Project Report On

ClassSpace
(Classroom Management System)

Submitted in partial fulfillment for the award of

Post Graduate Diploma in Advanced Computing

from

C-DAC ACTS (Pune)

Guided by

Mr. Abhilash Bhande

Presented By

Soham Patil 250840120191

Prasad Mahajan -250840120127

Durgesh Pagare-250840120053

Onkar Sonawane-250840120111

Arsh Shirsekar – 250840120172

Centre of Development of Advanced Computing (C-DAC), Pune

CERTIFICATE

TO WHOMSOEVER IT MAY CONCERN

This is to certify that

Soham Patil – 250840120191

Prasad Mahajan – 250840120127

Durgesh Pagare – 250840120053

Onkar Sonawane – 250840120111

Arsh Shirsekar - 250840120172

have successfully completed their project titled

"ClassSpace"

Under the Guidance of Mr.Abhilash Bhande

Project Guide **HOD ACTS**

ACKNOWLEDGEMENT

This project **"ClassSpace(Classroom Management System)"** was a great learning experience for us and we are submitting this work to Advanced Computing Training School (CDAC ACTS).

We all are very glad to mention the name of **Mr. Abhilash Bhande** for his valuable guidance to work on this project. His guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

Our most heartfelt thank goes to Ms **Swati mam** (Course Coordinator, PG- DAC) who gave all the required support and kind coordination to provide all the necessities like required hardware, internet facility and extra Lab hours to complete the project and throughout the course up to the last day here in C-DAC ACTS, Pune.

Soham Patil - 250840120191
Prasad Mahajan – 250840120127
Durgesh Pagare - 250840120053
Onkar Sonawane – 250840120111
Arsh Shirsekar -250840120172

TABLE OF CONTENTS

1. Introduction

ClassSpace is a secure, timetable-driven classroom management system designed to streamline academic operations and enhance interaction between students, teachers, and coordinators. The system aims to digitize and automate essential classroom activities such as class creation, timetable management, lecture handling, attendance tracking, feedback collection, and announcement dissemination within a unified platform. By providing role-based access control, ClassSpace ensures that each user interacts with the system according to their responsibilities, thereby improving efficiency, transparency, and accountability in academic environments.

The backend of ClassSpace is developed using **Spring Boot** and **RESTful web services**, enabling scalable and secure server-side processing. User authentication and authorization are implemented using **JWT-based security mechanisms**, ensuring safe access to system resources. The frontend is built using **React**, offering a dynamic and responsive user interface that allows seamless interaction across devices. Data persistence is managed using **MySQL**, which stores structured information related to users, roles, classes, timetables, lectures, attendance records, feedback, and integrity metrics.

A key feature of ClassSpace is its **lecture-level attendance and integrity evaluation system**, which records both declared and actual attendance and computes an integrity score for students. This mechanism promotes honesty and accountability while providing teachers with reliable attendance analytics. Additionally, the platform supports real-time announcements, structured note sharing, and lecture feedback with star ratings, enabling continuous improvement in teaching and learning processes. With its modular architecture and scalable design, ClassSpace provides a robust foundation for modern, technology-driven classroom management systems.

2. Software/Hardware Requirement

Server:

Processor: Intel Core i5 or equivalent AMD processor.

RAM: Minimum 8GB RAM.

Storage: SSD storage for improved performance. Network:

Ethernet or Wi-Fi connectivity.

Operating System: Linux distribution (Ubuntu, CentOS) preferred for server deployment.

Client Devices:

Processor: Dual-core processor or higher.

RAM: Minimum 4GB RAM.

Storage: Sufficient storage for caching and local data.

Network: Ethernet or Wi-Fi connectivity.

Browser: Compatible with latest versions of popular browsers like Google Chrome, Mozilla Firefox, and Safari.

3. Tools and technologies used

- Java
- SpringBoot
- RESTful Web Services
- Spring Security
- JWT(JSON web Token)
- SpringWeb
- HTML and CSS
- React JS
- MySQL
- Git

1. Java: Java is used as the core programming language for developing the backend of ClassSpace. It provides platform independence, robustness, and strong object-oriented features, making it suitable for building scalable academic management systems.

