PathFinder

A Learning Management System

GITHUB Repository: https://github.com/suhaskollur/PathFinder

PROBLEM STATEMENT

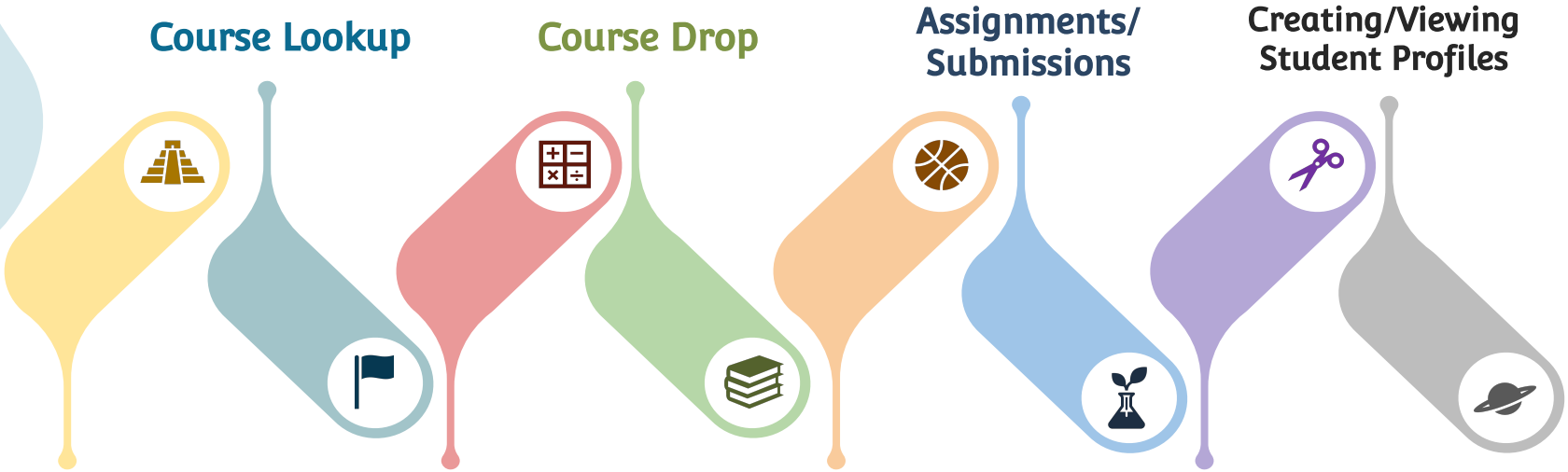
1. The project aims to develop a robust Learning Management System (LMS), namely PATHFINDER, tailored to the needs of Rutgers University-New Brunswick's Electrical and Computer Engineering Department.

2. This system will streamline various efficient student operations, including Student Operations that include, Student Registration/Login, Course Lookup, Course Enrollment, Course Drop, Adhering to Announcements posted globally, Submitting Assignments, Viewing Grades and setting up Student Profiles.

3. Coming to the Professor Operations, these include Professor Registration/Login, Listing Courses, enrolling to the courses to be taught by the professor, Posting Assignments and its grades to the respective students, publishing Announcements and Creating new Courses and many more.

4. The system is engineered to meet the demanding standards and dynamic needs of higher education in the fields of electrical and computer engineering.

Student Operations



Student Registration/Login

This component serves as the gateway for students to access various features and functionalities within our system.

The Student Registration/Login Component provides students with a streamlined process to create accounts and log in securely. It is designed to handle user authentication, account creation, and password management efficiently.

1. User Authentication: Ensures only authorized users access the system through robust verification.
2. Account Creation: Students easily create accounts with user-friendly registration.
3. Password Management: Safeguards user credentials with secure storage and features like hashing.

id	net_id	email	userPassword	first_name	last_name
1	sk2870	kollursuhas.us@gmail.com	Suhas123$	Suhas	Kollur
2	pk674	parth.kharkar@gmail.com	Parth1810	Parth	Kharkar
3	ssk241	shreyashkalaled@gmail.com	ShreyashKalal	Shreyash	Kalal
4	dp1351	devanshipatel885@gmail.com	DevanshiPatel	Devanshi	Patel
5	sk2907	sarvesh.kharche@rugters.edu	SarveshKharche	Sarvesh	Kharche
6	rk1108	rakshitha.kollur@rugters.edu	RakshithaKollur	Rakshitha	Kollur
7	ag2384	ag2384@scarletmail.rutgers.edu	AdishGolechha	Adish	Golechha
NULL	NULL	NULL	NULL	NULL	NULL

students table

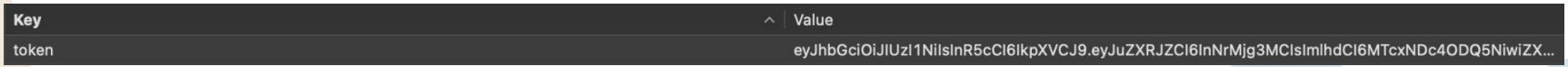
JWT Token Authentication

When a student logs in with valid credentials, our server generates a JWT token containing user information and privileges. This acts like a session token, which authenticates every subsequent request, ensuring security.

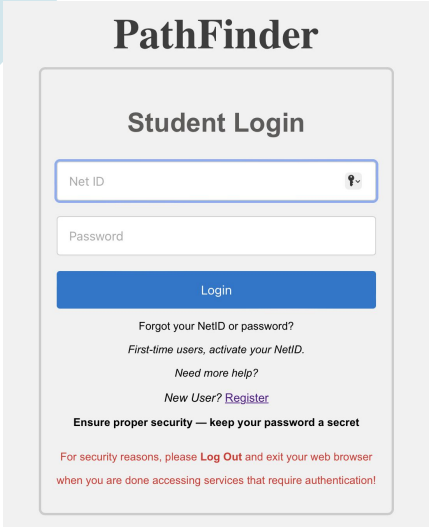
Upon receiving the JWT token, the client stores it securely in local storage or cookies. For each incoming request to protected endpoints, our server verifies the authenticity and integrity of the JWT token. Verification involves checking the token's signature using a secret key shared between the server and the authentication service.

