list, allowing them to monitor the enrollment of students and the creation of courses of teachers.

Vision Statement:

জীবন যুদ্ধ is an online web application for students and teachers who want a simple but effective education platform. Unlike currently available platforms, জীবন যুদ্ধ is a free platform. It will provide a simple, intuitive interface that will also make sure that the teachers are verified and make study much more comfortable.

Features:

Student:

- Registration and login
- Enroll and Drop
- View materials of enrolled course
- Edit profile

Teacher:

- Registration and login
- Upload course material
- View Course list
- View all Student of their respective course
- Remove a particular student from a course
- Edit Profile

Admin:

- Login
- Verify teachers (reject/accept)
- Remove user (teacher, student)
- Remove Course
- Manage the whole system

Major Functional Requirements:

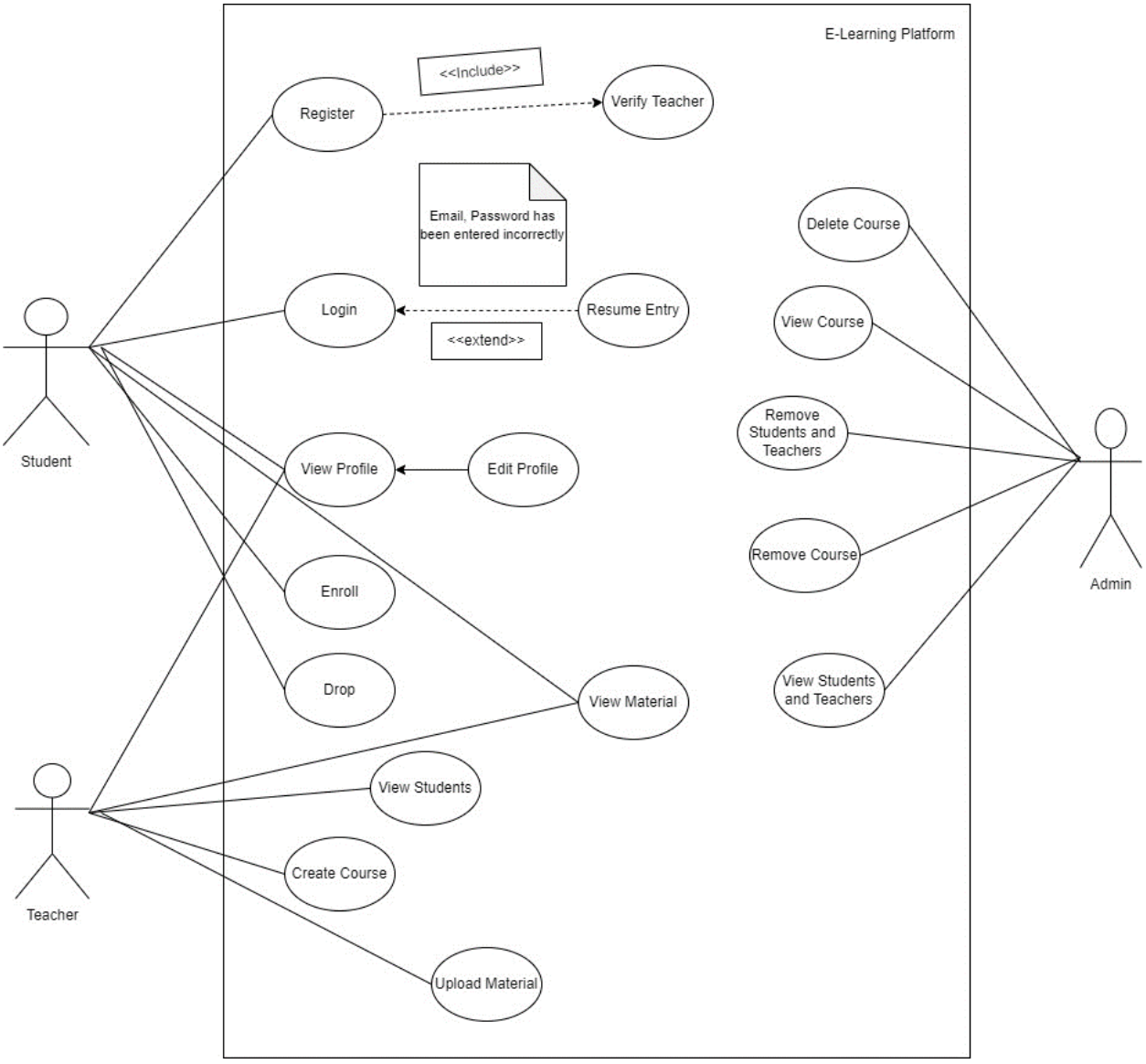
1. Students must register to enroll
2. Teachers must complete the verification process to register
3. The student must be able to search for courses
4. The system must allow the students to see the teacher's profile.
5. Teachers should be able to publish their course materials
6. The system must allow the admin to remove students and teachers account. Also irrelevant courses.

Non-Functional Requirements:

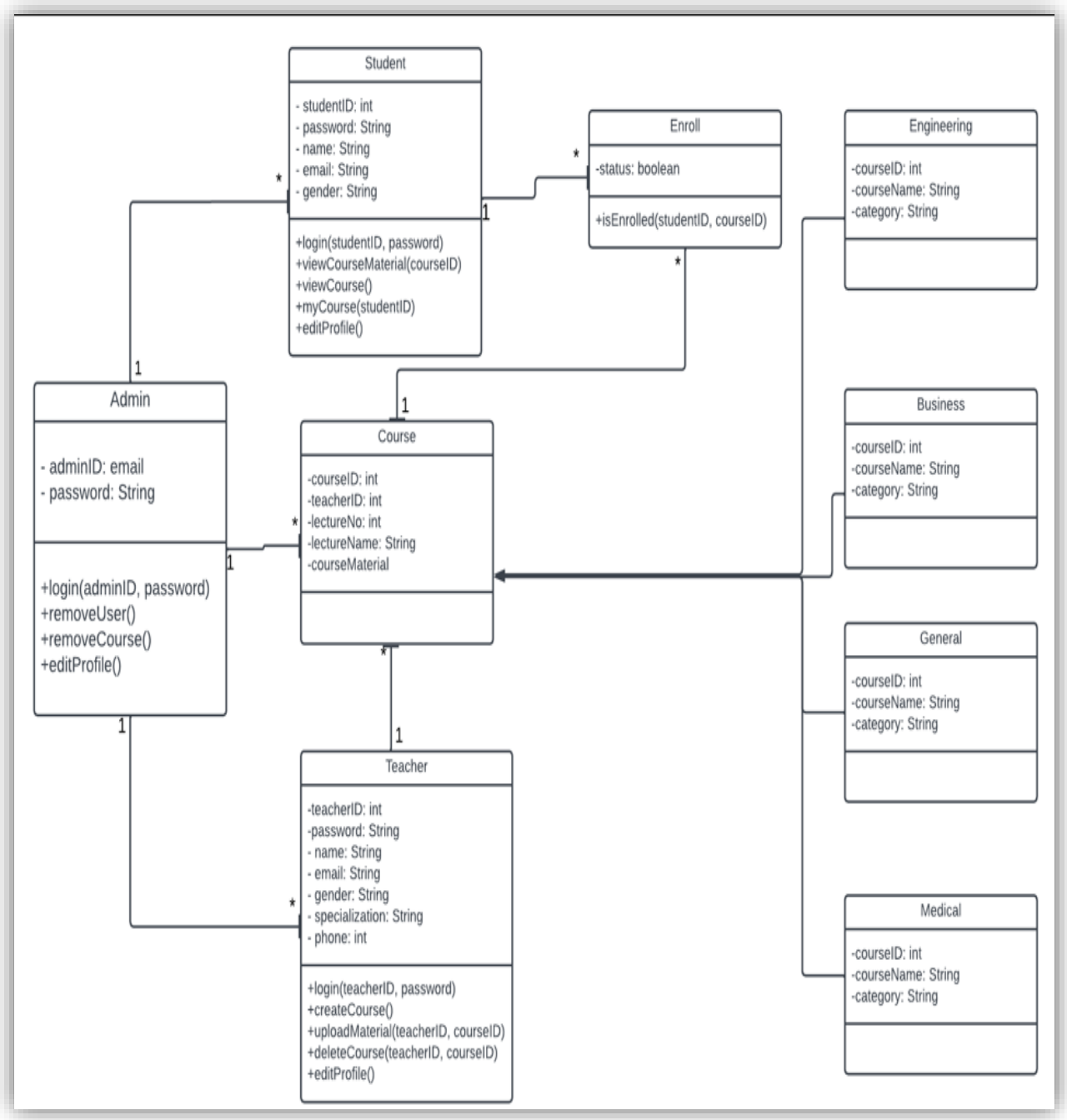
1. The users can request to remember their passwords.
2. Forgetful users can reset their password.
3. The website will keep track of the total number of teachers, students and courses. Also, display the count in the homepage.
4. Students can also search for courses by category from their respective profile page.

Diagrams:

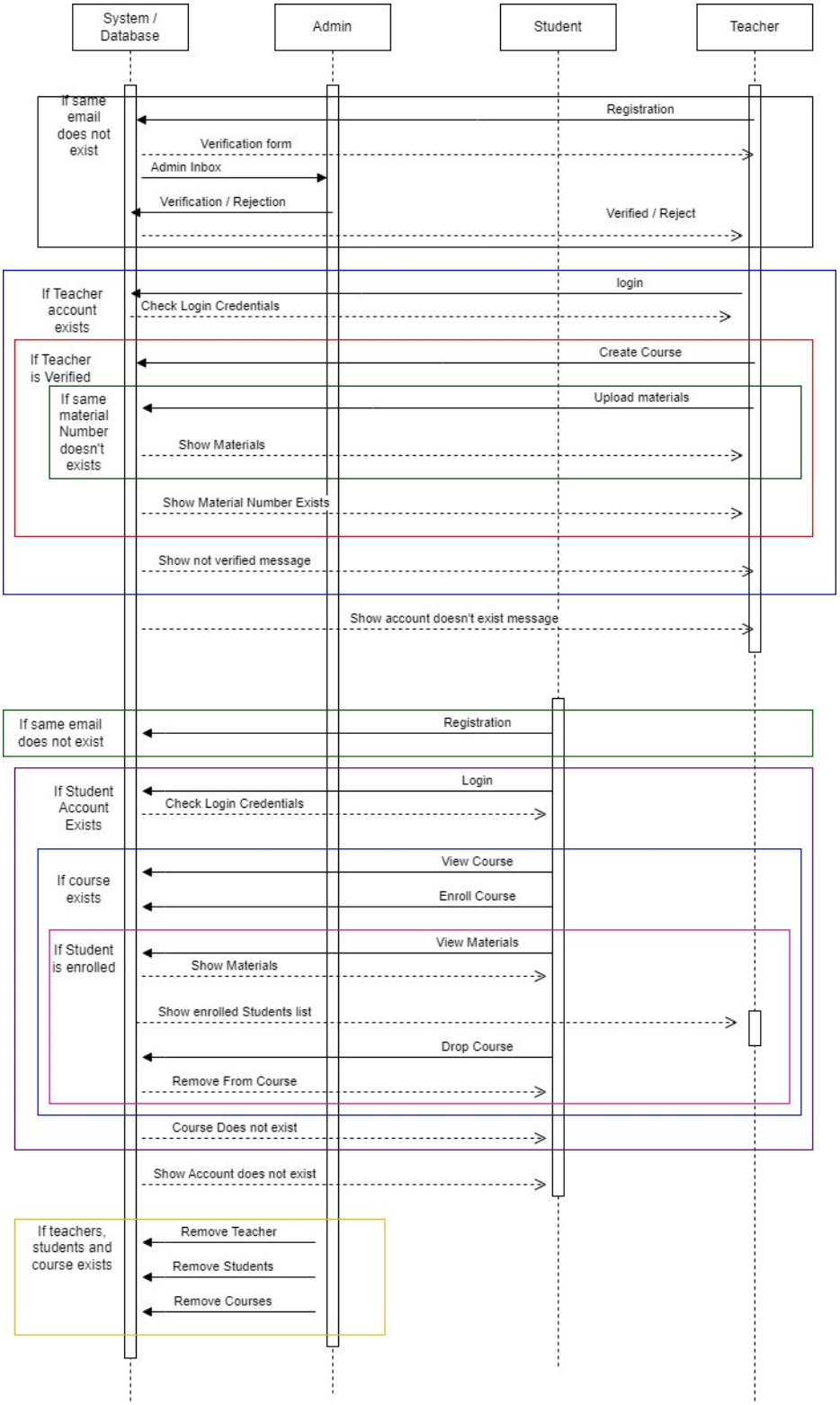
1. Use Case Diagram:



2. UML Diagram:



3. Sequence Diagram:



Use Cases:

Name:	Registration
Description:	Users have to register for the very first time before login
Actors:	Teacher, Student
Type:	Primary
Preconditions:	Null
Main Flow: 1. Go to the website 2. Click the register button 3. The system will take the user to the register page	
Alternative Flows: 1. If the user enters invalid information, an error message will be prompted 2. If there is already an existing user with the same email, a message will be prompted	

Name:	Verification
Description:	Teachers have to be verified before registering
Actors:	Admin, Teacher
Type:	Primary
Preconditions:	Registration
Main Flow: 1. Go to the website 2. Click register as teacher 3. The system will take the user to the verification form	
Alternative Flows: 1. If the user clicks register as student, the system will not show the verification form	

Name:	Login
Description:	Users can login into the system
Actors:	Student, Teacher, Admin
Type:	Primary
Preconditions:	Registration
Main Flow: 1. Go to the website 2. Click on Login 3. Enter login information	
Alternative Flows: 1. If the from is not filled up properly, an error message will be shown	

Name:	Edit profile
Description:	Users can edit their profile to update information
Actors:	Admin, Teacher, Student
Type:	Secondary
Preconditions:	Login
Main Flow: [Login use case] 3. Go to profile 4. Click edit profile 5. Fill up the form 6. Click update	
Alternative Flows: 1. If the form contains invalid information, error message will be shown	

Name:	View Course
Description:	Students can see the list of offered course lists
Actors:	Student
Type:	Primary
Preconditions:	Login
Main Flow: [Login use case] 4. Click on Courses button 5. The system will display the list of courses to the user	
Alternative Flows: 1. The user can also see courses by categories from their profile	

Name:	Enroll
Description:	Students can enroll into their desired courses form the offered course list
Actors:	Student
Type:	Primary
Preconditions:	Login
Main Flow: [View course use case] 6. Enter course ID 7. Click on enroll	
Alternative Flows: 1. The system will also show details of the course	

Name:	Drop course
Description:	Students can drop any enrolled course
Actors:	Students
Type:	Secondary
Preconditions:	Enroll
Main Flow: [Enroll use case] 8. Click on my course 9. Enter course ID of the course 10. Click on drop button	
Alternative Flows: 1. Student have to enroll before dropping any course	