2. SpringBoot: Spring Boot is used to develop the backend application and manage business logic. It simplifies application configuration and enables rapid development of RESTful services for handling users, classes, timetables, lectures, and attendance operations.

3. RESTful Web Services: RESTful APIs are used to enable communication between the frontend and backend. These services handle operations such as login, class management, lecture handling, attendance submission, feedback collection, and announcement retrieval in a stateless manner.

4. JWT(JSON web token): JWT is used for secure user authentication and session management. After successful login, a token is generated and validated for each request, ensuring secure and efficient access without maintaining server-side sessions.

5. Spring Security: RESTful APIs are used to enable communication between the frontend and backend. These services handle operations such as login, class management, lecture handling, attendance submission, feedback collection, and announcement retrieval in a stateless manner.

6. HTML, CSS and JavaScript: These standard web technologies are used for structuring, styling, and adding interactivity to the frontend components. They help in creating a clean and intuitive user interface

7. React JS: React JS is used to build the frontend of ClassSpace. Its component-based architecture enables the development of a dynamic, responsive, and user-friendly interface for students, teachers, and coordinators

8. MySQL: MySQL is used as the relational database management system to store structured data such as user details, roles, classes, timetables, lectures, attendance records, feedback, integrity scores, and announcements.

9. Git : Git is used as a version control system to manage source code, track changes, and support collaborative development among team members