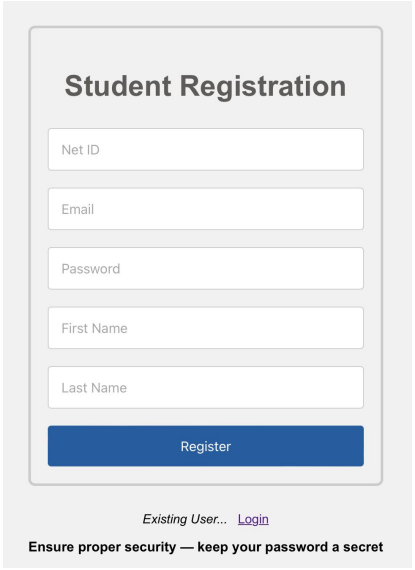
Benefits in our Project:

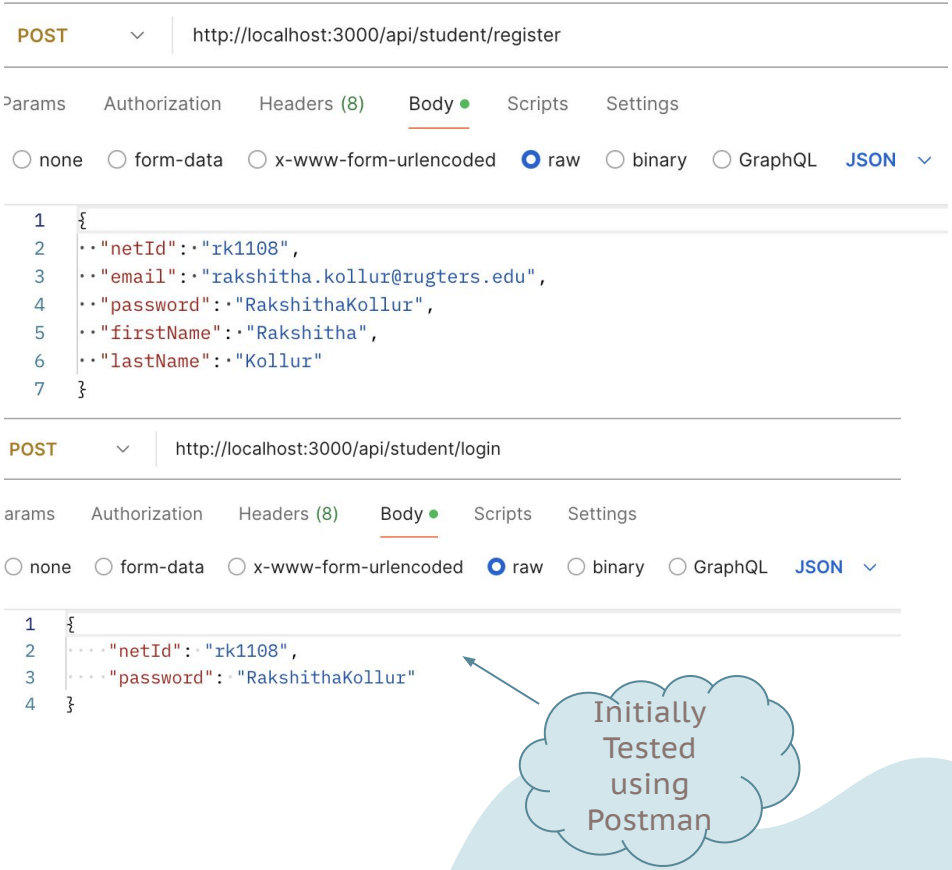
1. Improved security: JWT tokens encrypt user data and prevent unauthorized access to protected resources.
2. Scalability: Stateless nature of JWT tokens reduces server-side storage requirements and enhances performance.
3. Seamless integration: JWT authentication seamlessly integrates with our project's RESTful architecture, supporting distributed systems and microservices.



Built a ReactJS Application







Initially Tested using Postman

Course Retrieval and Enrollment

Students have the flexibility to explore available courses based on various criteria such as department, level, or specific keywords. Each course listing presents detailed information including course code, name, description, and instructor details.

After browsing the available courses, students can select their desired courses from the list presented. Upon successful enrollment, students receive instant confirmation in the Courses dashboard, which lists the enrolled courses for the respective student.

id	course_code	course_name	course_description
1	16:332:501	SYSTEMS ANALYSIS	Fundamentals of linear system concepts via sol…
2	16:332:502	TECHNOLOGY ENTREPRENEURSHIP	Structure and framework of entrepreneurial end…
3	16:332:503	PROGRAMMING METHODOLOGY FOR NUM…	Fundamentals of object-oriented programming a…
4	16:332:504	SENSOR-BASED SYSTEMS AND APPPLICATI…	The course will develop skills in designing, prog…
5	16:332:505	CONTROL SYSTEM THEORY	Review of basic feedback concepts and basic c…
6	16:332:506	APPLIED CONTROLS	Review of state space techniques; transfer funct…
7	16:332:507	SECURITY ENGINEERING	Essential principles, techniques, tools, and meth…
8	16:332:508	DIGITAL CONTROL SYSTEMS	Review of linear discrete-time systems and the…
9	16:332:509	CONVEX OPTIMIZATION FOR ENGINEERING…	The course develops the necessary theory, algo…

courses table

Web Scraped Data from Rutgers ECE Website

Enrolled Courses

16:332:532

MULTIMODAL MACHINE LEARNING FOR SENSING SYSTEMS

16:332:568

SOFTWARE ENGINEERING WEB APPLICATIONS

16:332:577

ANALOG AND LOW-POWER DIGITAL VLSI DESIGN

```
Executing SQL query: [Function: query]
Parameters: [ 'sk2870' ]
Enrolled courses: [
  {
    course_code: '16:332:512',
    course_name: 'NONLINEAR AND ADAPTIVE CONTROL THEORY'
  },
  {
    course_code: '16:332:503',
    course_name: 'PROGRAMMING METHODOLOGY FOR NUMERICAL COMPUTING AND COMPU
'IONAL FINANCE'
  },
  { course_code: '16:332:900', course_name: 'Cognitive Science' },
  { course_code: '16:332:901', course_name: 'Engineering Management' }
]
Connected to the database
```

Course Enrollment

Enroll in a Course

Course Code:

Enter Course Code

Enroll

id	net_id	course_id	course_code	course_name
1	sk2870	11	16:332:512	NONLINEAR AND ADAPTIVE CONTROL THEORY
2	sk2870	3	16:332:503	PROGRAMMING METHODOLOGY FOR NUMERI…
3	sk2870	76	16:332:900	Cognitive Science
4	sk2870	77	16:332:901	Engineering Management
5	rk1108	76	16:332:900	Cognitive Science
6	ssk241	21	16:332:532	MULTIMODAL MACHINE LEARNING FOR SENSI…
7	ssk241	41	16:332:568	SOFTWARE ENGINEERING WEB APPLICATIONS
8	ssk241	48	16:332:577	ANALOG AND LOW-POWER DIGITAL VLSI DESI…
NULL	NULL	NULL	NULL	NULL

enrollments table

Assignments/Submissions

Upon logging into the platform, students are greeted with an intuitive user interface where they can navigate to the "Assignments" section effortlessly.
Within the "Assignments" section, students can view a list of assignments organized by their enrolled courses. So a student can view the assignments posted by the respective professors for the courses he/she has enrolled for.

id	course_id	course_code	course_name	course_instruc...	assignment_ti...	assignment_description	assignment_deadli...
1	76	16:332:900	Cognitive Science	Suhas Kollur	Assignment 1	Basics of Cognitive Science	2024-05-16 00:00:00
2	76	16:332:900	Cognitive Science	Suhas Kollur	Assignment 2	Hands-on Cognitive Science	2024-05-21 00:00:00
3	77	16:332:901	Engineering Management	Suhas Kollur	Assignment 1	Introduction to Engineering Management	2024-05-23 00:00:00
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

professor_assignment table

```
// Route to retrieve assignments for a student
router.get('/assignments', authenticateStudent, async (req, res) => {
  const { netId } = req.student; // Extract netId from authenticated student

  try {
    const assignments = await getAssignmentsForStudent(netId); // Call the function to retrieve assignments
    return res.status(200).json(assignments);
  } catch (error) {
    console.error('Error getting assignments for student:', error);
    return res.status(500).json({ message: 'Internal server error' });
  }
});
```