Name:	Create Course
Description:	Teachers can create their course
Actors:	Teacher
Type:	Primary
Preconditions:	Login, Verification
Main Flow: [Login use case] 4. Click on create course 5. Fill up the form with course information	
Alternative Flows: 1. Teachers cannot create a course before they are verified	

Name:	Upload Course Materials
Description:	Teachers can upload their course materials
Actors:	Teacher
Type:	Secondary
Preconditions:	Create course
Main Flow: [Create course use case] 6. Go to the course 7. Click on upload materials	
Alternative Flows: 1. A teacher has to create the course first before uploading any content	

Name:	Delete Course
Description:	Teachers can delete their course
Actors:	Teacher
Type:	Secondary
Preconditions:	Create course
Main Flow: 1. Go to the course 2. Enter course code of the course 3. Click on delete course	
Alternative Flows: 1. Deleting the course will also delete the contents	

Name:	View Users
Description:	Admin can see the list of all teachers, students and course
Actors:	Admin
Type:	Primary
Preconditions:	Login
Main Flow: [Login use case] 4. Go to profile 5. Select Students/Teachers/Course	
Alternative Flows: 1. If the website has no teacher/student/course, the list will be empty	

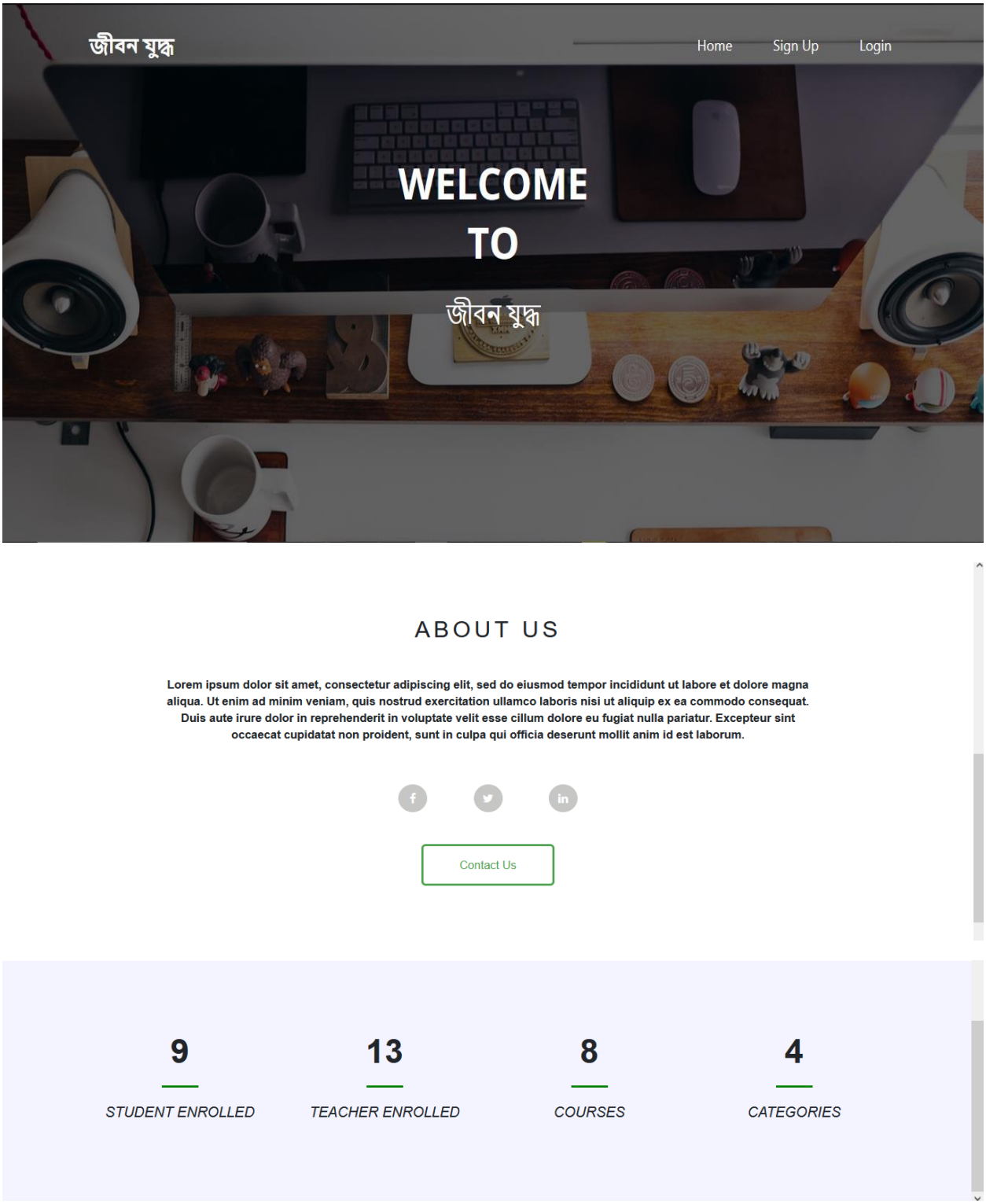
Name:	Remove user
Description:	Admin can remove any unwanted user
Actors:	Admin
Type:	Primary
Preconditions:	Login
Main Flow: [View Users use case] 6. Click on teacher/students 7. Enter teacher ID/Student ID 8. Click on remove user	
Alternative Flows: 1. If the site has no teacher/student, the list will be empty	

Implementation:

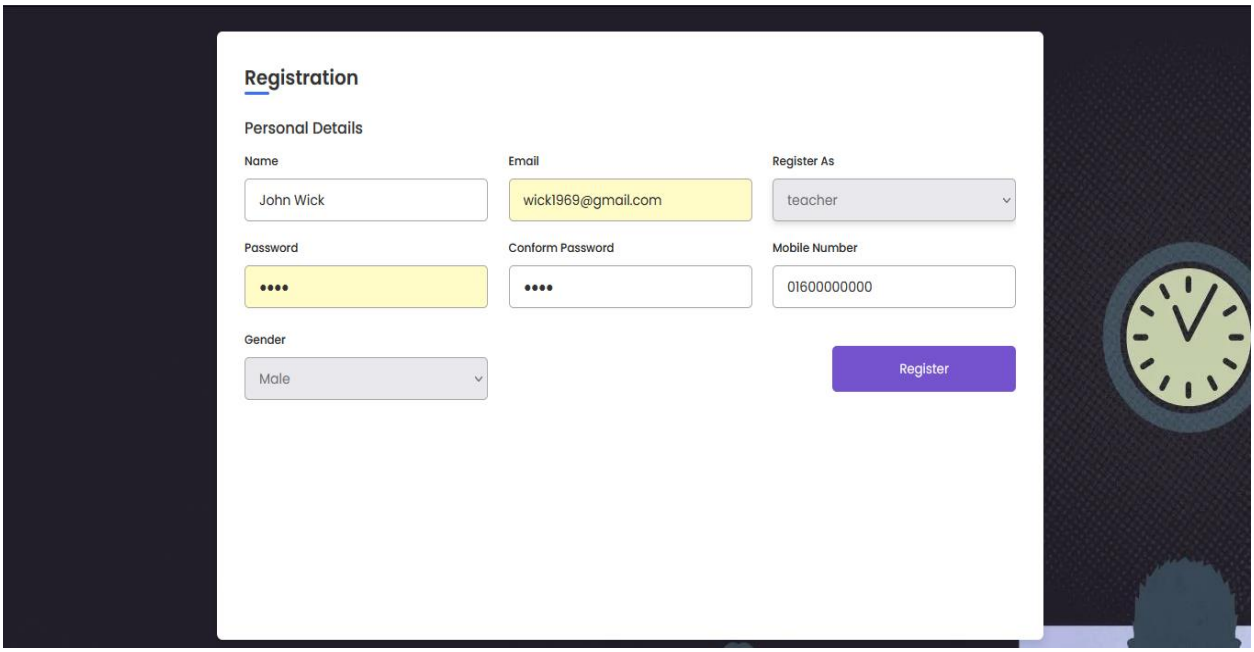
1. Database

	Table	Action	Rows	Type	Collation	Size	Overhead
☐	admin	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	business	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	course	Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	engineering	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	enrolls	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	general	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	medical	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	pdf_file	Browse Structure Search Insert Empty Drop	9	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	student	Browse Structure Search Insert Empty Drop	9	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	teacher	Browse Structure Search Insert Empty Drop	14	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	verification	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 KiB	-
	11 tables	Sum	50	InnoDB	utf8mb4_general_ci	176.0 KiB	0 B

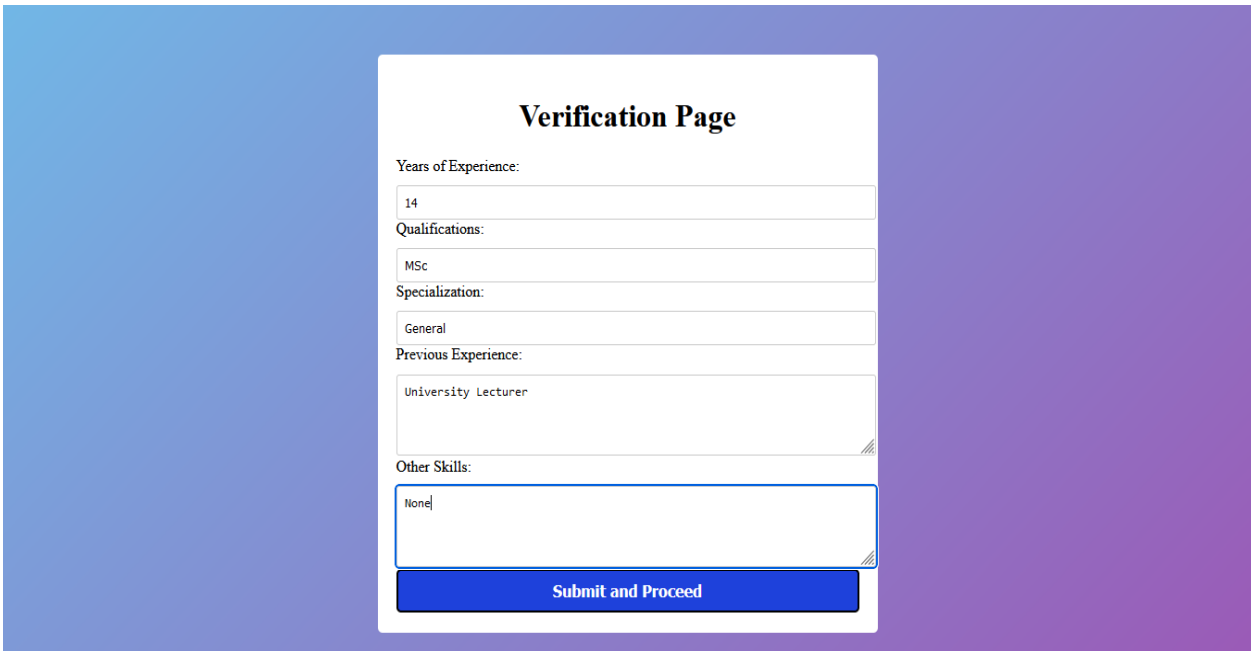
2. Interface



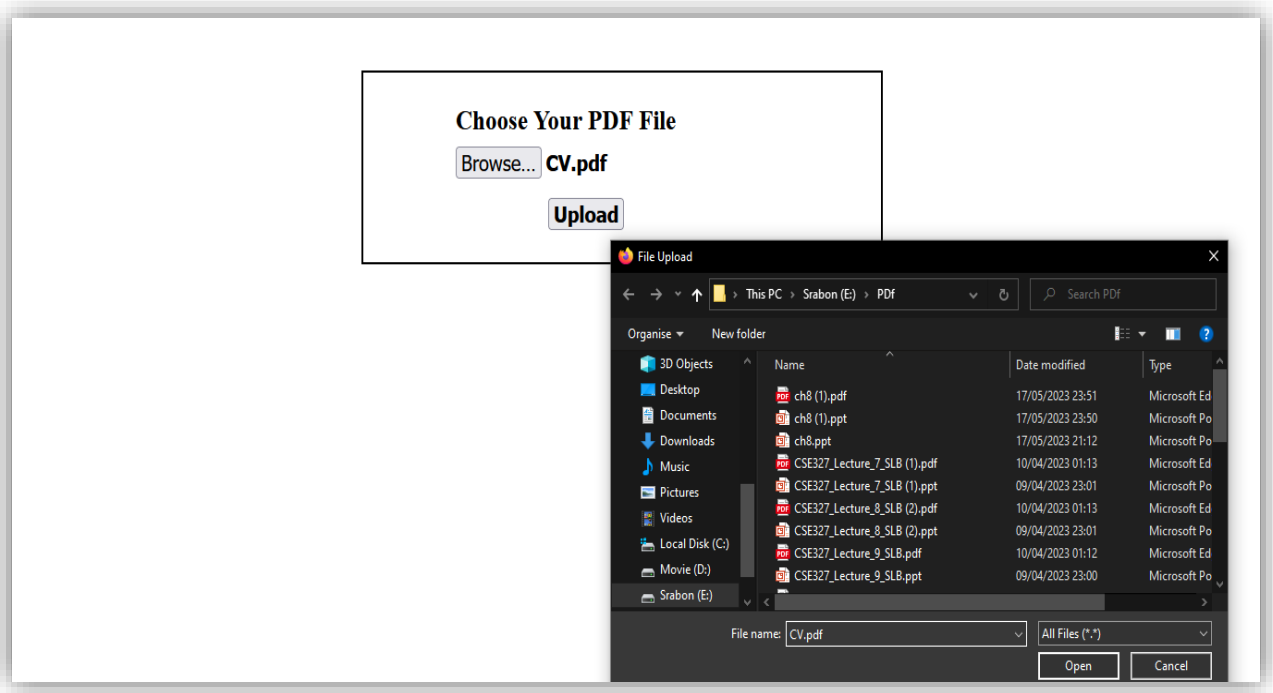
Home Page



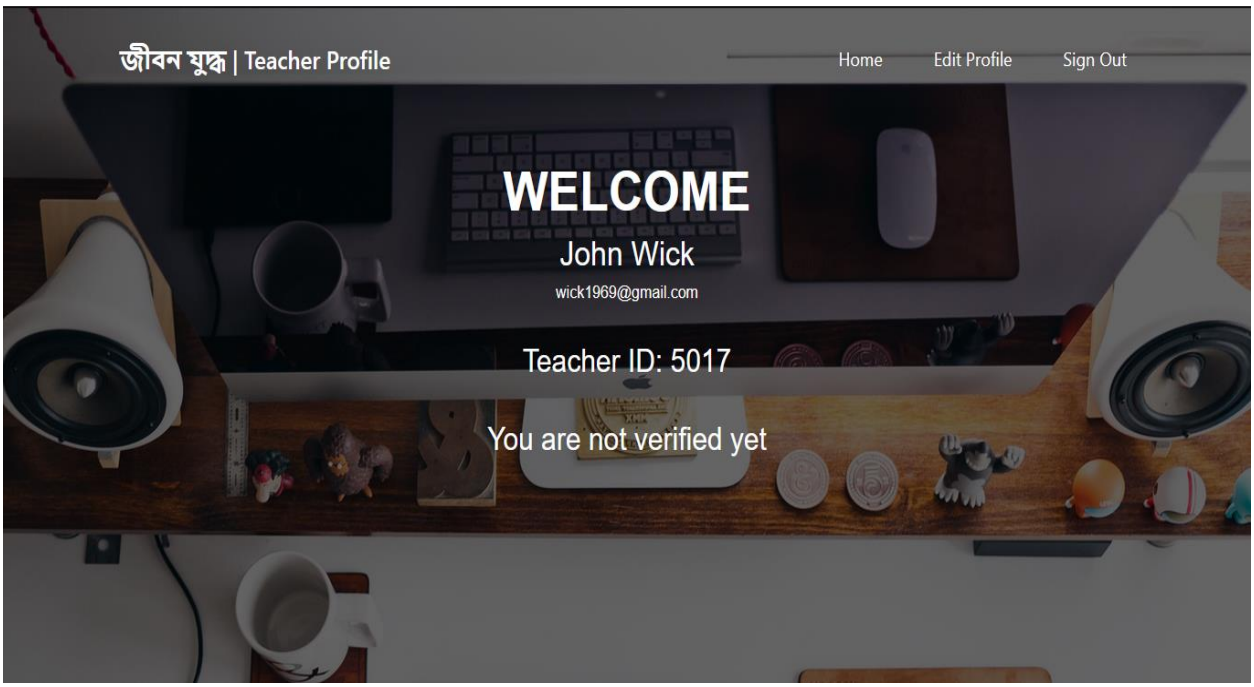
Registration Page (Teacher)