4. Project E-R(Entity relationship) Diagram

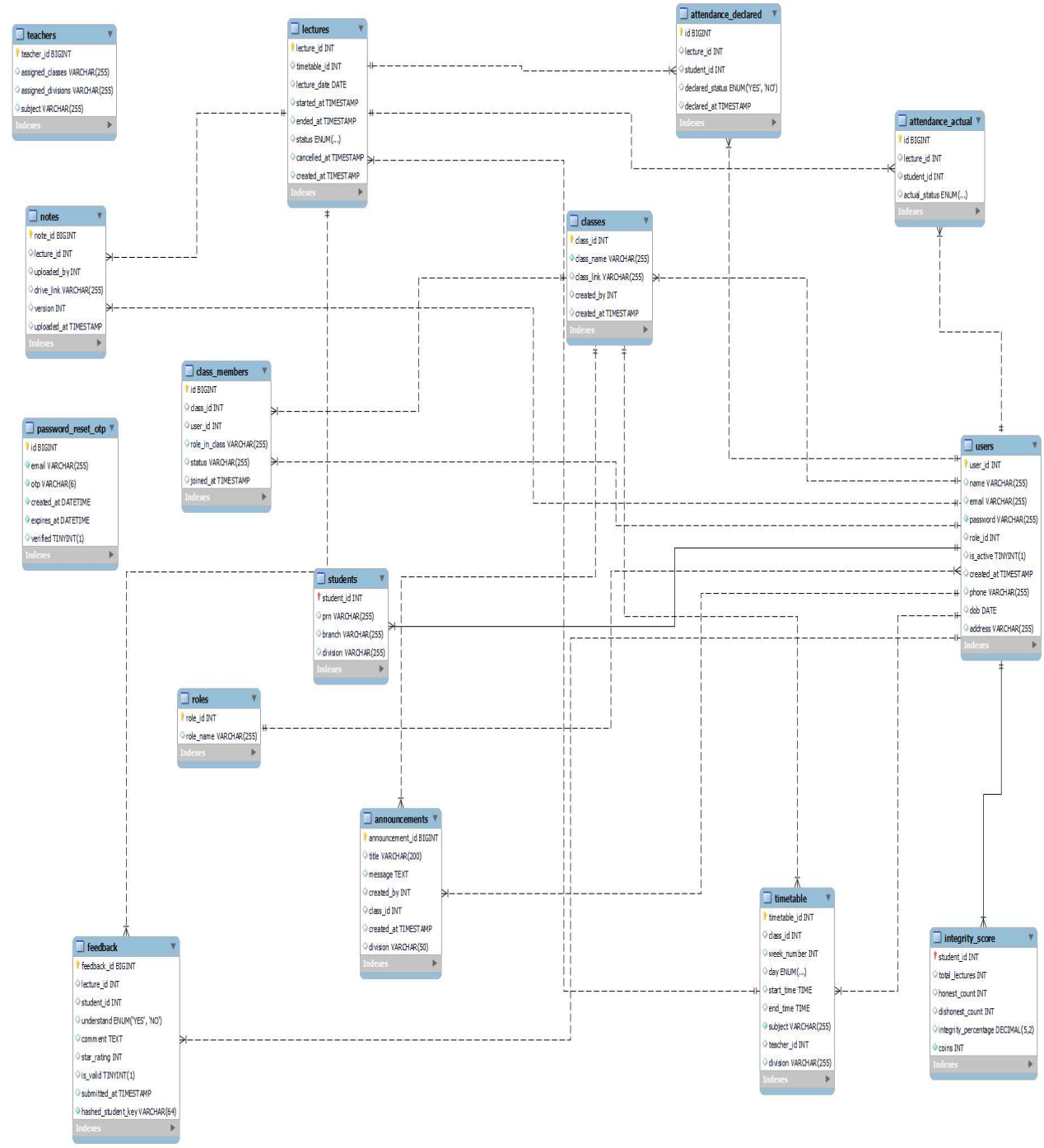


Fig.5.1 ClassSpace

5. Advantages

- **Java**
  Java provides platform independence, allowing the application to run across different operating systems without modification. Its object-oriented nature improves code reusability and maintainability, while its robustness and reliability make it suitable for enterprise-level applications such as ClassSpace.

- **Spring Boot**
  Spring Boot simplifies backend development by minimizing configuration and boilerplate code. It supports rapid development of RESTful services and provides scalability, making it suitable for handling classroom management operations efficiently.

- **Spring Security**
  Spring Security ensures secure authentication and authorization within the system. It enables role-based access control, allowing coordinators, teachers, and students to access only their permitted functionalities, thereby enhancing system security.

- **RESTful Web Services**
  RESTful web services enable smooth communication between the frontend and backend components of the application. The stateless nature of REST improves system scalability and allows easy integration with different client platforms.

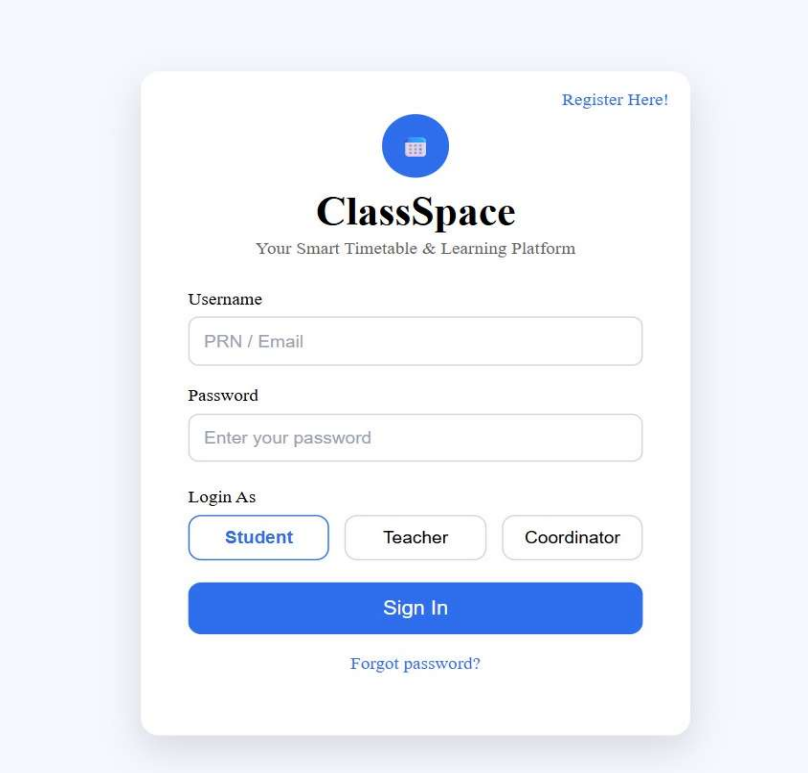
- **JWT (JSON Web Token)**
  JWT enables secure and stateless user authentication by generating tokens after successful login. This reduces server-side session management overhead and improves performance while maintaining data security.

- **React JS**
  React JS provides a component-based architecture that supports the development of dynamic and responsive user interfaces. Its virtual DOM enhances performance, ensuring smooth interaction for users accessing ClassSpace.