Assignments

Course Code: 16:332:900

Course Name: Cognitive Science

Course Instructor: Suhas Kollur

Assignment Title: Assignment 1

Assignment Description: Basics of Cognitive Science

Assignment Deadline: 2024-05-16T04:00:00.000Z

Submit

Course Code: 16:332:900

Course Name: Cognitive Science

Course Instructor: Suhas Kollur

Assignment Title: Assignment 2

Assignment Description: Hands-on Cognitive Science

Assignment Deadline: 2024-05-21T04:00:00.000Z

Submit

Allows students access to submit assignments


```
Assignments for student: [
  {
    id: 1,
    course_id: 76,
    course_code: '16:332:900',
    course_name: 'Cognitive Science',
    course_instructor: 'Suhas Kollur',
    assignment_title: 'Assignment 1',
    assignment_description: 'Basics of Cognitive Science',
    assignment_deadline: 2024-05-16T04:00:00.000Z
  },
  {
    id: 2,
    course_id: 76,
    course_code: '16:332:900',
    course_name: 'Cognitive Science',
    course_instructor: 'Suhas Kollur',
    assignment_title: 'Assignment 2',
    assignment_description: 'Hands-on Cognitive Science ',
    assignment_deadline: 2024-05-21T04:00:00.000Z
  },
  {
    id: 3,
    course_id: 77,
    course_code: '16:332:901',
    course_name: 'Engineering Management',
    course_instructor: 'Suhas Kollur',
    assignment_title: 'Assignment 1',
    assignment_description: 'Introduction to Engineering Management',
    assignment_deadline: 2024-05-23T04:00:00.000Z
  }
]
```

Students can submit their assignments through a dedicated submission interface accessible within the platform.

Made use of Multer, a popular middleware for handling file uploads in Node.js, is utilized for file upload functionality. In addition to uploading files, students can provide a brief description or commentary related to their assignment submission.

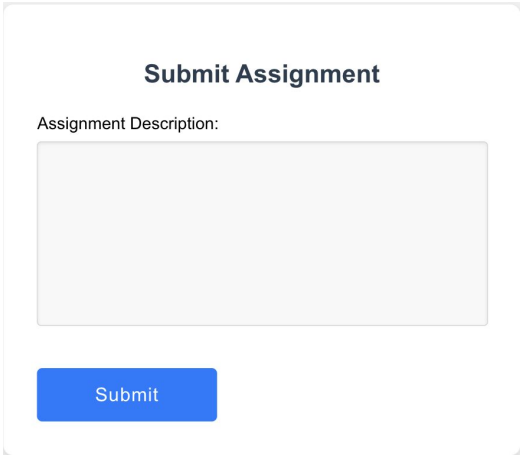
Once the student submit the assignment it gets stored in the submissions table, and the file gets stored in a specific file path accessible to our application.

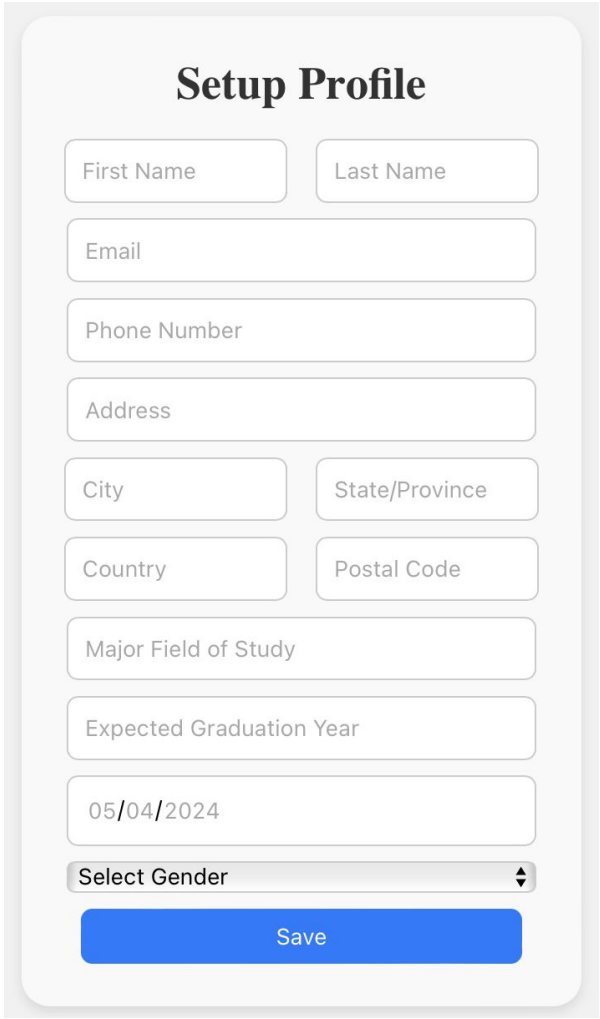
```
// Store the file in a location accessible to your application
const filePath = `path/to/storage/${Date.now()}_${Math.random()}.pdf`;

fs.writeFile(filePath, file, 'base64', (err) => {
  if (err) {
    console.error('Error writing file:', err);
    return res.status(500).json({ message: 'Error submitting assignment' }
  }
  console.log('File written successfully:', filePath);
});
```

```
CREATE TABLE submissions (
    id INT AUTO_INCREMENT PRIMARY KEY,
    assignment_id INT NOT NULL,
    student_id INT NOT NULL,
    file_path VARCHAR(255) NOT NULL,
    description TEXT,
    submission_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (assignment_id) REFERENCES assignments(id),
    FOREIGN KEY (student_id) REFERENCES students(id)
);
```

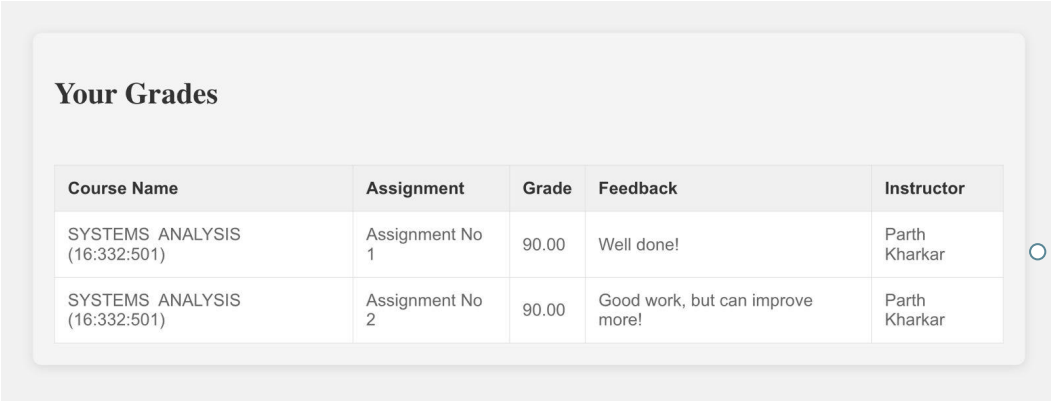
ReactJS Application

```
Compiled successfully!

You can now view student-dashboard-final in the browser.

  Local:            http://localhost:3001
  On Your Network:  http://192.168.1.236:3001

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```





Viewing Grades



Your Grades

Course Name	Assignment	Grade	Feedback	Instructor
SYSTEMS ANALYSIS (16:332:501)	Assignment No 1	90.00	Well done!	Parth Kharkar
SYSTEMS ANALYSIS (16:332:501)	Assignment No 2	90.00	Good work, but can improve more!	Parth Kharkar

Professor posts grades for the courses enrolled by the student