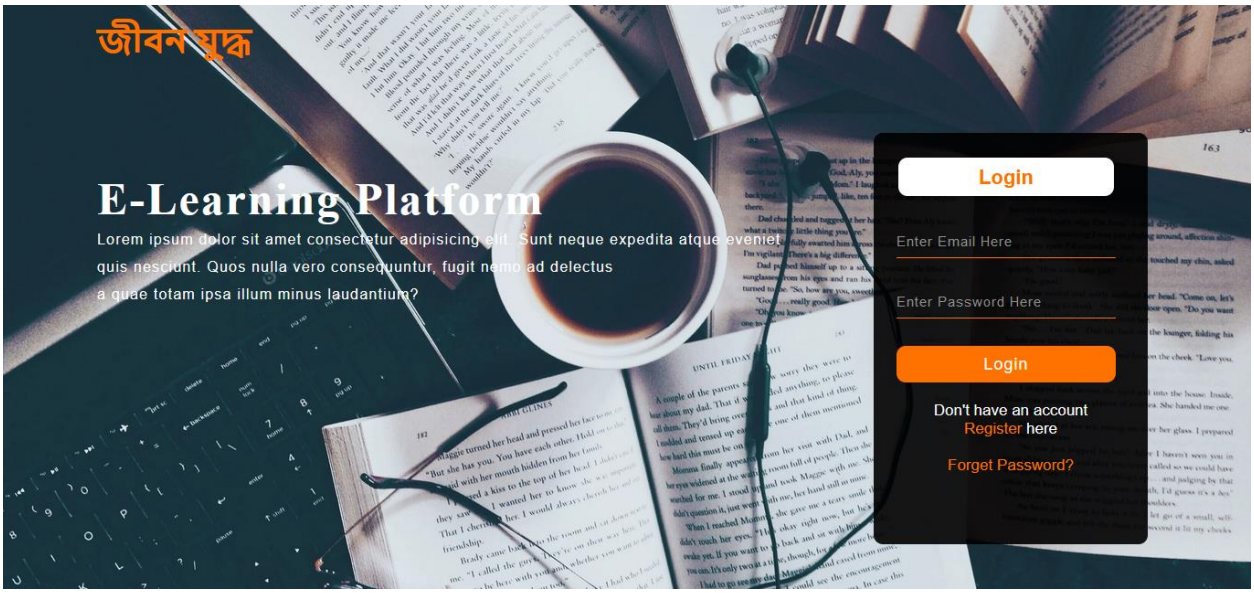
Teacher's Verification Page



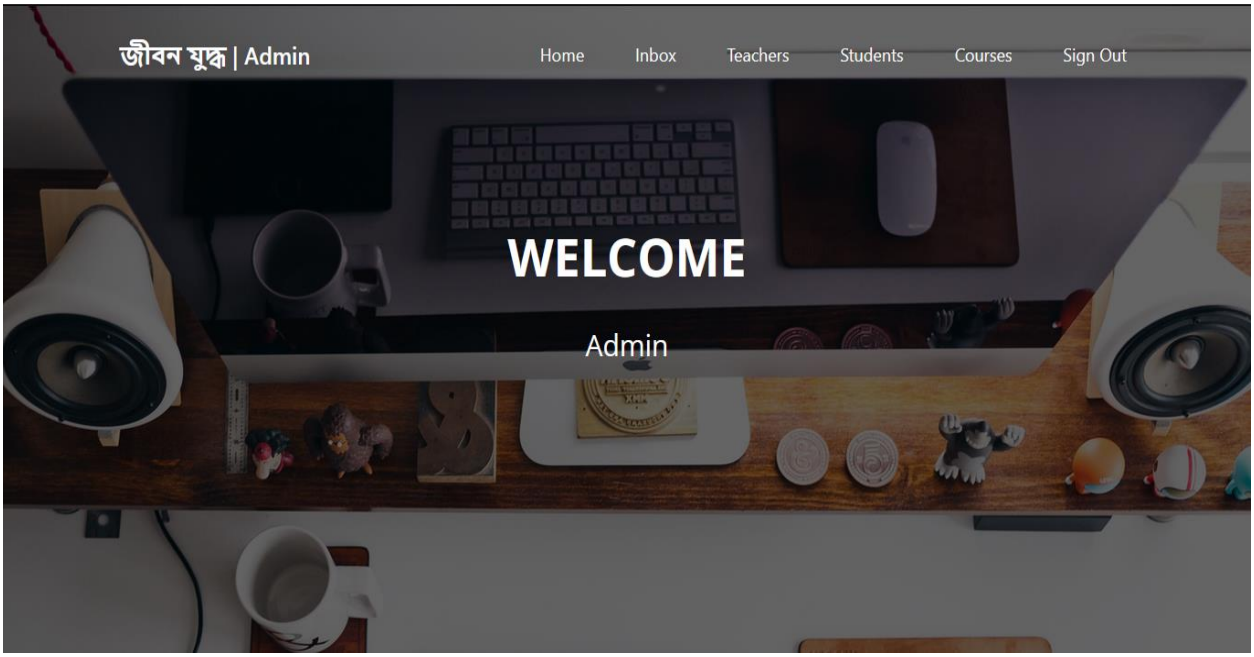
Upload Teacher's Verification Document



Teacher's Profile (pre-verification)

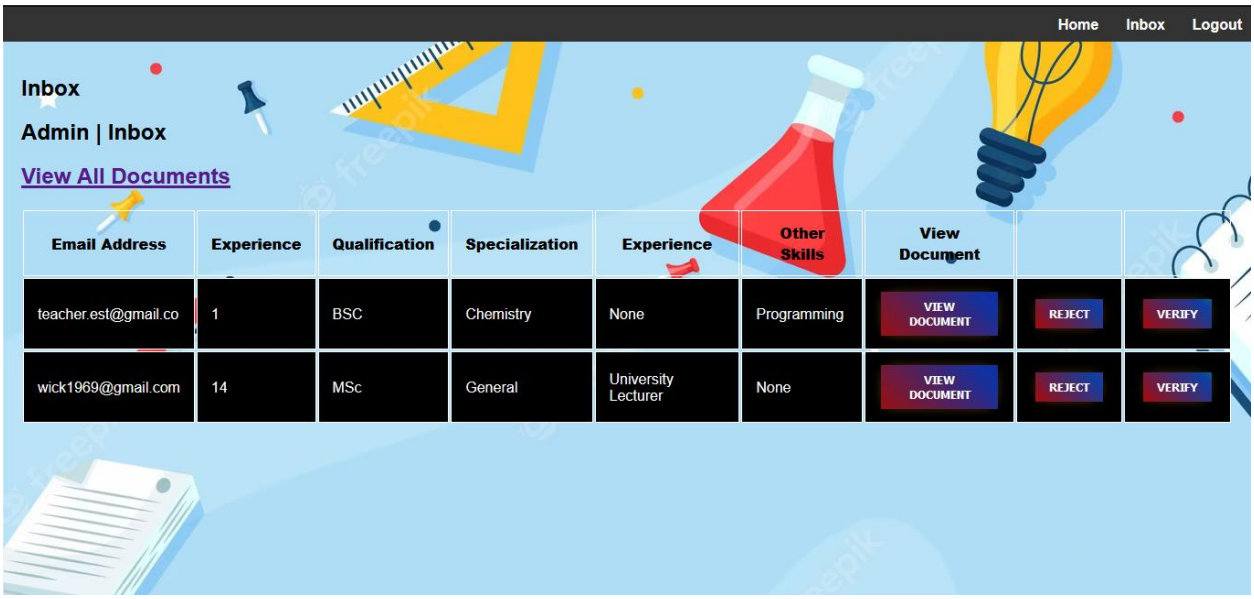


Login Page, for all users (Student, Teacher & Admin)



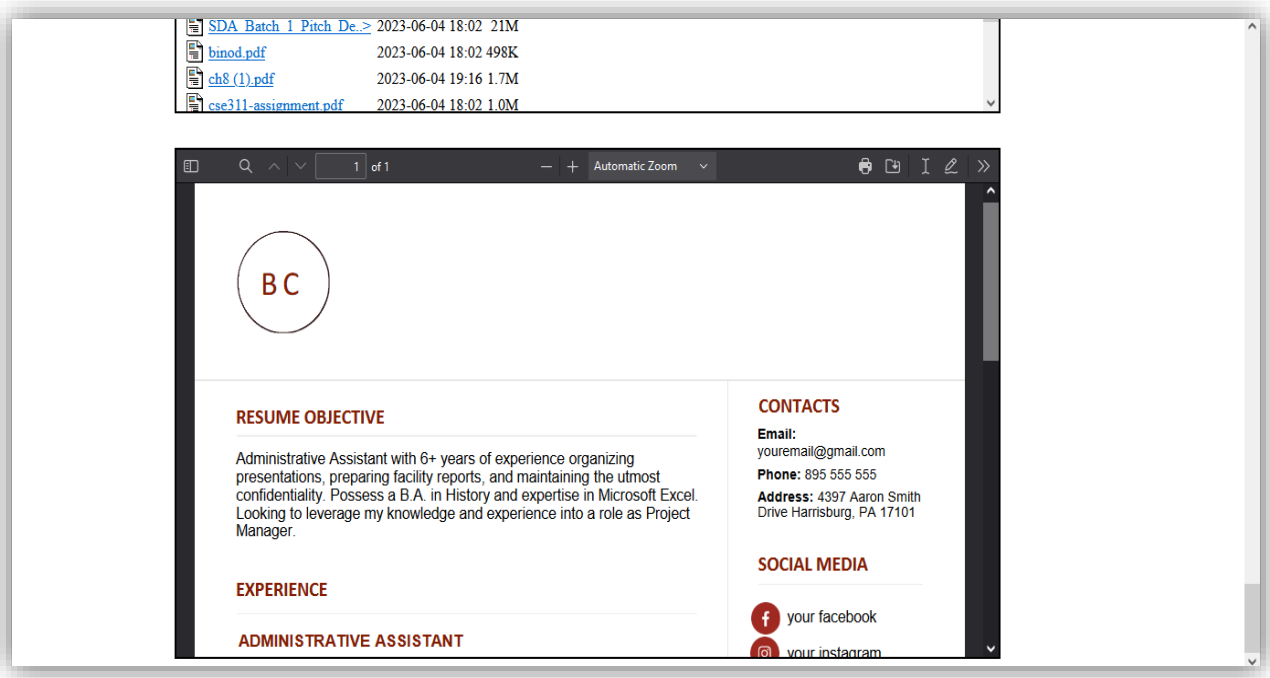
Admin Profile

Admin inbox (before)

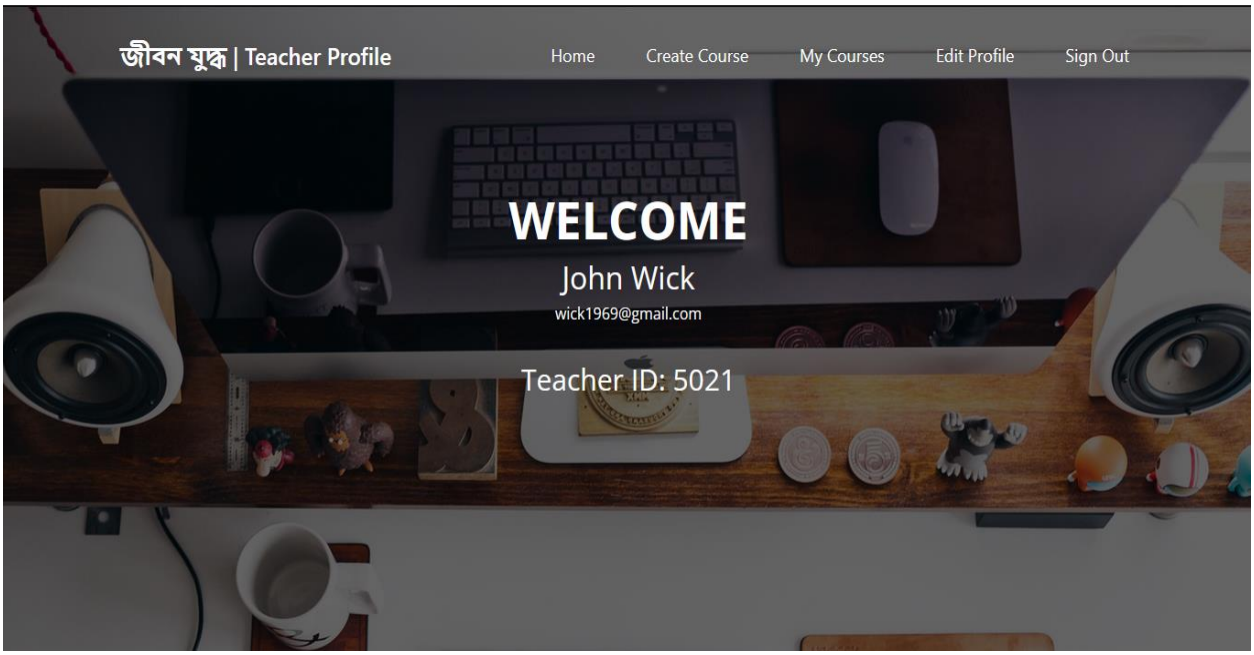


Admin Inbox (Update 1, 2)