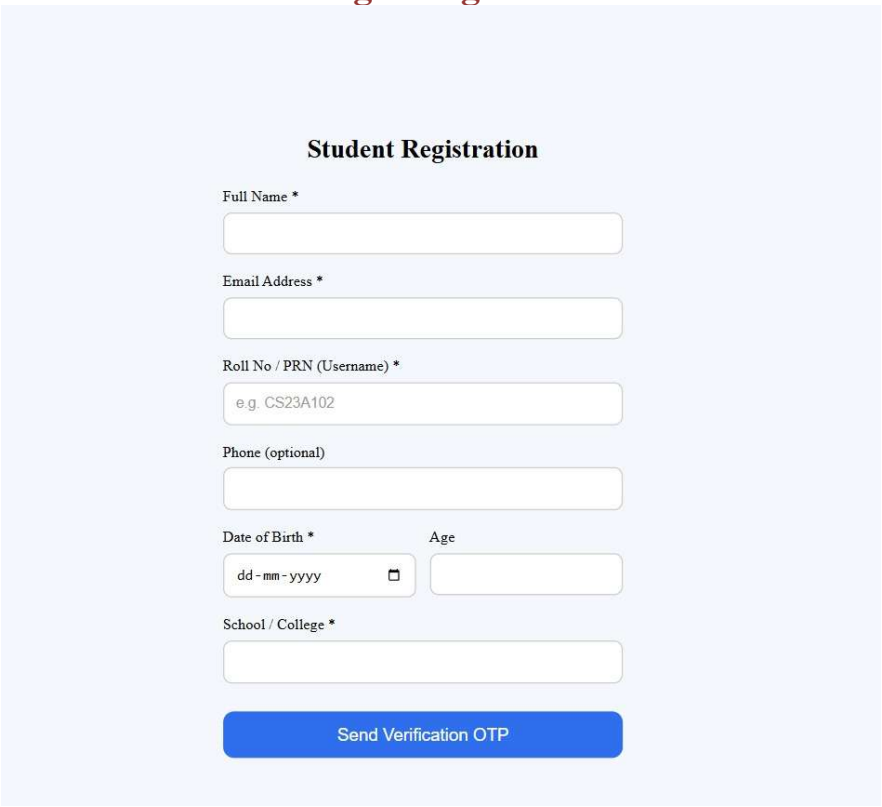
- **MySQL**
  MySQL provides reliable relational data storage with strong support for data integrity and consistency. It efficiently manages structured academic data such as users, attendance records, timetables, and feedback.

- **Git**
  Git is used for version control, enabling efficient code tracking and collaboration among developers. It helps maintain project history, manage updates, and reduce development risks.