```
CREATE TABLE grades (
  id INT AUTO_INCREMENT PRIMARY KEY,
  student_id INT,
  assignment_id INT,
  grade DECIMAL(5, 2),
  feedback VARCHAR(255),
  FOREIGN KEY (student_id) REFERENCES students(id),
  FOREIGN KEY (assignment_id) REFERENCES professor_assignment(id)
);
```

Adhering to Announcements

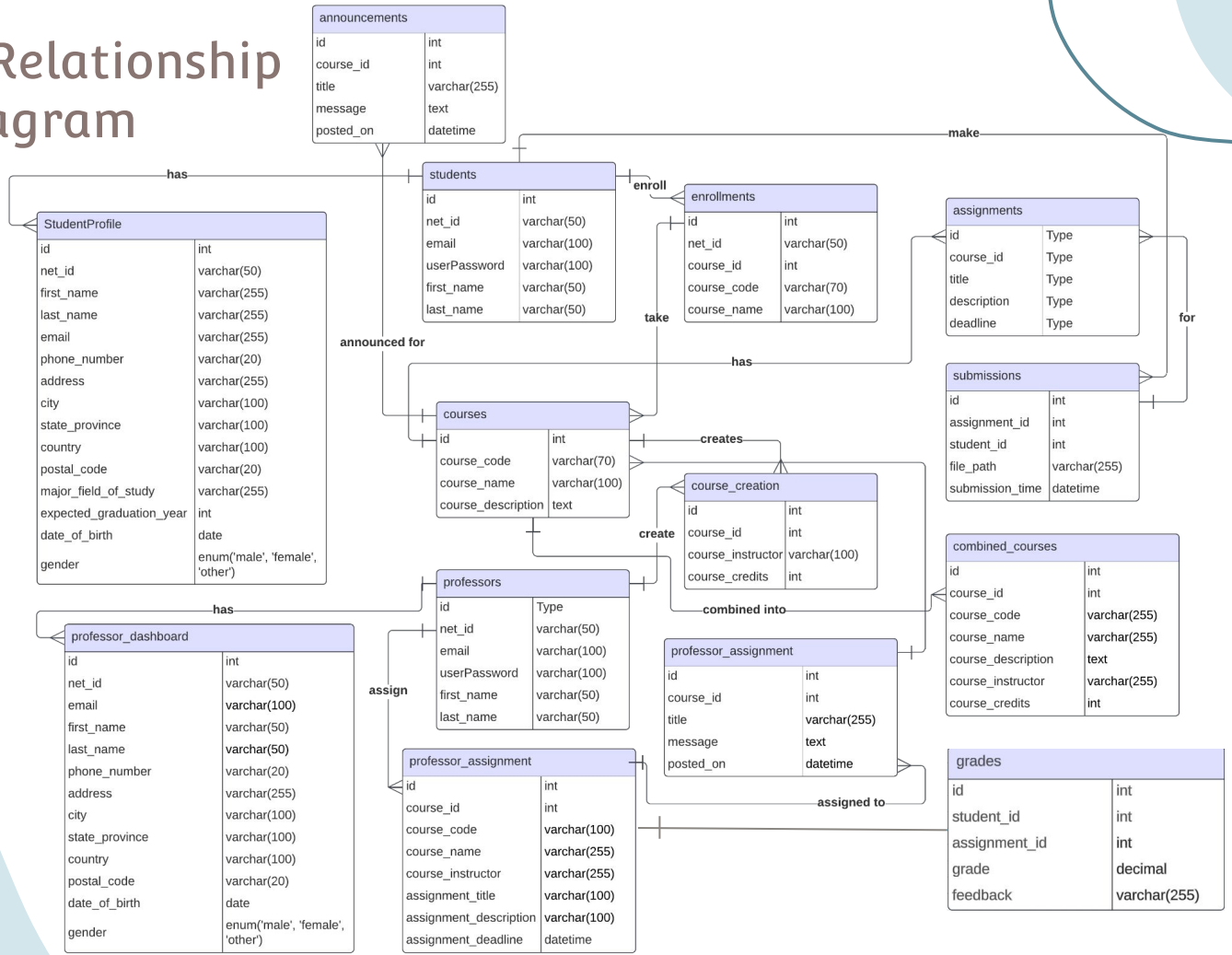
Announcements

Course Code	Course Name	Title	Message	Course Instructor	Posted On
16:332:501	SYSTEMS ANALYSIS	Midterm Reviews	The midterm reviews for the course are now open. Please fill them out. I will create an assignment so you can form your groups and start planning your final projects.	Parth Kharkar	5/5/2024
16:332:501	SYSTEMS ANALYSIS	Assignment No 1	I have just posted Assignment No 1, which is to work in groups. Please create groups and start working on it!	Parth Kharkar	5/5/2024
16:332:501	SYSTEMS ANALYSIS	Welcome to ECE501	Today is our First Class. Please bring you laptop if you can. I just created a zoom link for anyone who is sick and cannot make it to class. You are expected to come class unless you have a valid reason.	Parth Kharkar	5/5/2024

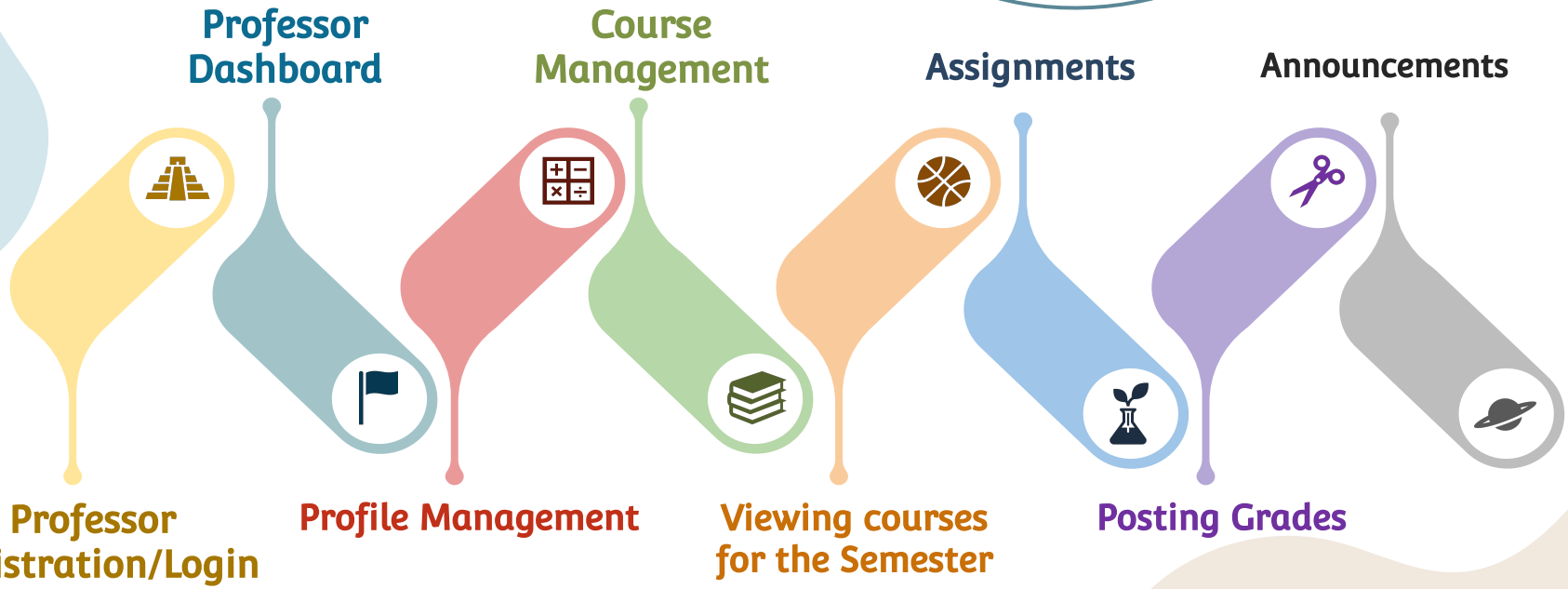