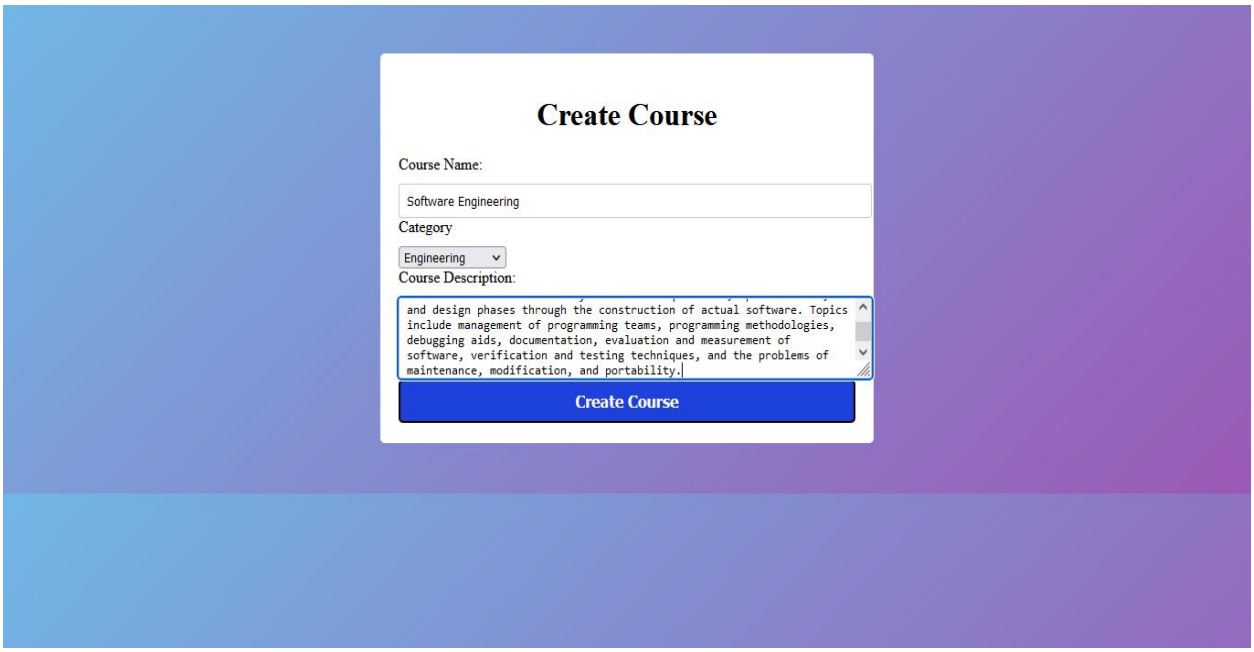
Here, if the admin clicks on **view Document,** he/she would be able to view individual documents. Whereas, if he/she clicks **View All Documents** then he/she would view the entire document page.



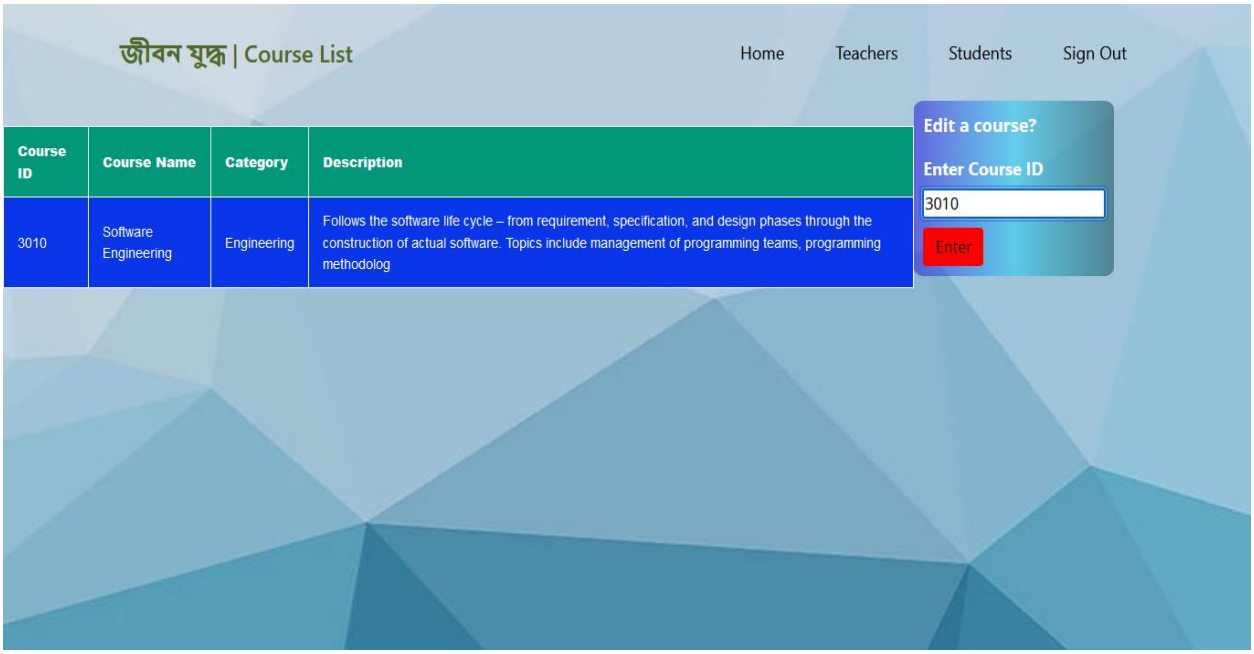
View Document Page, for admin



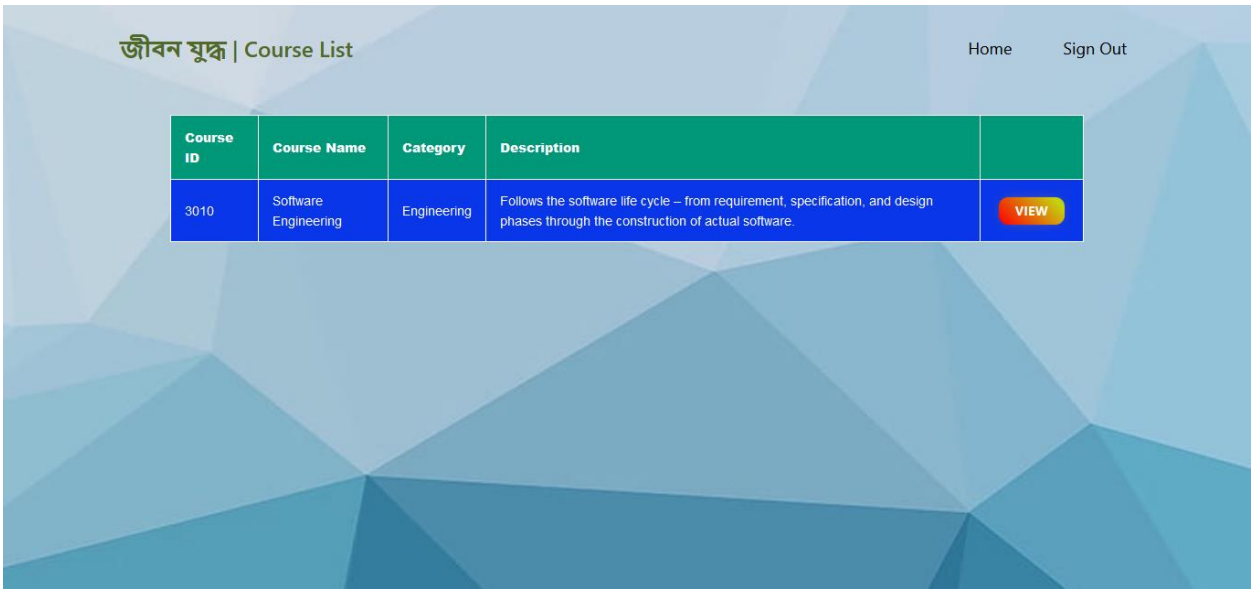
Teacher's Profile (post-verification)



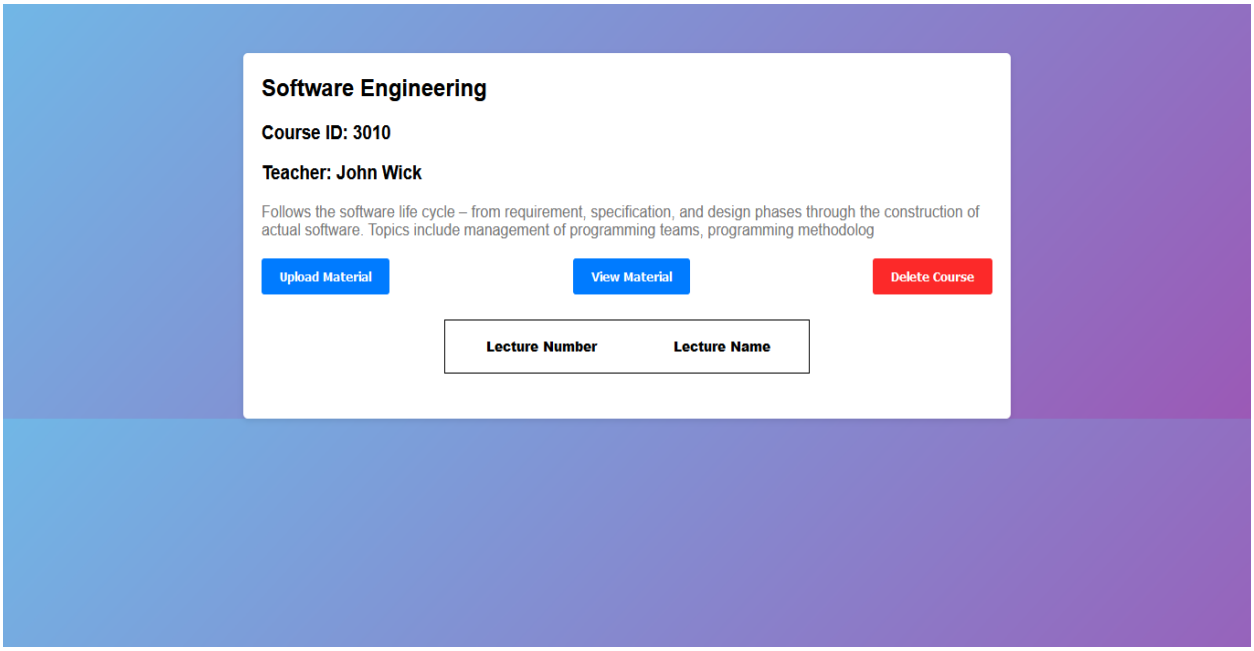
Create Course Page



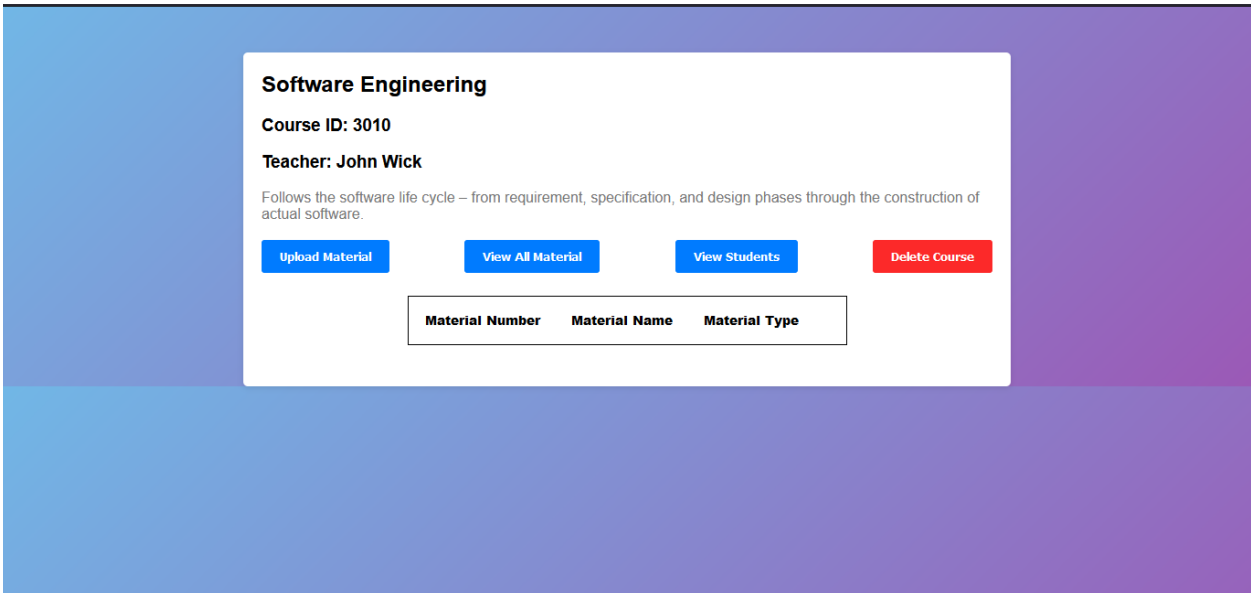
My Courses Page, for Teacher (Before)



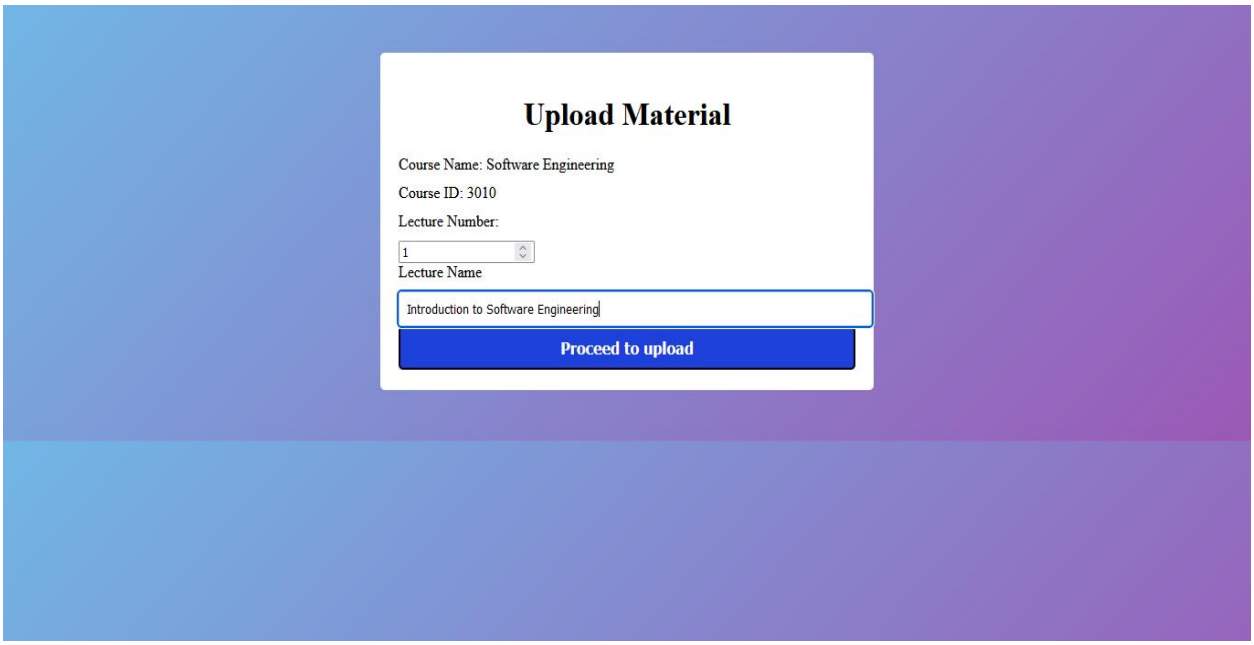
My Courses Page, for Teacher (Update 1)