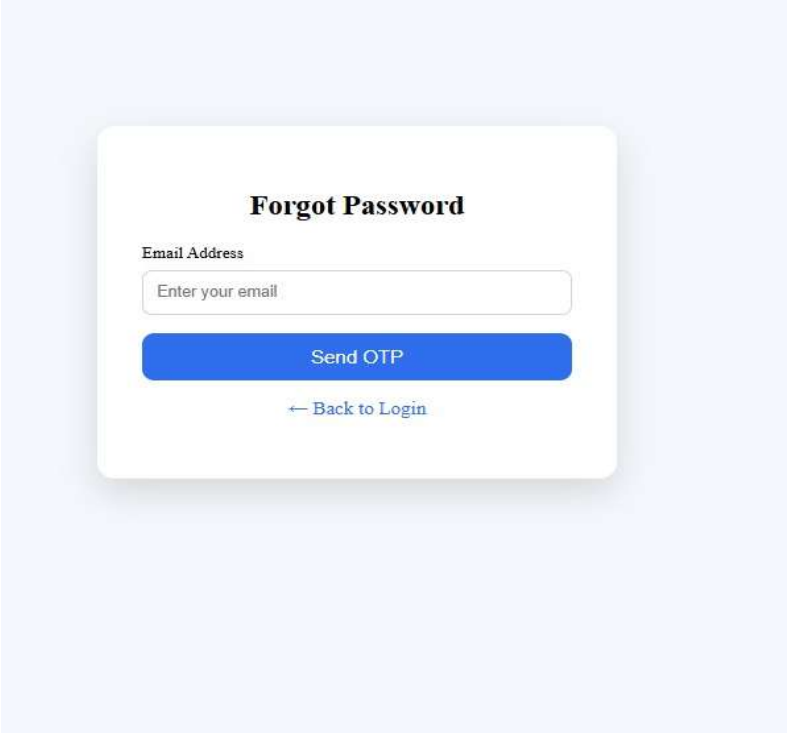
6. Screenshots



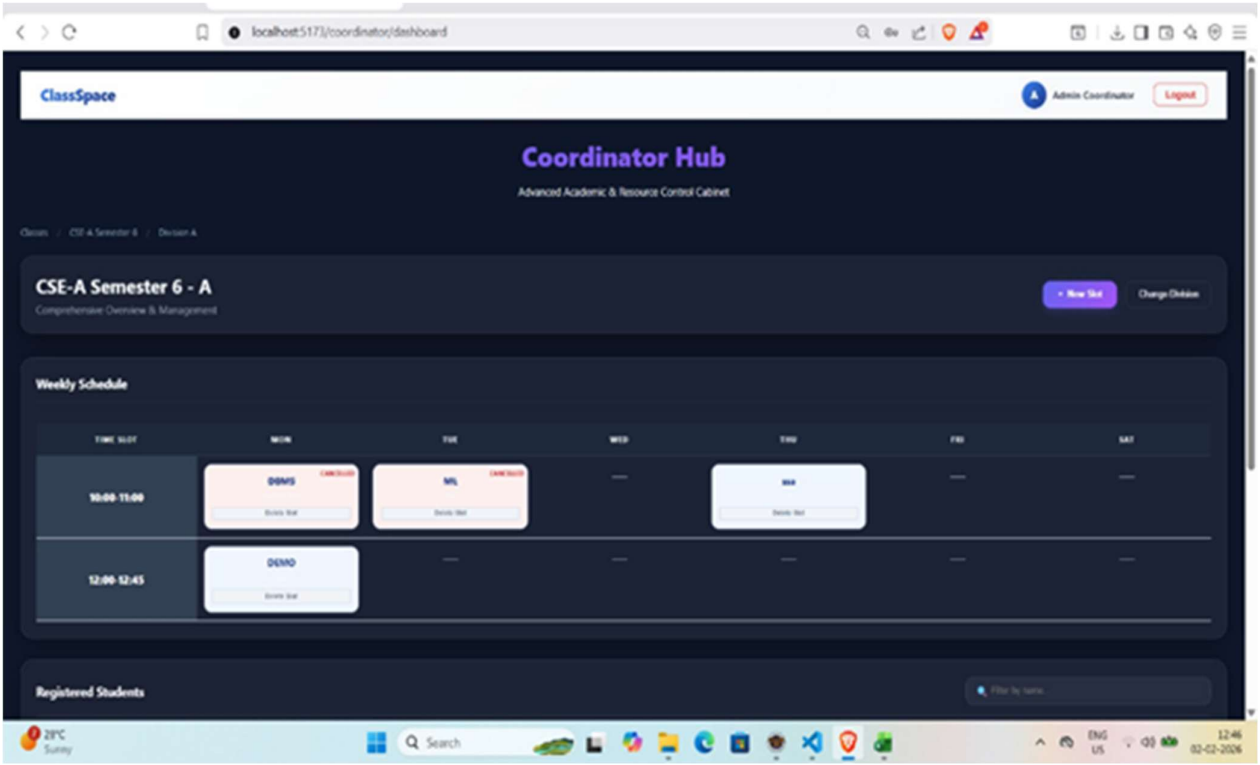
Login Page



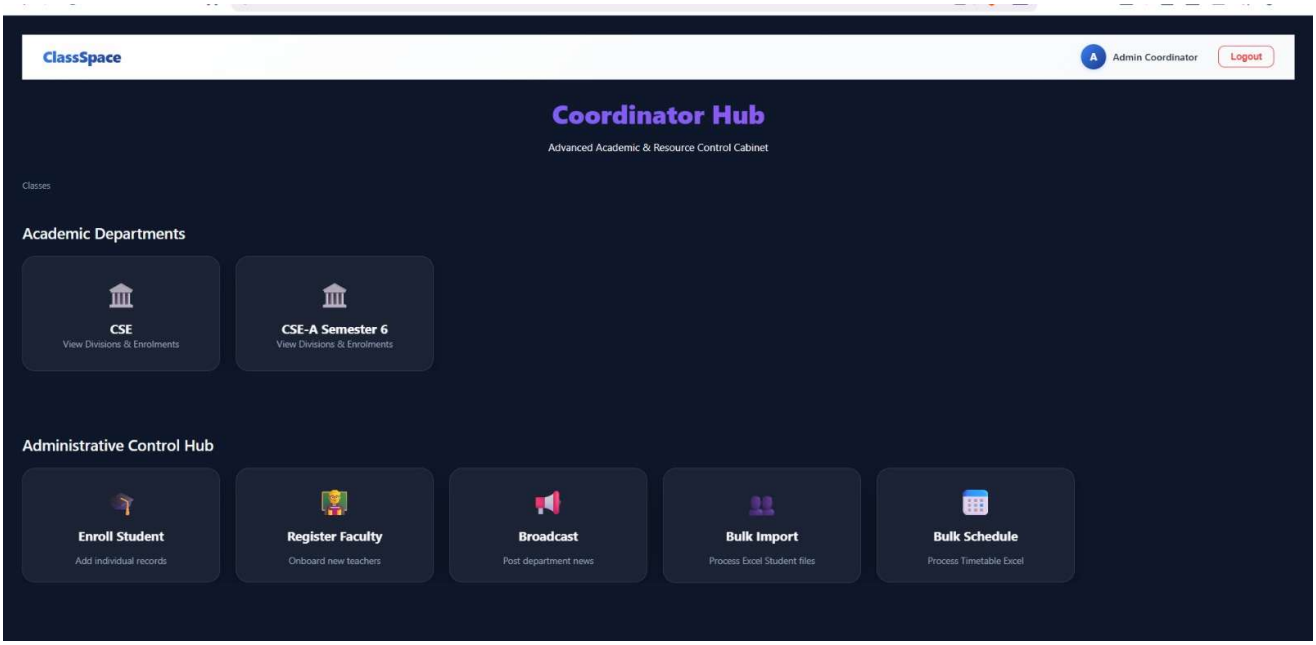