```sql
CREATE TABLE announcements (
    id INT AUTO_INCREMENT PRIMARY KEY,
    course_id INT NOT NULL,
    title VARCHAR(255) NOT NULL,
    message TEXT NOT NULL,
    posted_on DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (course_id) REFERENCES courses(id)
);
```

Students can access announcements posted by the professor

Entity – Relationship Diagram



Professor's Flow of Operations



Professor Registration / Login

This component serves as the gateway for professor's to be able to access various features associated within our system.

The Professor Registration/Login provides the professor with a streamlined process to create account and log in securely, as it is designed to handle user authentication, account creation and secure password management.

1. User Authentication: Ensures only authorized users access the system through robust verification.
2. Account Creation: User-friendly registration.
3. Password Management: Safeguards user credentials with secure storage.

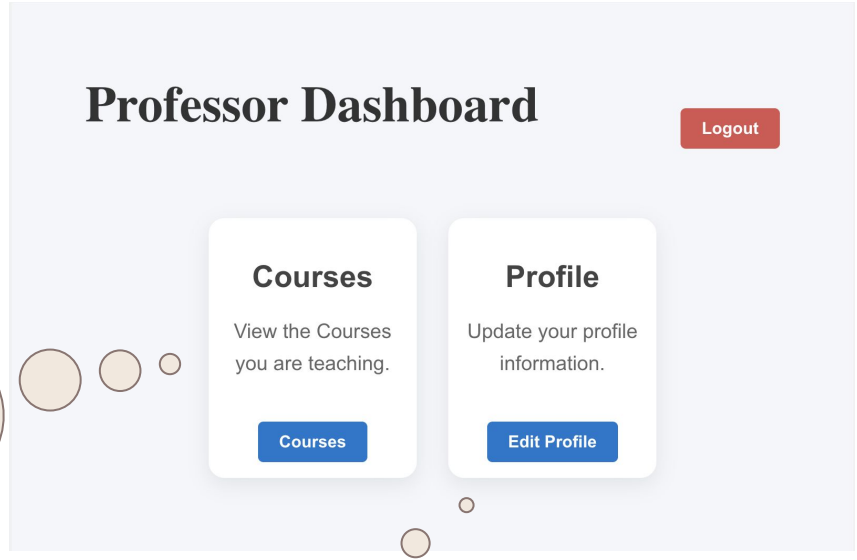
id	net_id	email	userPassword	first_name	last_name
1	pk674	parthkharkar@gmail.com	Parth123	Parth	Kharkar