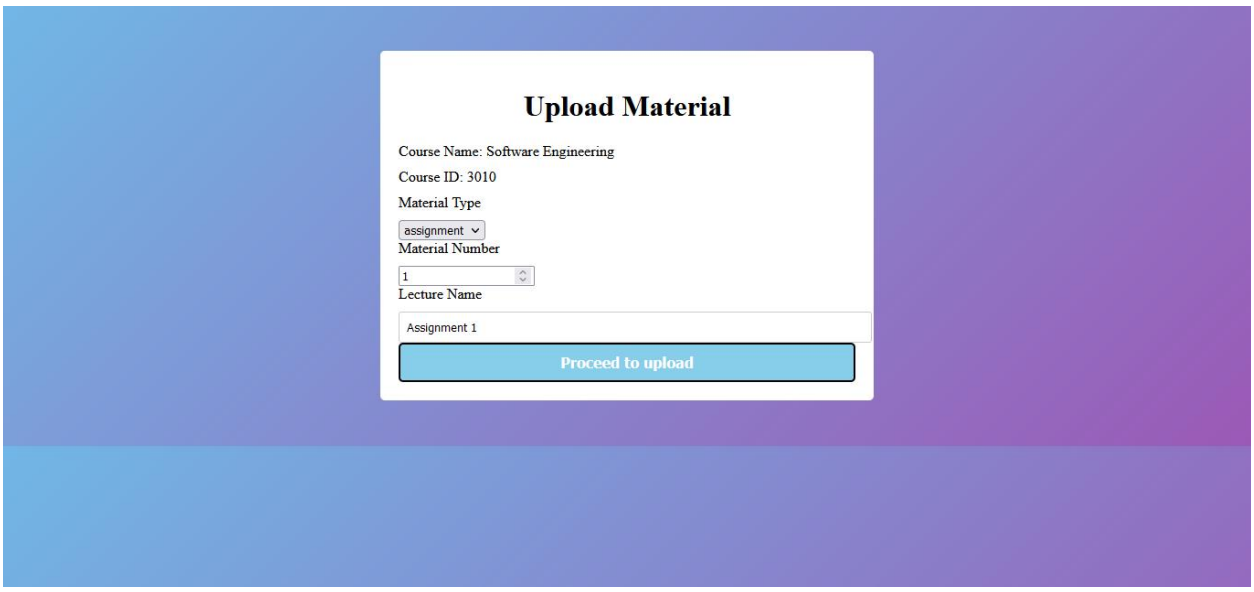
View Course, for Teacher (Before)



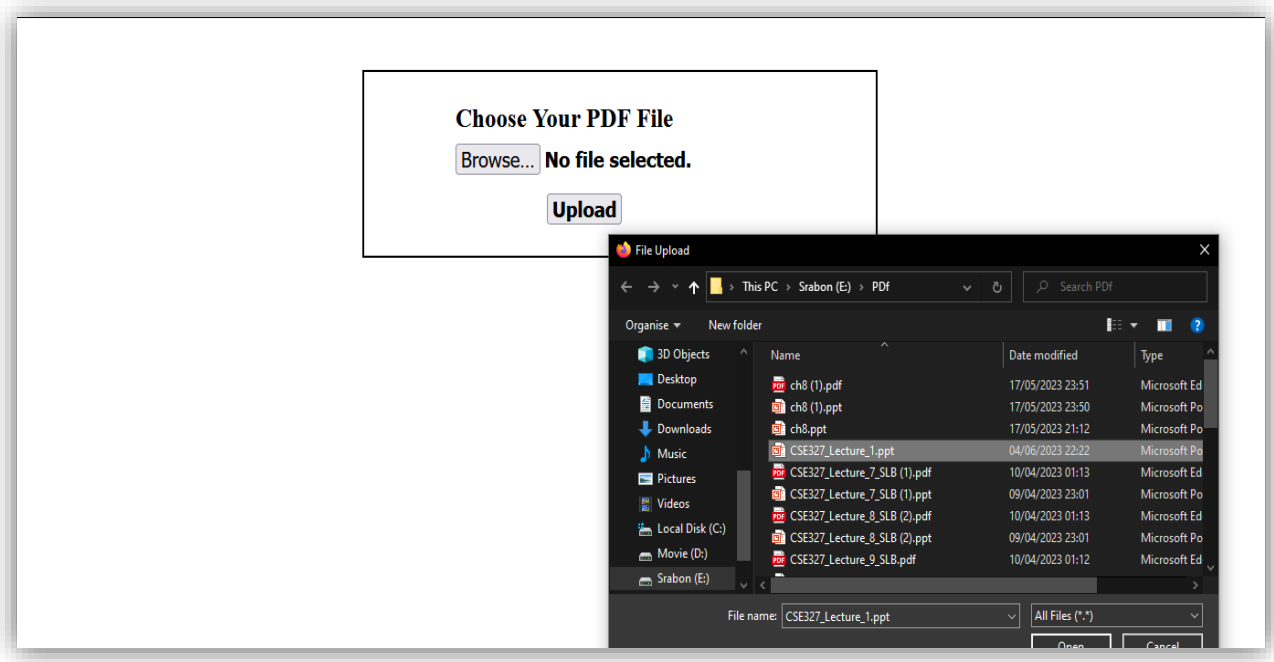
View Course, for Teacher (Update 3)



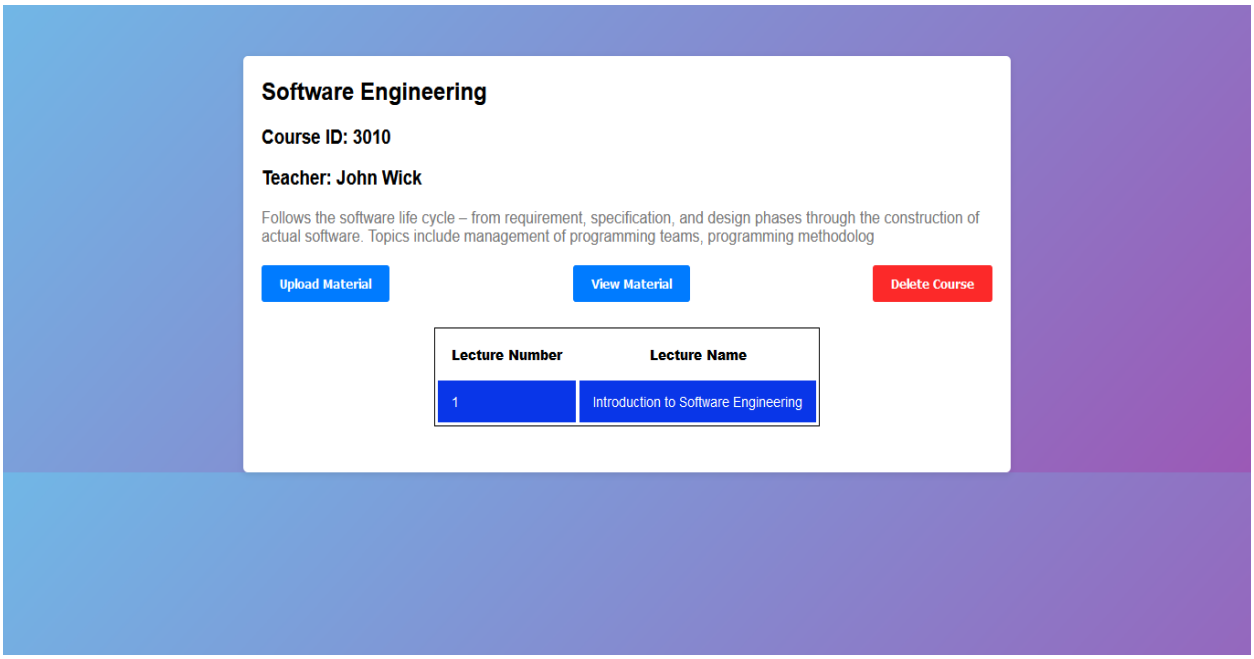
Upload Material, for Teacher (Before)



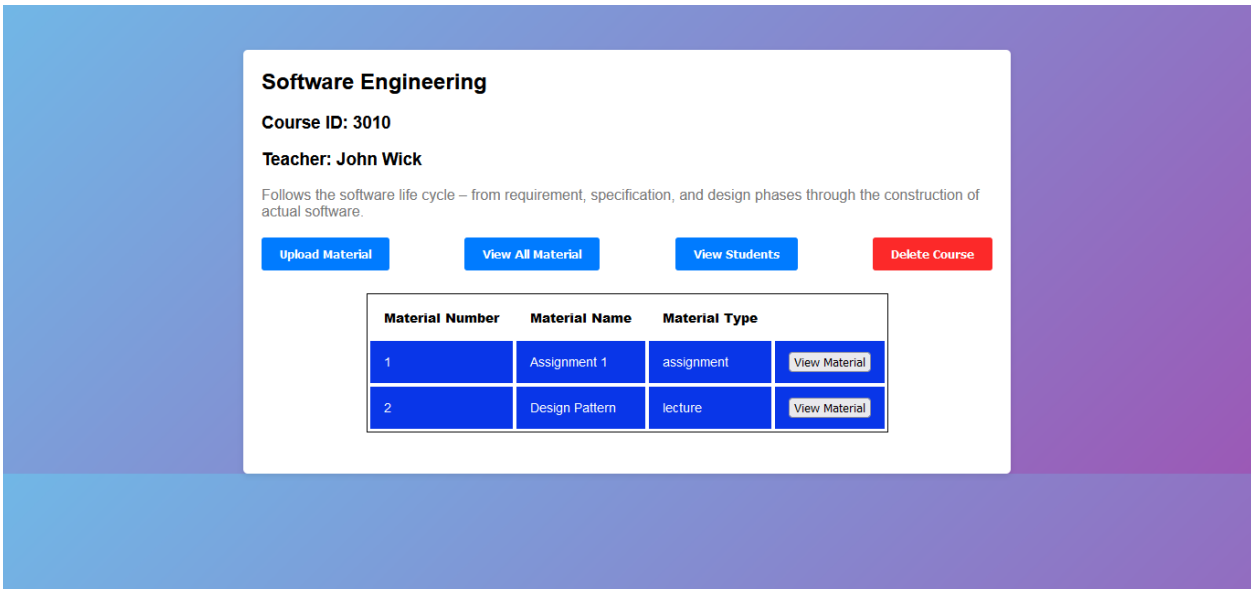
Upload Material, for Teacher (Update 5)



Choose Material, for Teacher

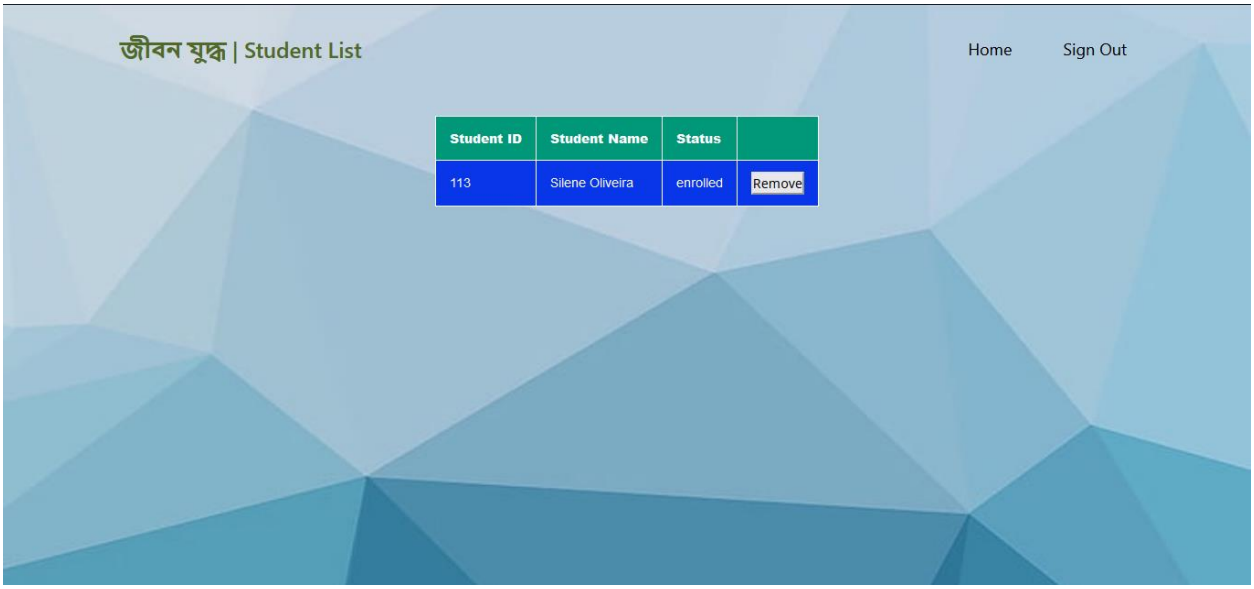


Course Profile Page, for Teacher (Before)