Register Page



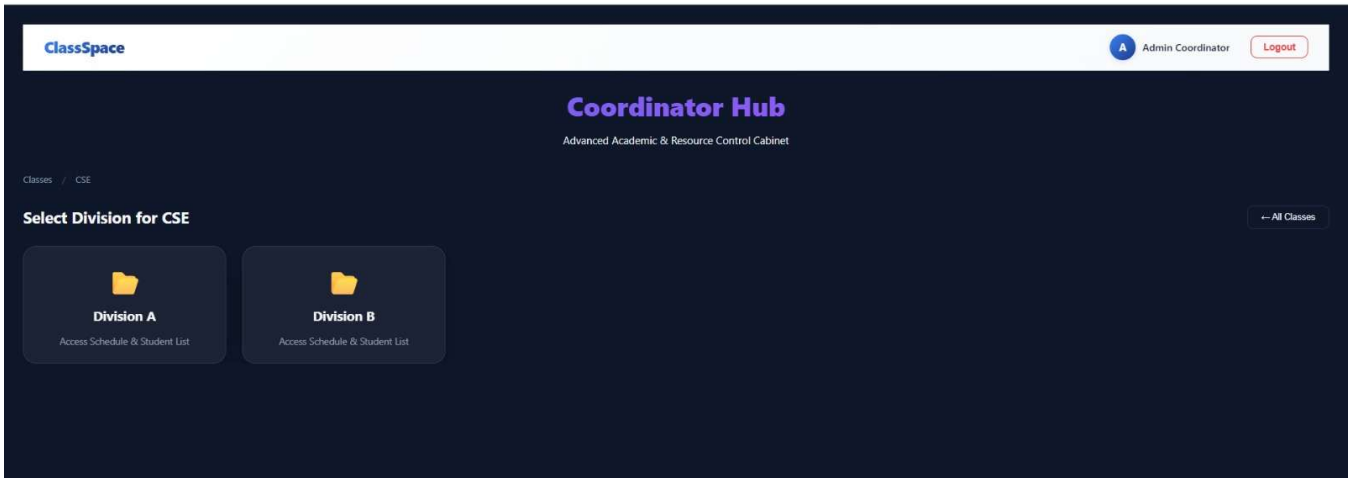
Forgot Password Page



Coordinator Dashboard Page