2	sk2870	suhaskollur@gmail.com	Suhas999	Suhas	Kollur
3	mm288	manasmaskar@gmail.com	mm456	Manas	Maskar
4	av860	amaanvora@gmail.com	av888	Amaan	Vora

professors table

Professor Dashboard

The Professor Dashboard contains two functional components visually represented as cards:

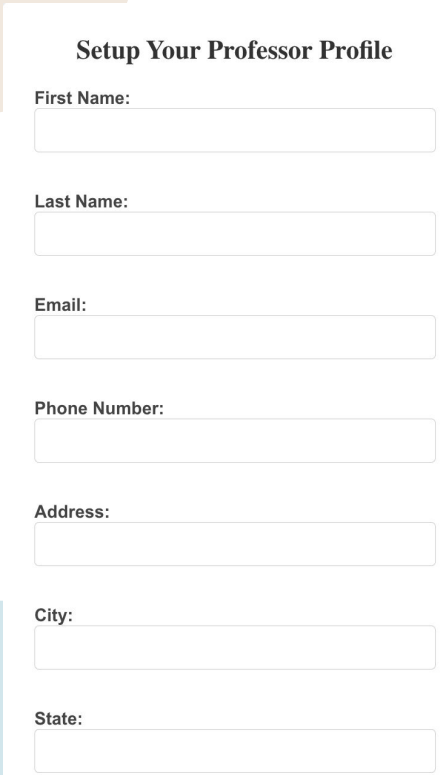
1. Courses
2. Profile

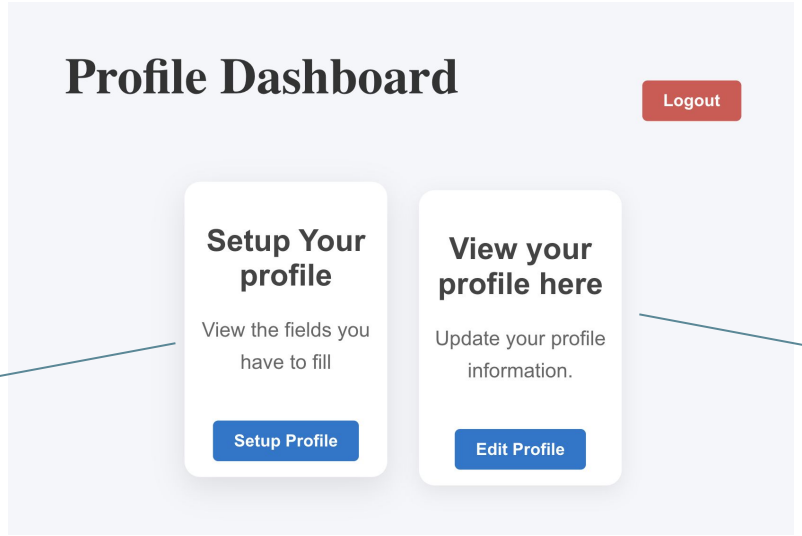


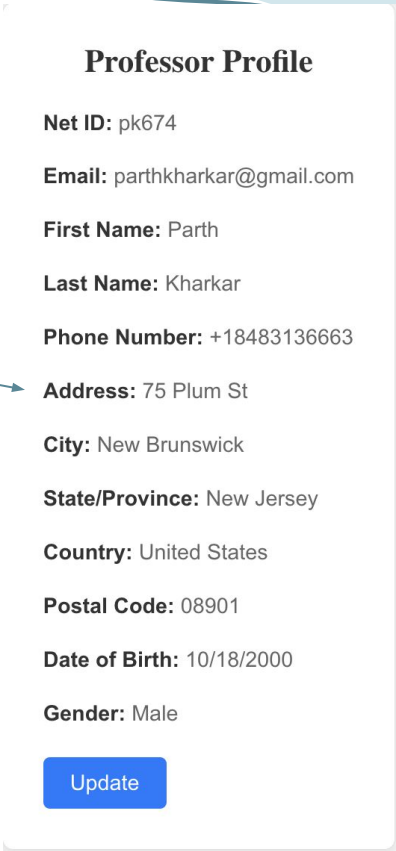
The Courses card contains two important features such as adding courses as well as viewing courses

The Profile card contains two main functionalities such as setting up the professor profile and updating profile

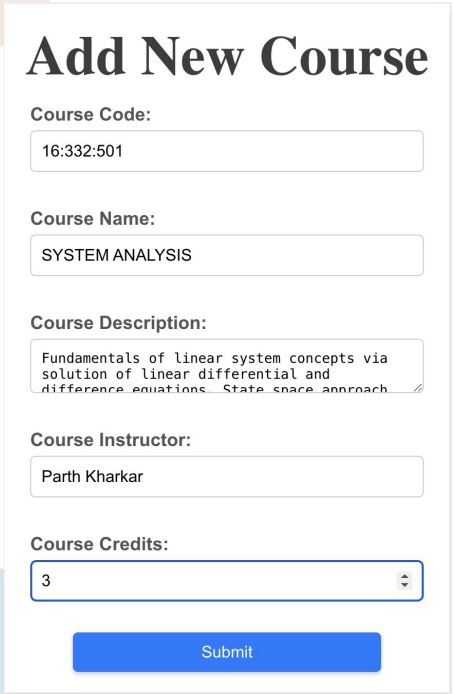
Professor Dashboard: Profile

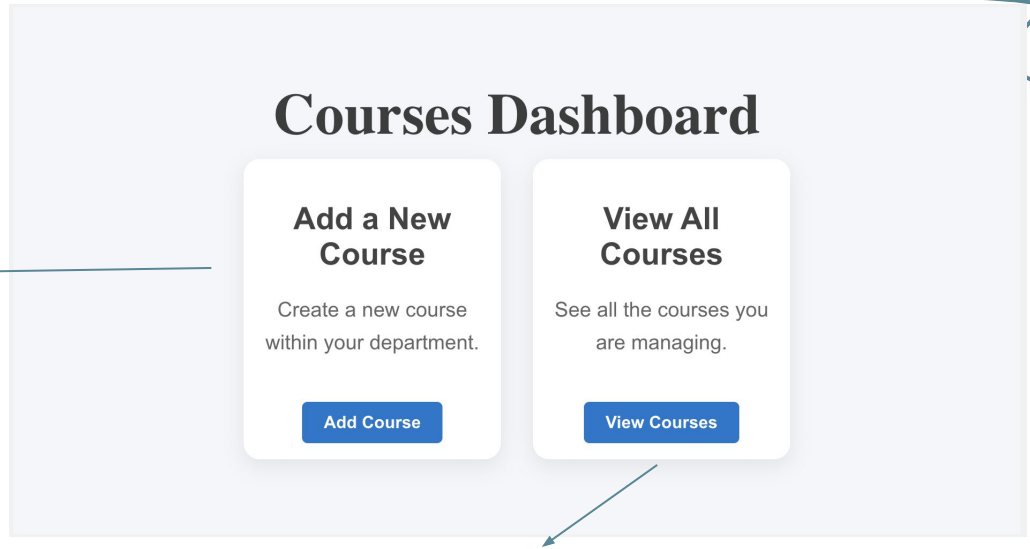


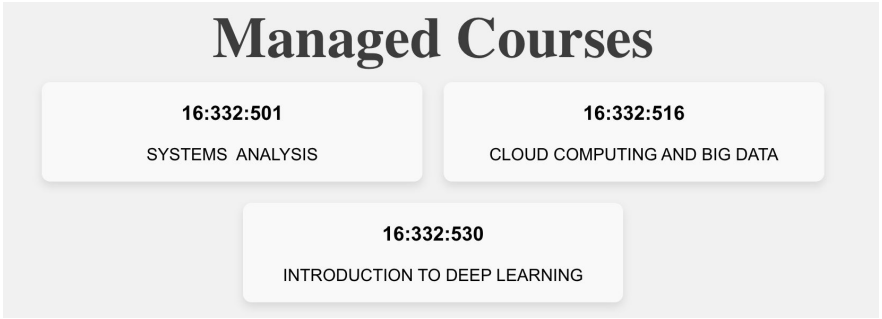




Professor Course Dashboard







Professor Course Feature Dashboard

Update Course

Course Code:

Course Name:

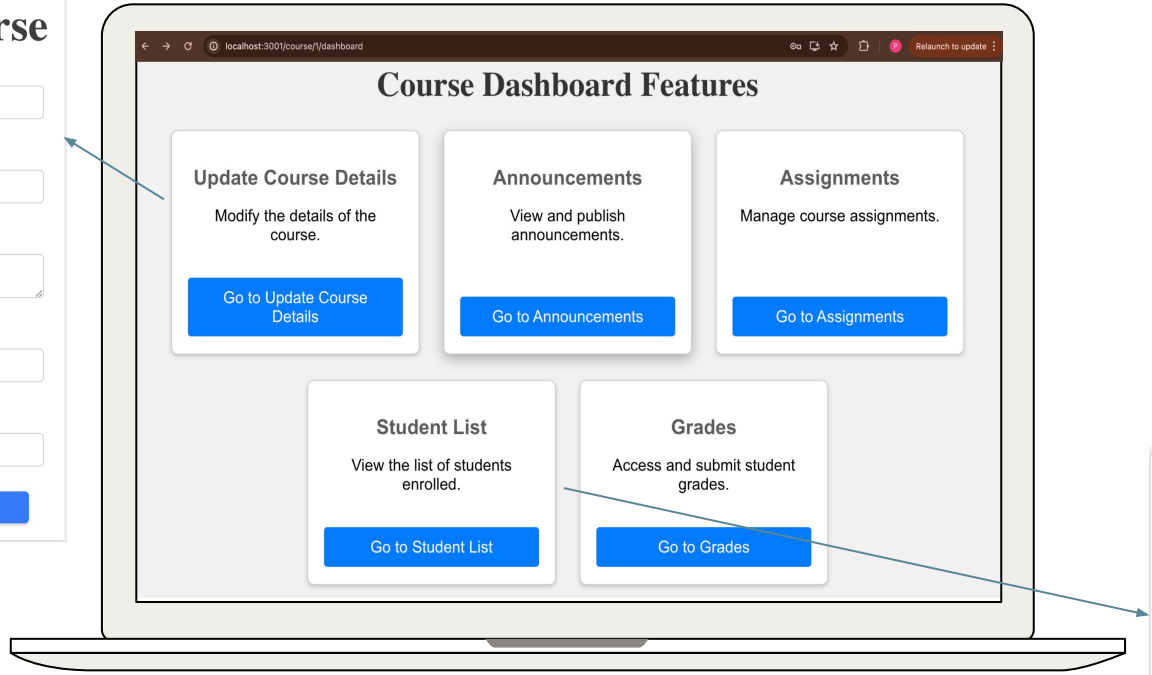
Course Description:

Course Instructor:

Course Credits:

0

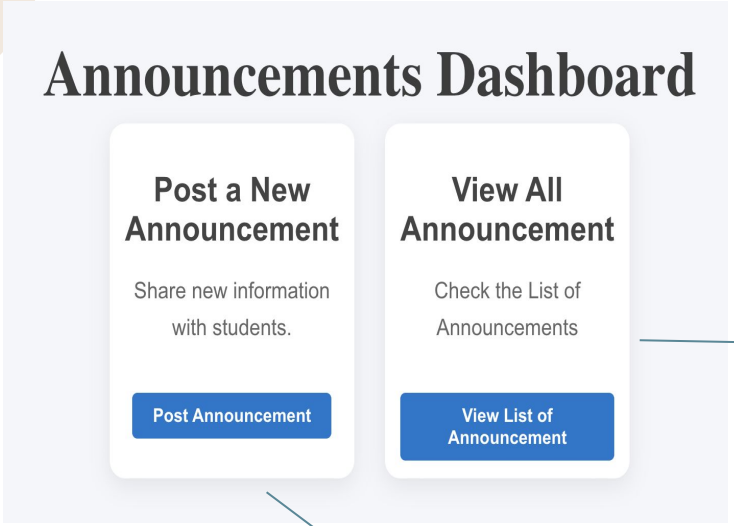
Update Course

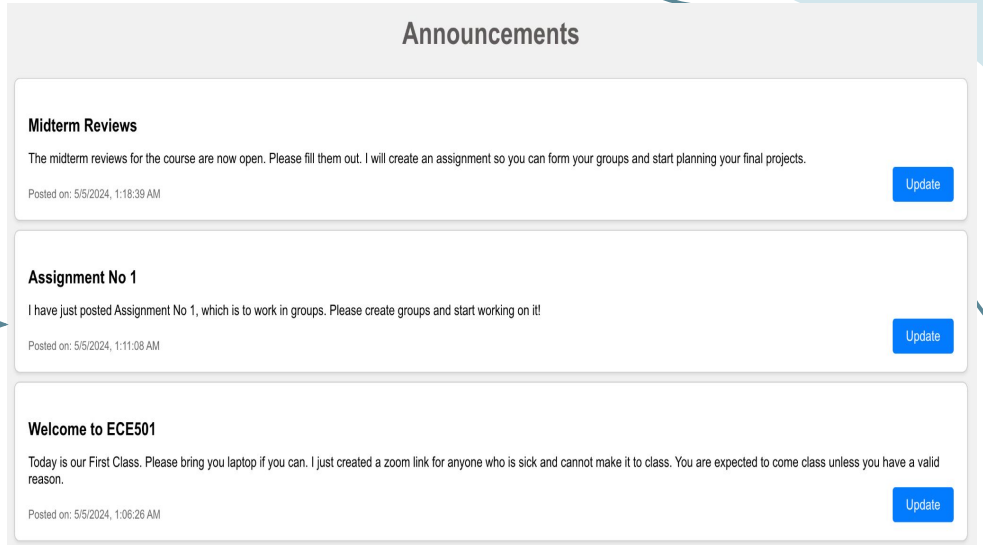


Students Enrolled in Course