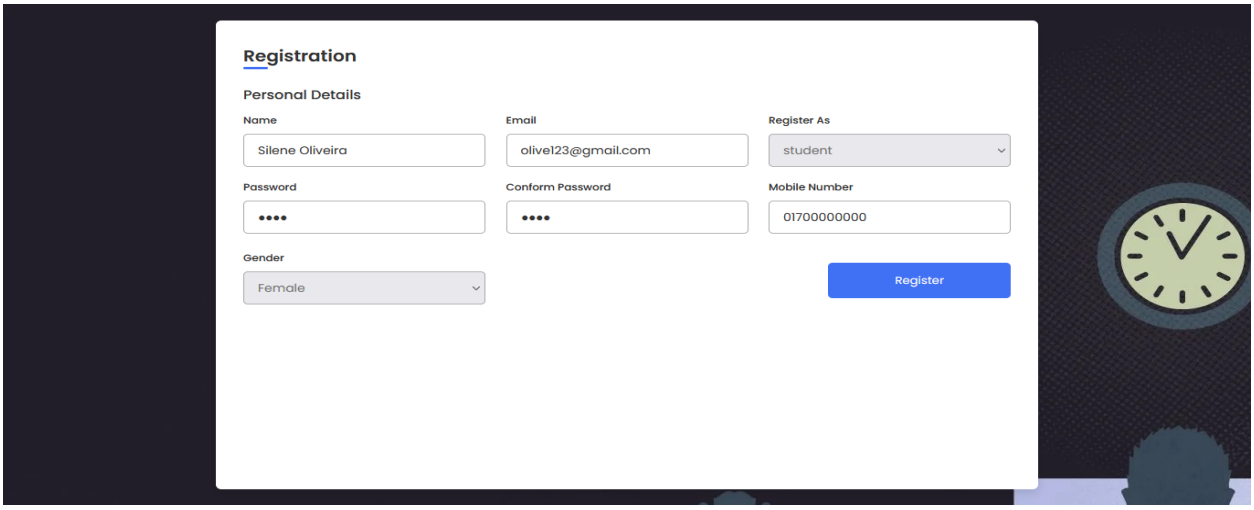


Course Profile Page, for Teacher (Update 1, 3)

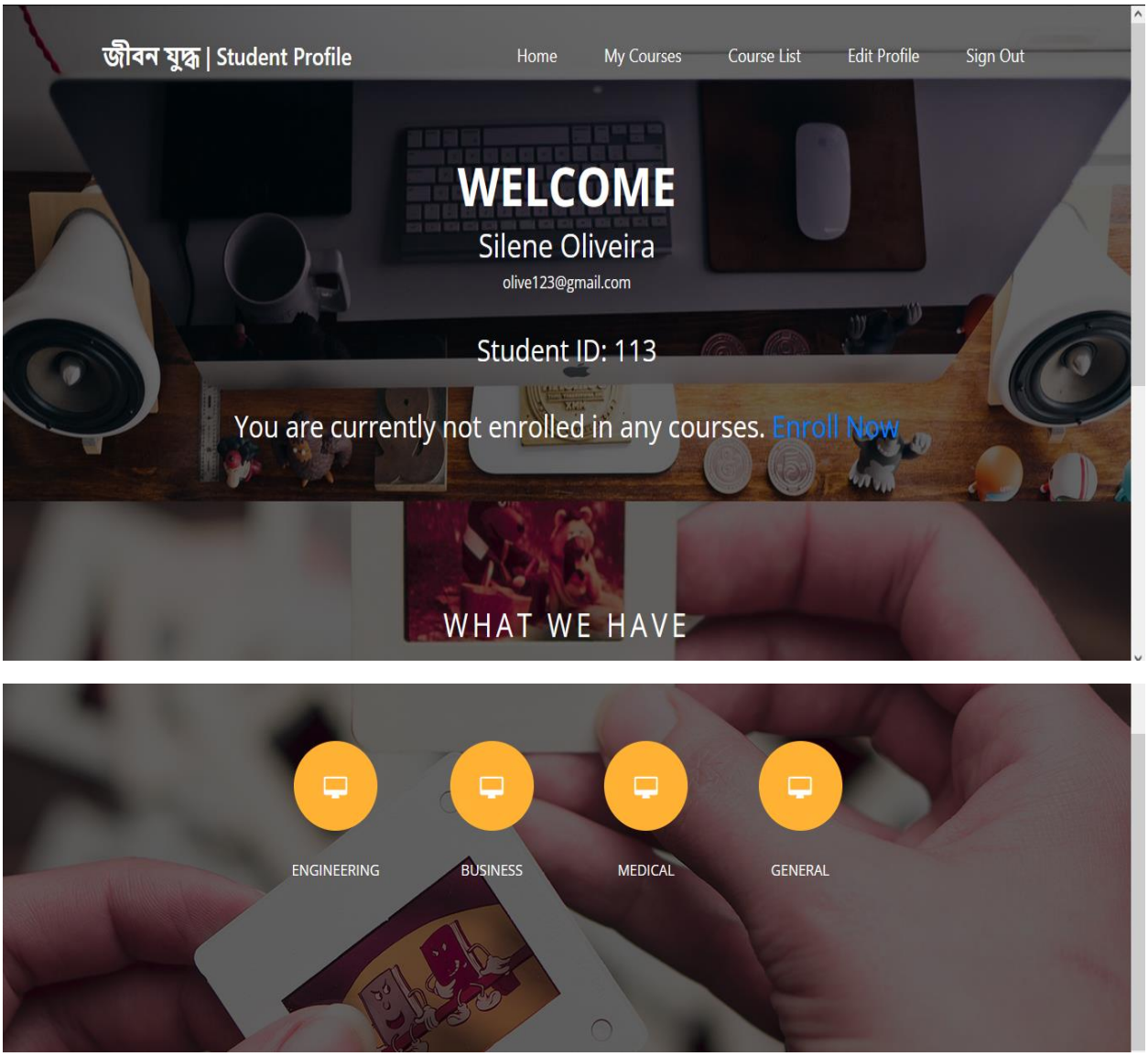
Here, if the Teacher clicks on **view material** he/she would be able to view individual materials. Whereas, if he/she clicks **View all material** then he/she would view the entire material page.



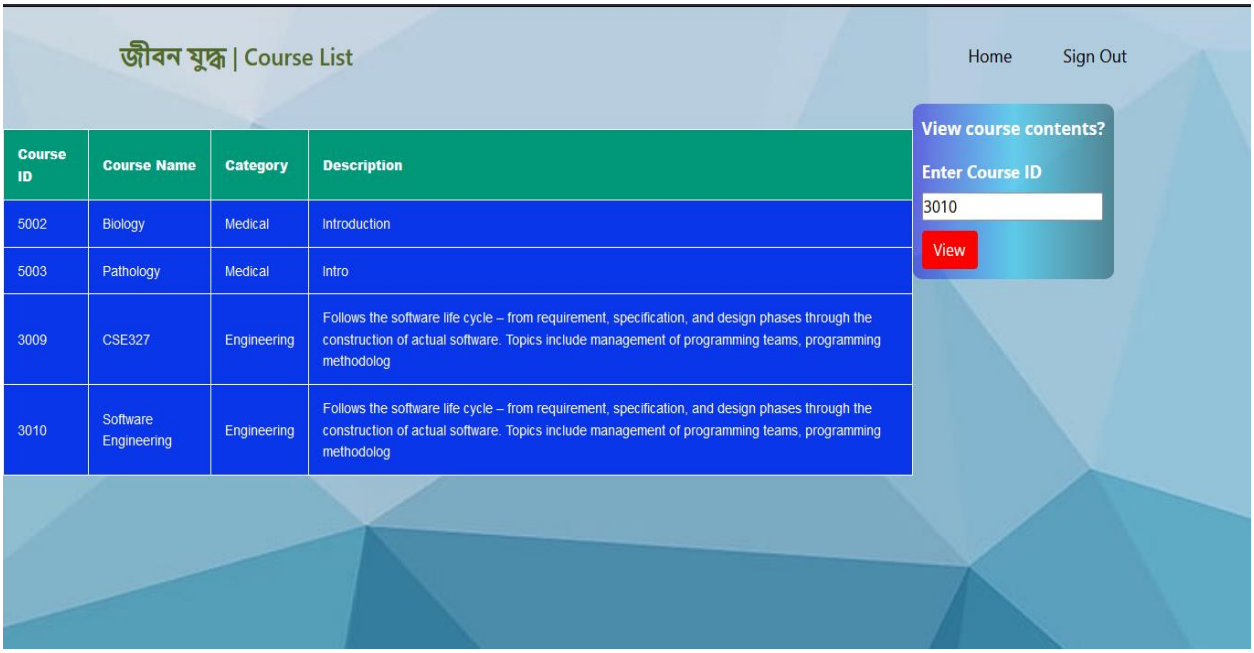
Enrolled Student List, for Teacher (Update 1, 3, 4)



Student Registration Page

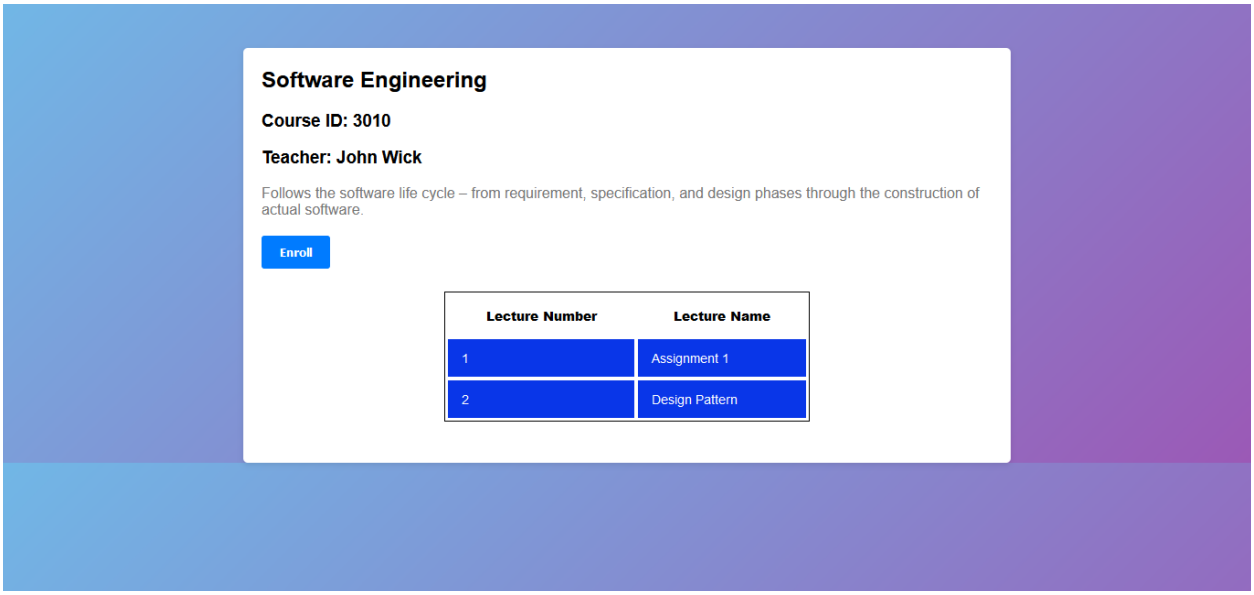


Student Profile (not enrolled yet)

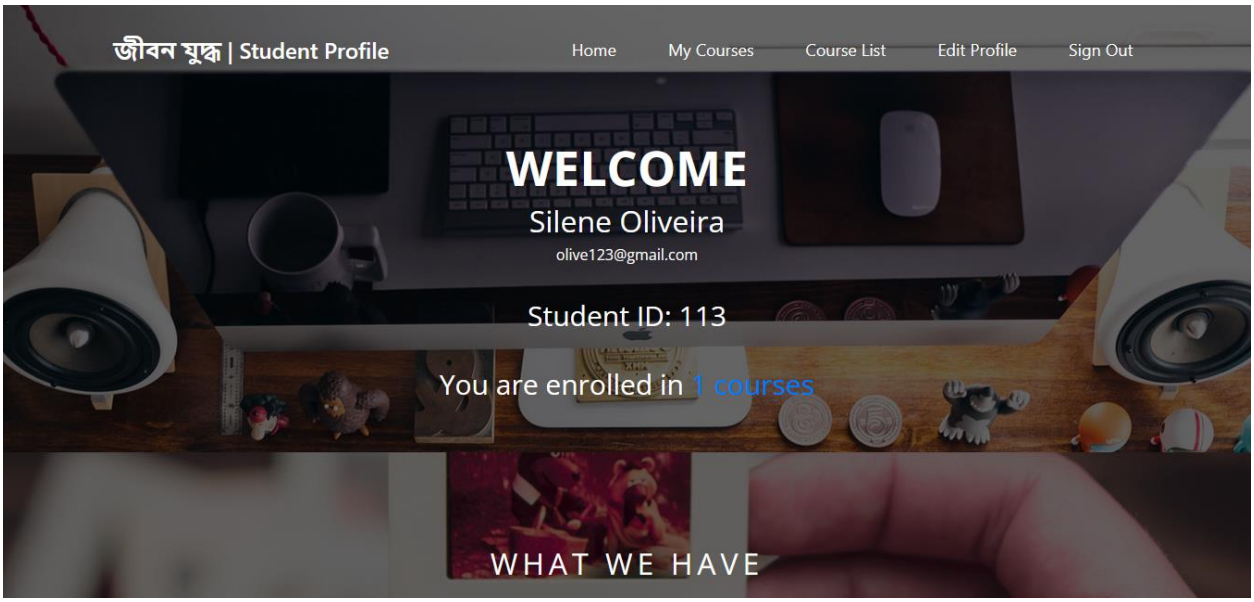


Course List, for Student (Before)

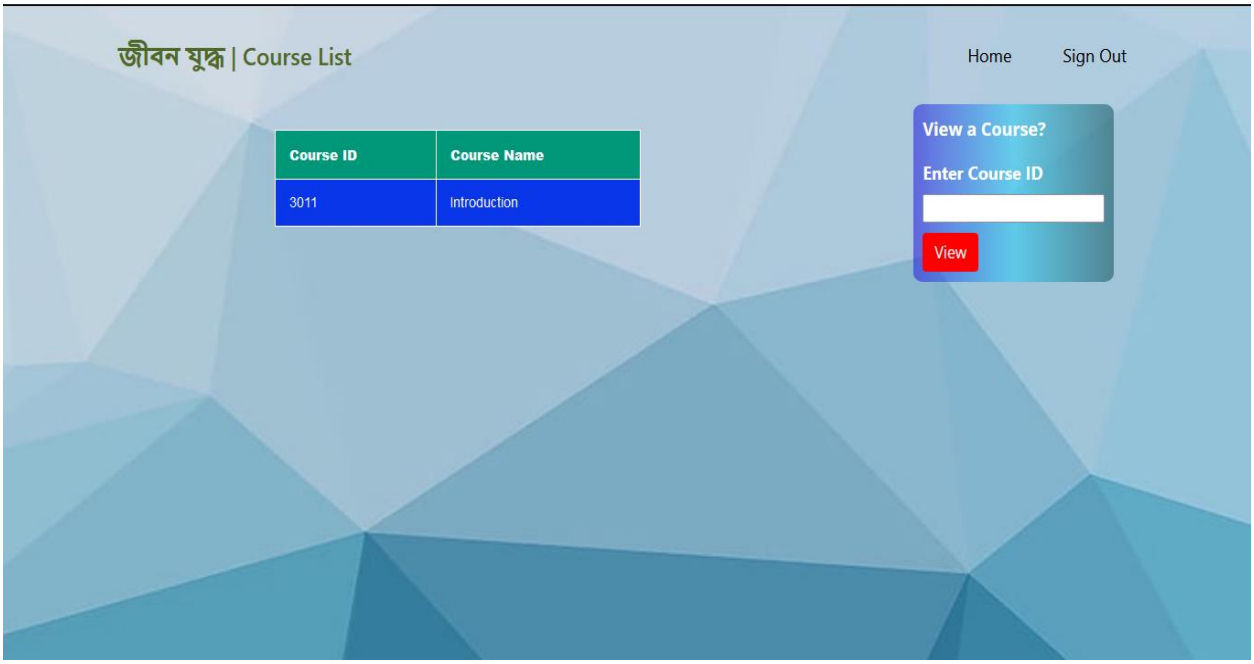
Course List, for Student (Update 1)