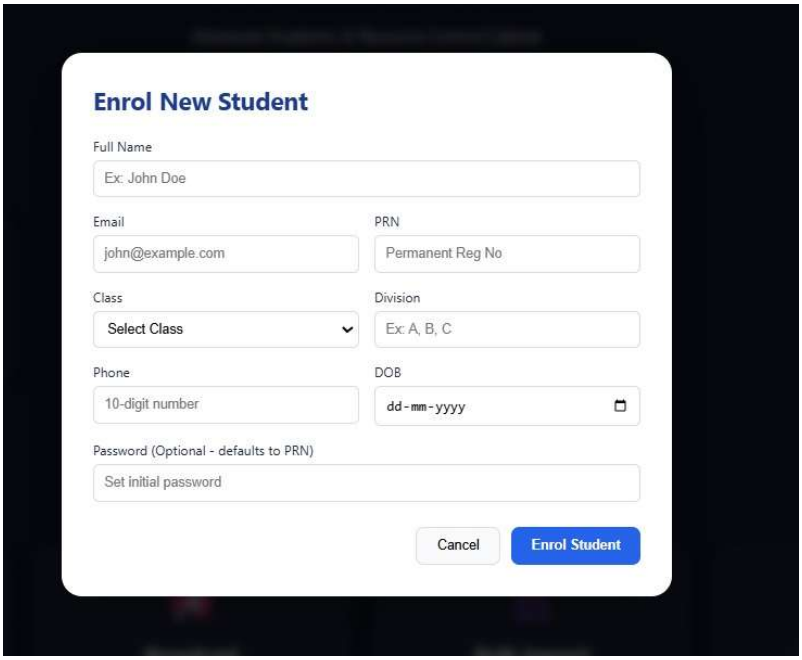


Coordinator Dashboard Page

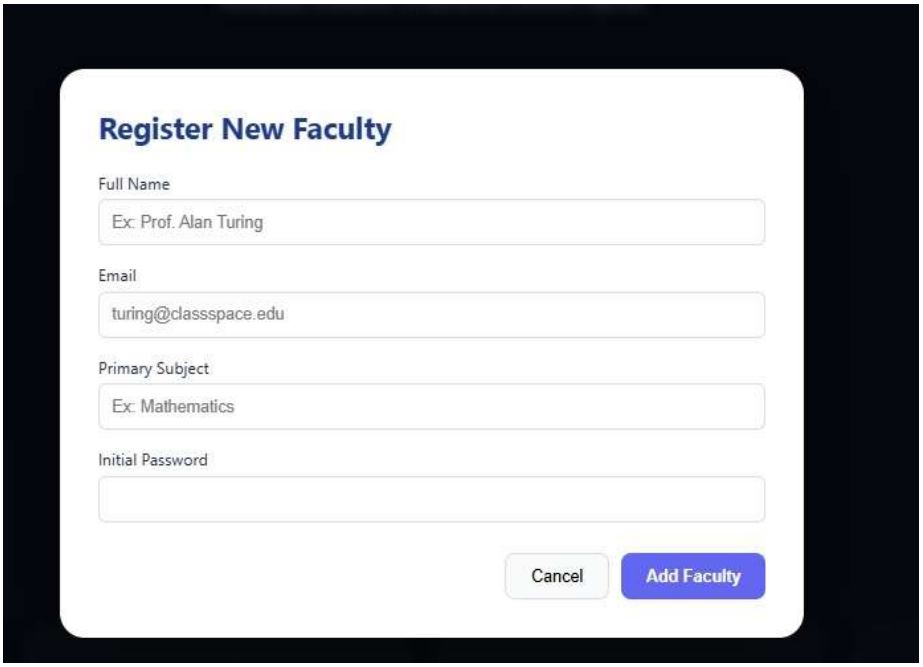


Coordinator Dashboard Page