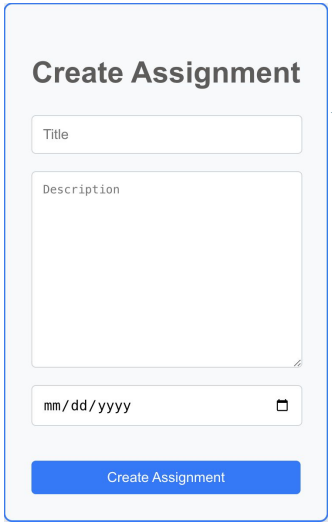
Atharva Bhusari	Student
Nikhil Mishra	Student
Jahnavi Shah	Student
Jash Shah	Student

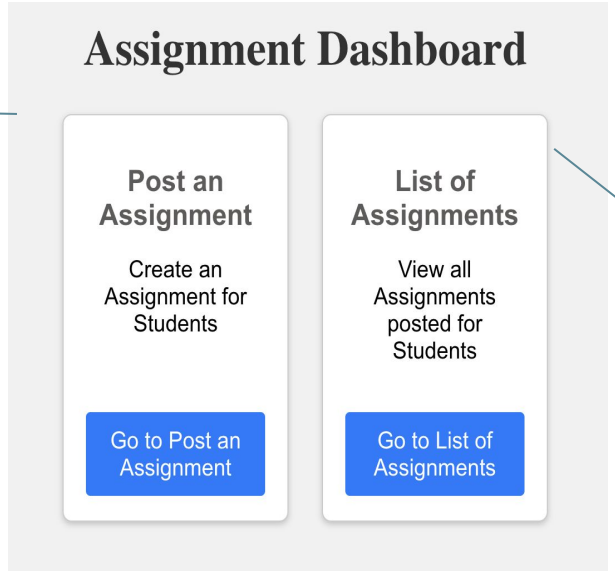
Course Feature: Announcements

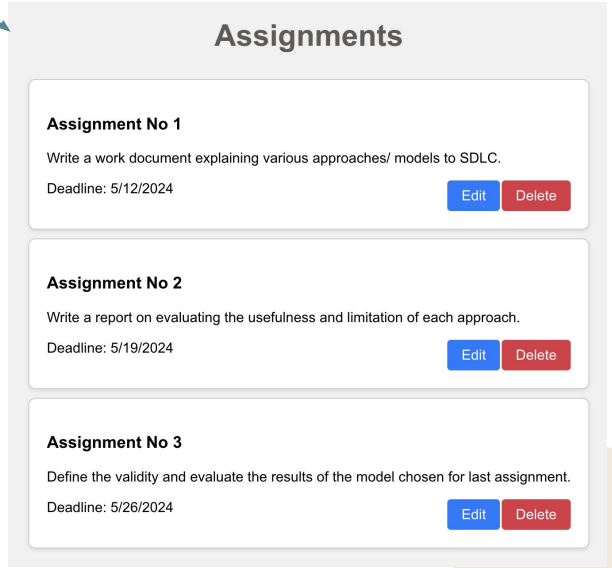




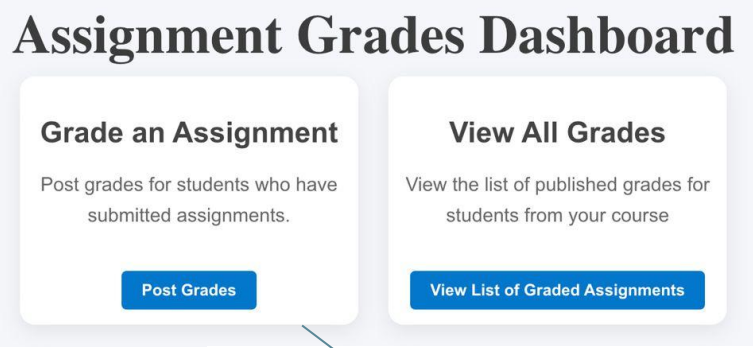
Course Feature: Assignments







Course Feature: Grades



List of Grades for Students per Assignment

Assignment Title	Student Name	Submission Time	Grade	Feedback
Assignment No 1	Atharva Bhusari	2024-05-05T23:35:37.000Z	90.00	Well done!
Assignment No 1	Nikhil Mishra	2024-05-06T00:43:43.000Z	100.00	Excellent Work!
Assignment No 2	Atharva Bhusari	2024-05-05T23:36:52.000Z	90.00	Good work, but can improve more!
Assignment No 3	Atharva Bhusari	2024-05-05T23:37:41.000Z		
Assignment No 3	Nikhil Mishra	2024-05-06T01:27:02.000Z		

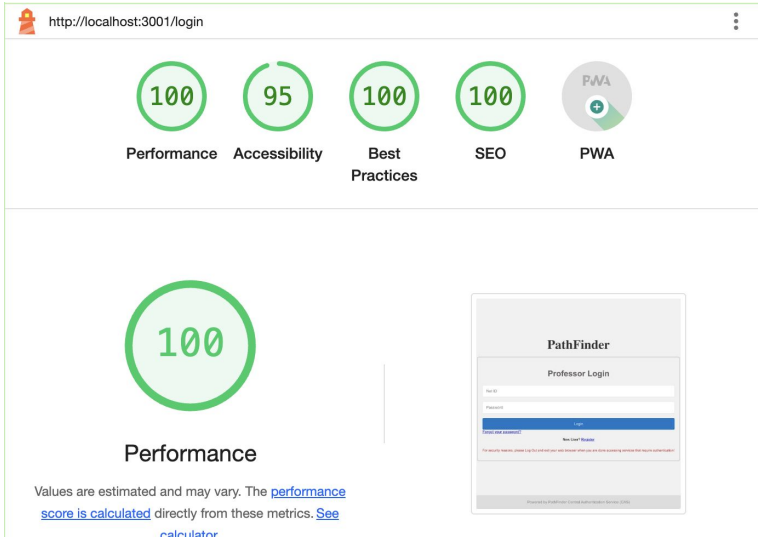
Assignment No 1

Student Name	Submission Time	Description	Action
Atharva Bhusari	May 5, 2024 at 07:35:37 PM	Software Development Life Cycle (SDLC) is a systematic process used by software development teams to design, develop, and test high-quality software.	Grade