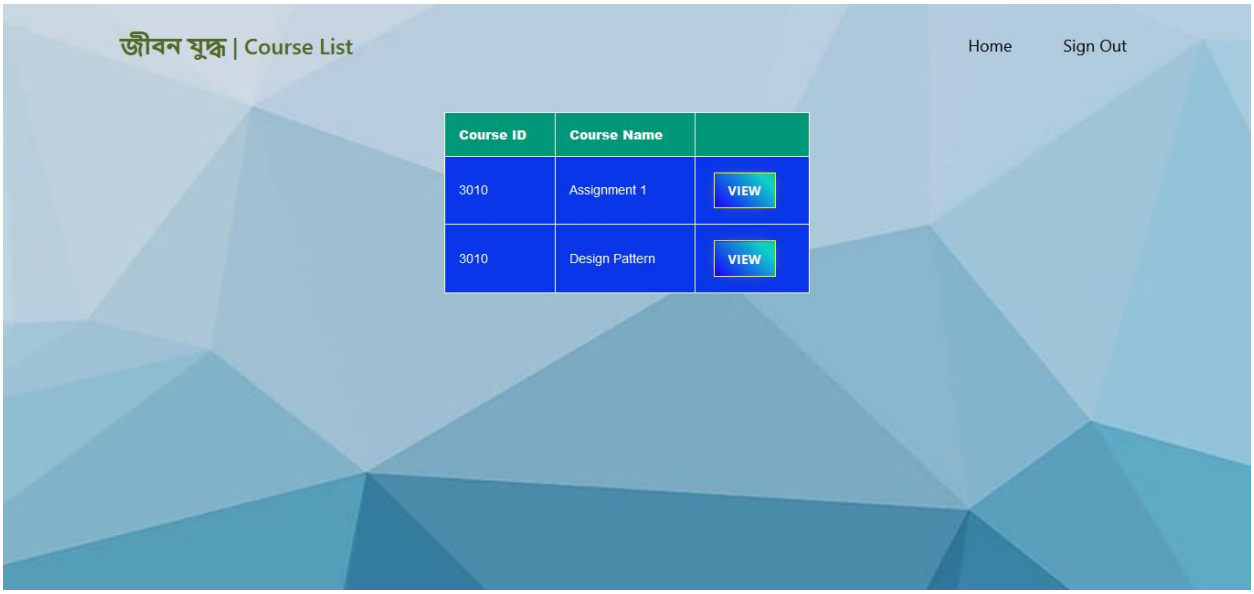
Enroll Page, for Student



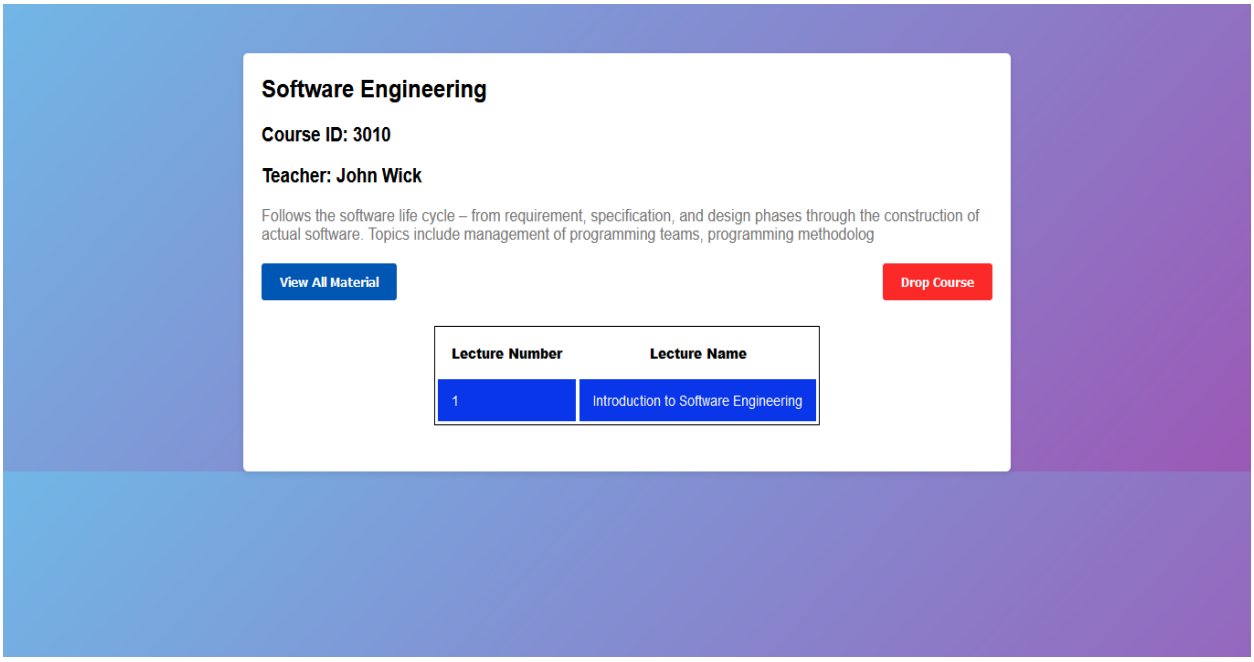
Student Profile (Post-Enrollment)



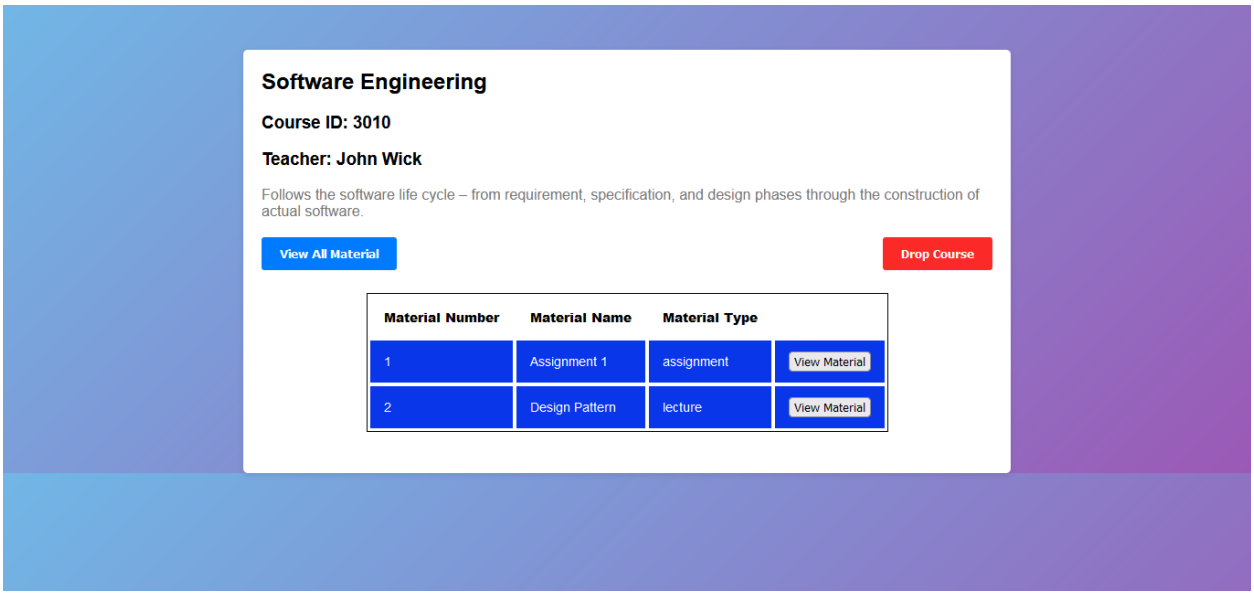
My Courses Page, for Student (Before)



My Courses Page, for Student (Update 1)

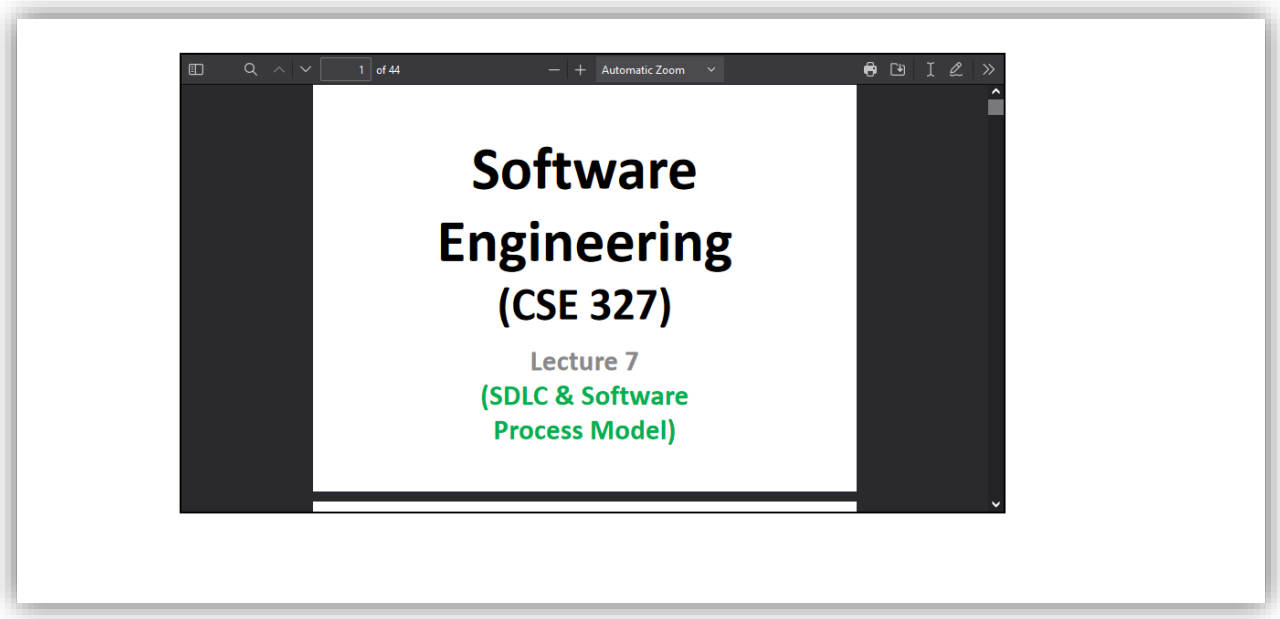


View Course, for Student (Before)

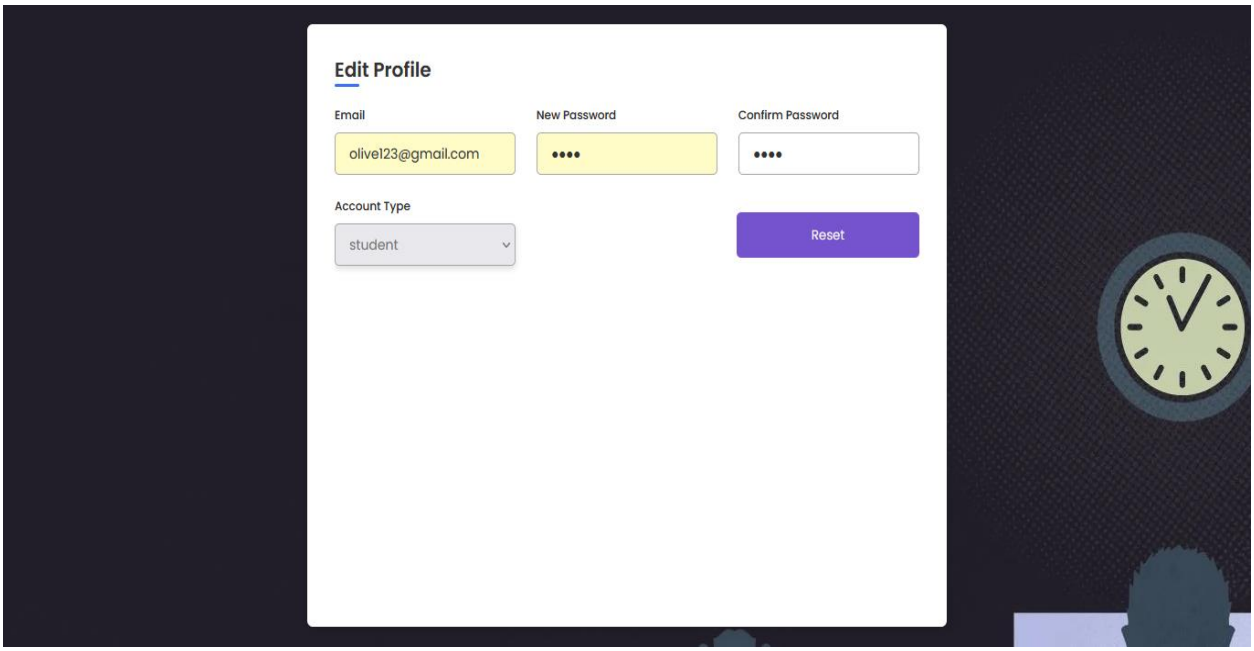


View Course, for Student (Updated 1, 6)

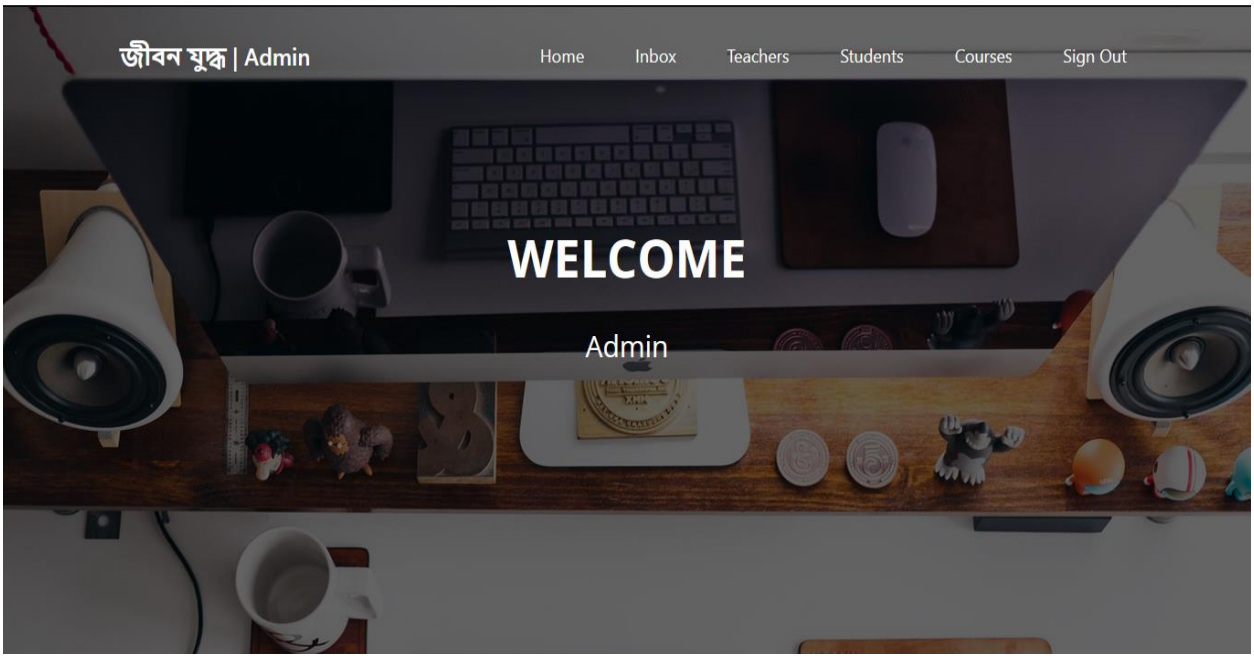
Here, if the student clicks on **view material** he/she would be able to view individual materials. Whereas, if he/she clicks **View all material** then he/she would view the entire material page.



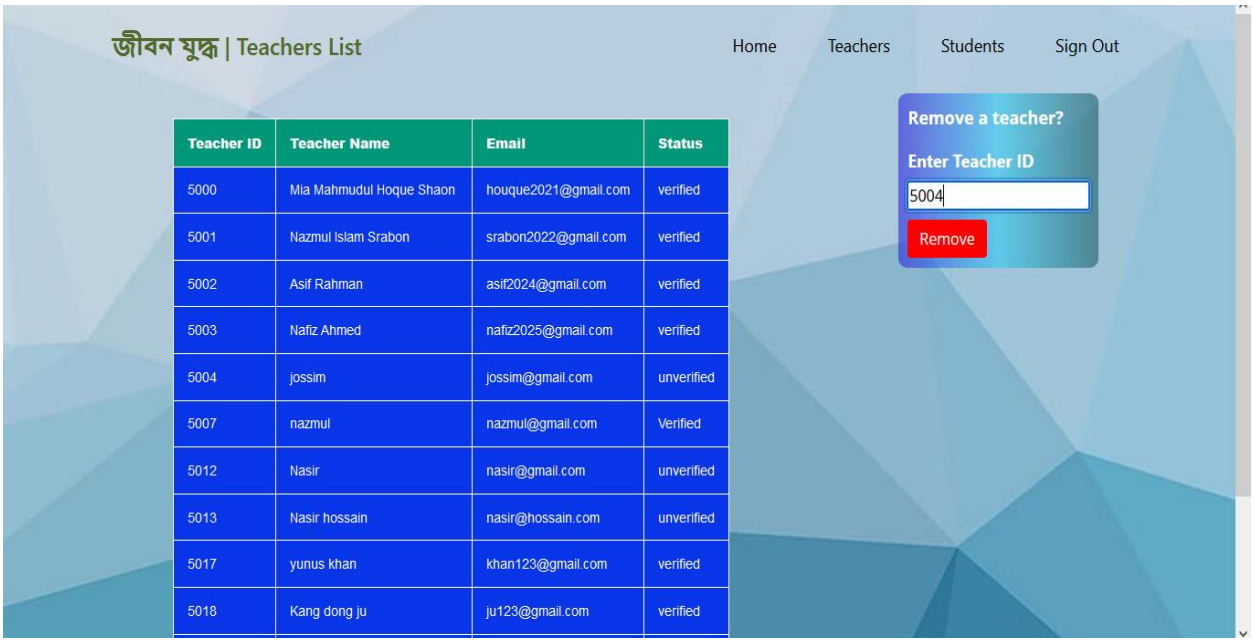
View Material Page, for Student



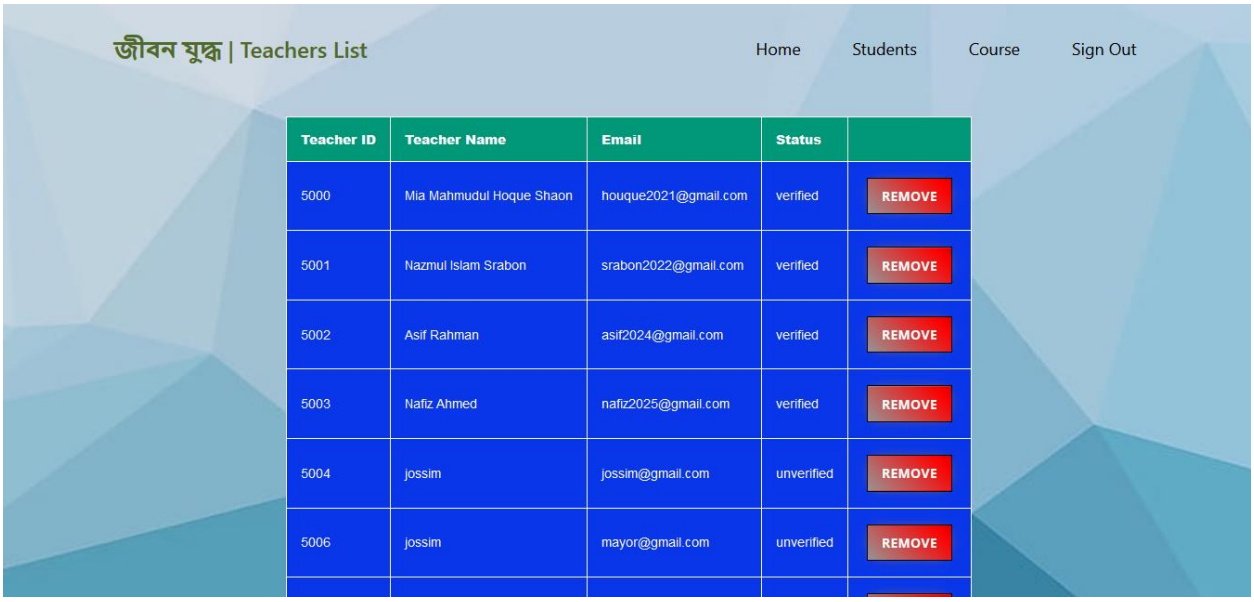
Edit Profile Page, for both Student & Teacher



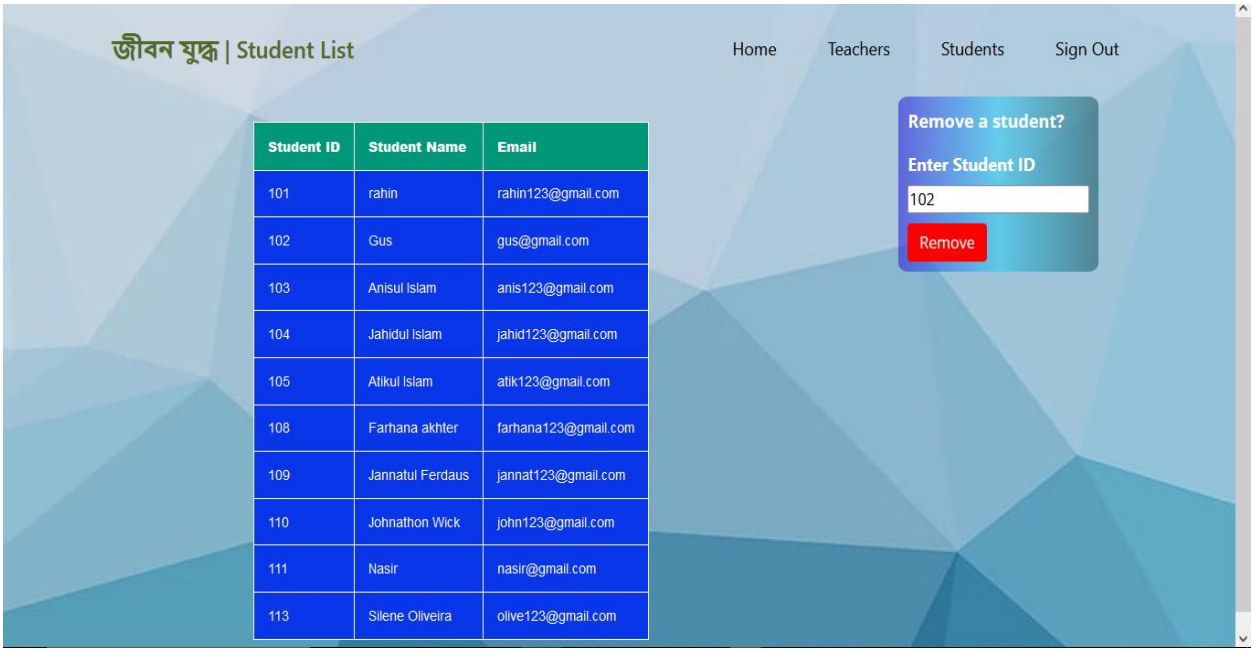
Profile Page, for Admin



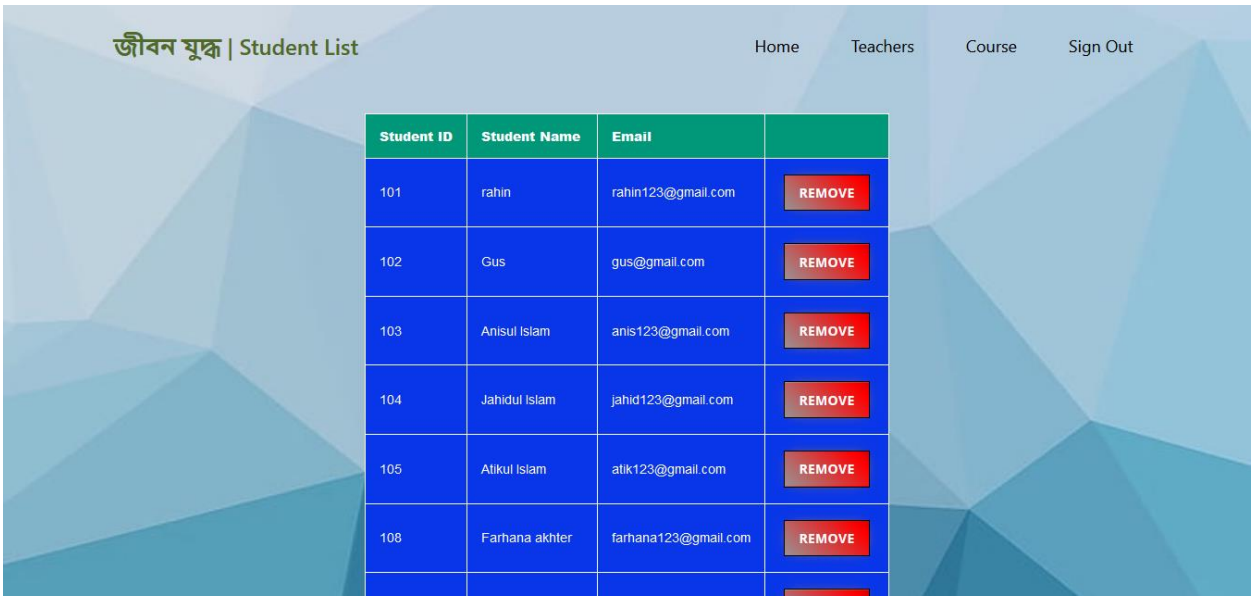
Teachers List, for Admin (Before)



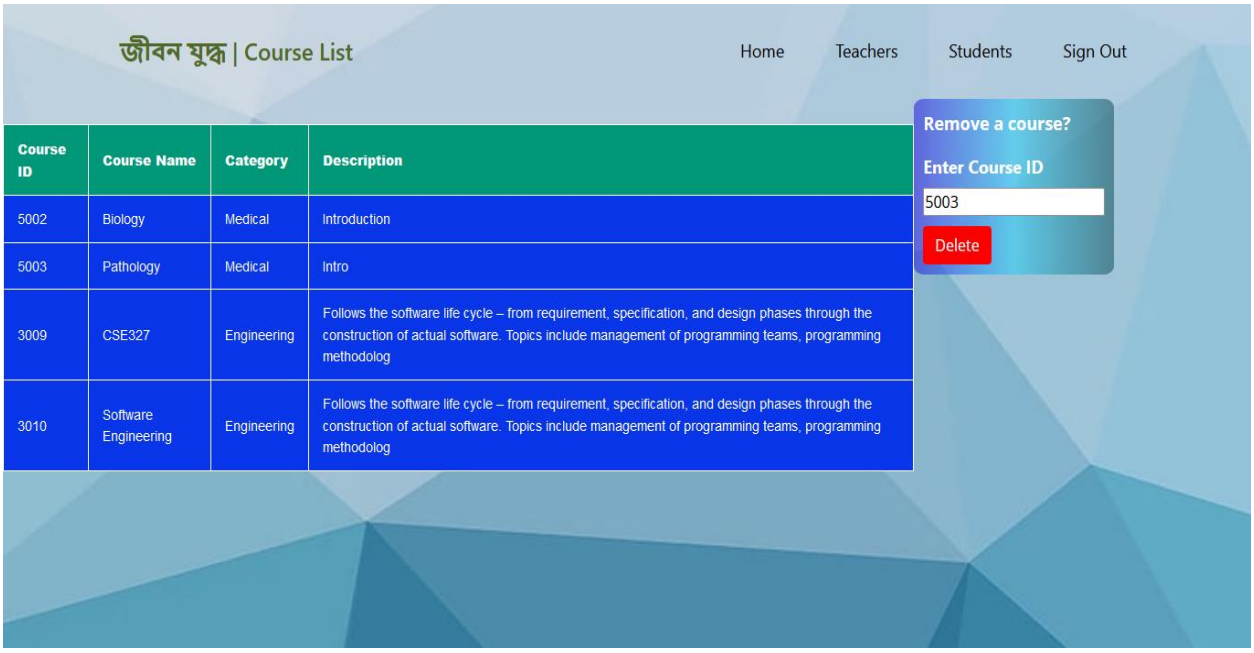
Teacher's List, for Admin (Update 1)



Student List, for Admin (Before)



Student List, for Admin (Update 1)



Course List, for student (Before)

Course List, for Admin (Update 1)