Atharva Bhusari	May 5, 2024 at 08:43:03 PM	The SDLC provides a structured framework that enables the production of software that meets or exceeds customer expectations, is completed within time and cost estimates, and is efficient, reliable, and maintainable.	Grade
Nikhil Mishra	May 5, 2024 at 08:43:43 PM	The SDLC provides a structured framework that enables the production of software that meets or exceeds customer expectations, is completed within time and cost estimates, and is efficient, reliable, and maintainable.	Grade

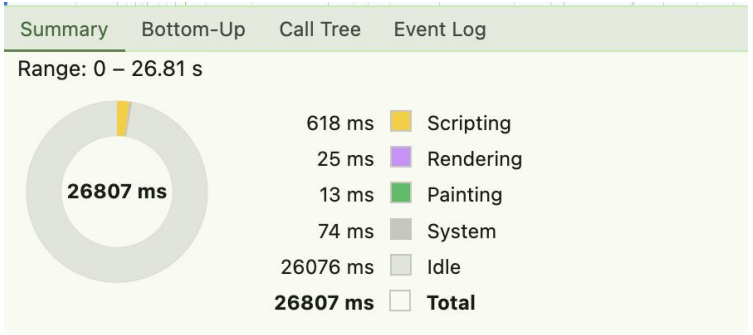
Assignment No 2

Student Name	Submission Time	Description	Action
Atharva Bhusari	May 5, 2024 at 07:36:52 PM	Software Development Life Cycle (SDLC) Models- Evaluation of Usefulness and Limitations: 1. Waterfall Model 2. V-Model	Grade

Performing Unit Test: Professor

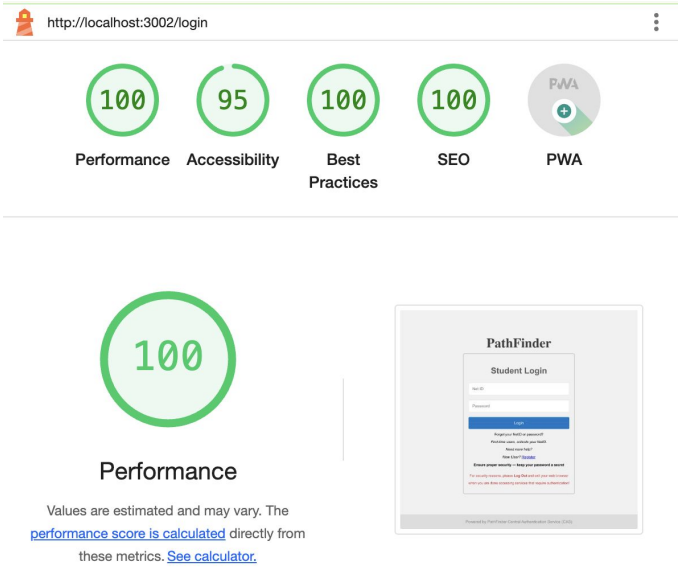


Lighthouse Performance Test for Professor Operations

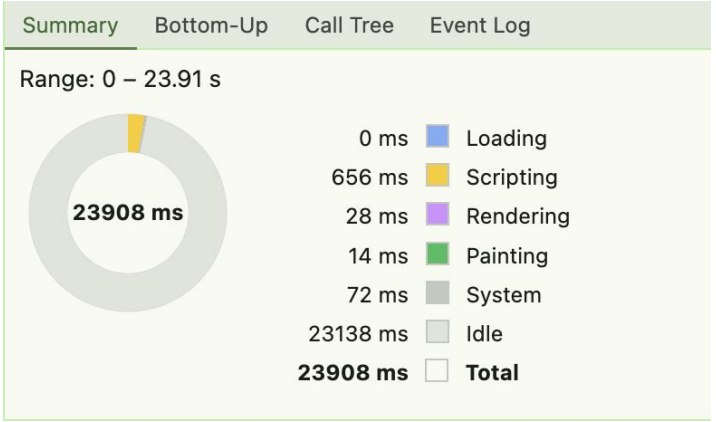


Performance Summary for Professor Operations

Performing Unit Test: Student



Lighthouse Performance Test for Student Operations



Performance Summary for Student Operations