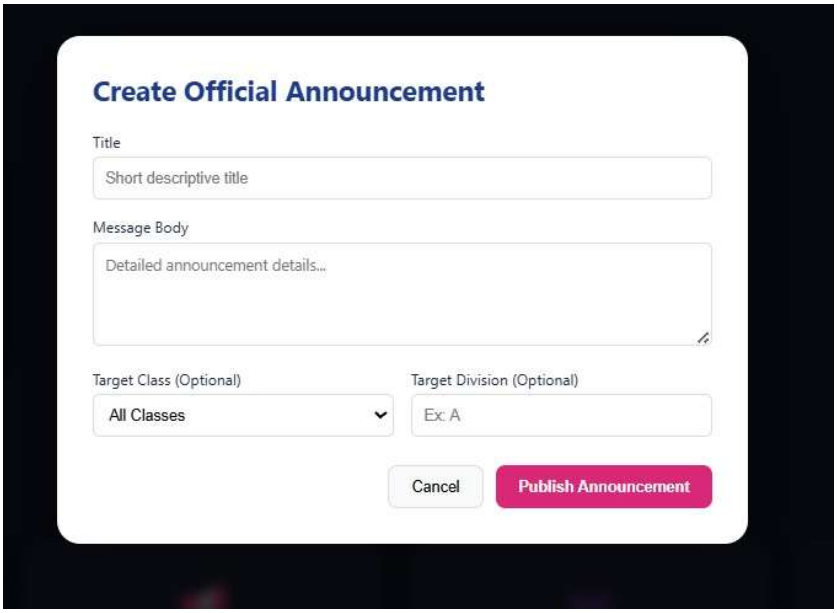
Coordinator Dashboard Page



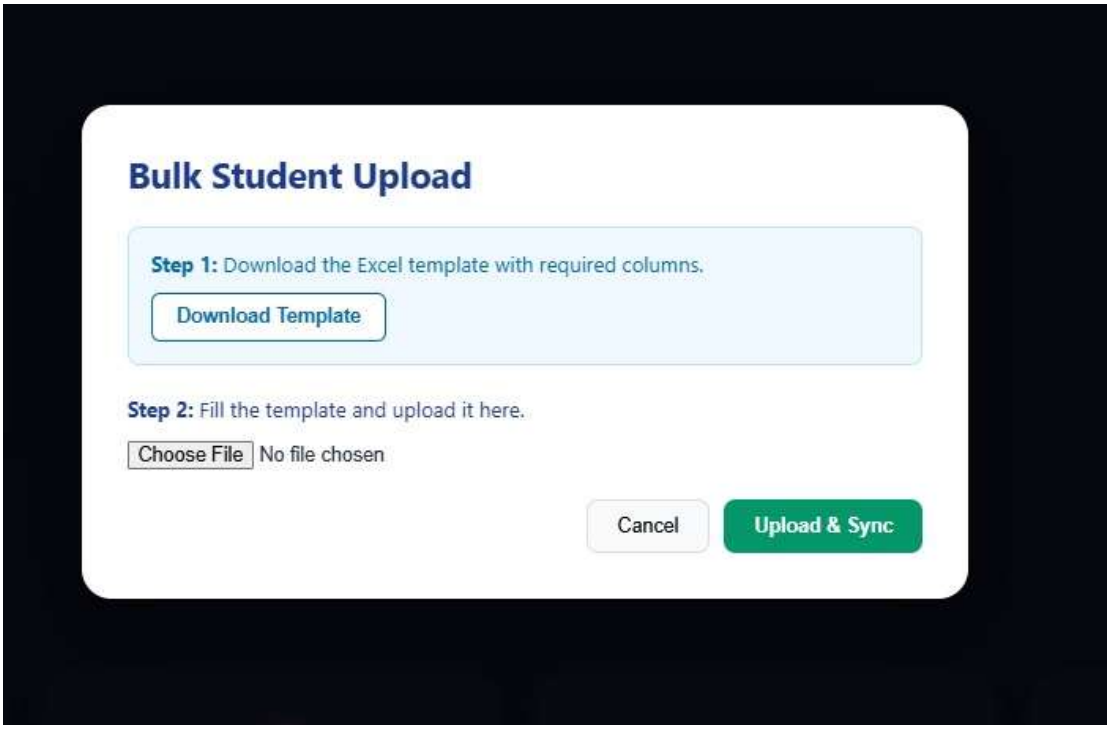
Register New Student



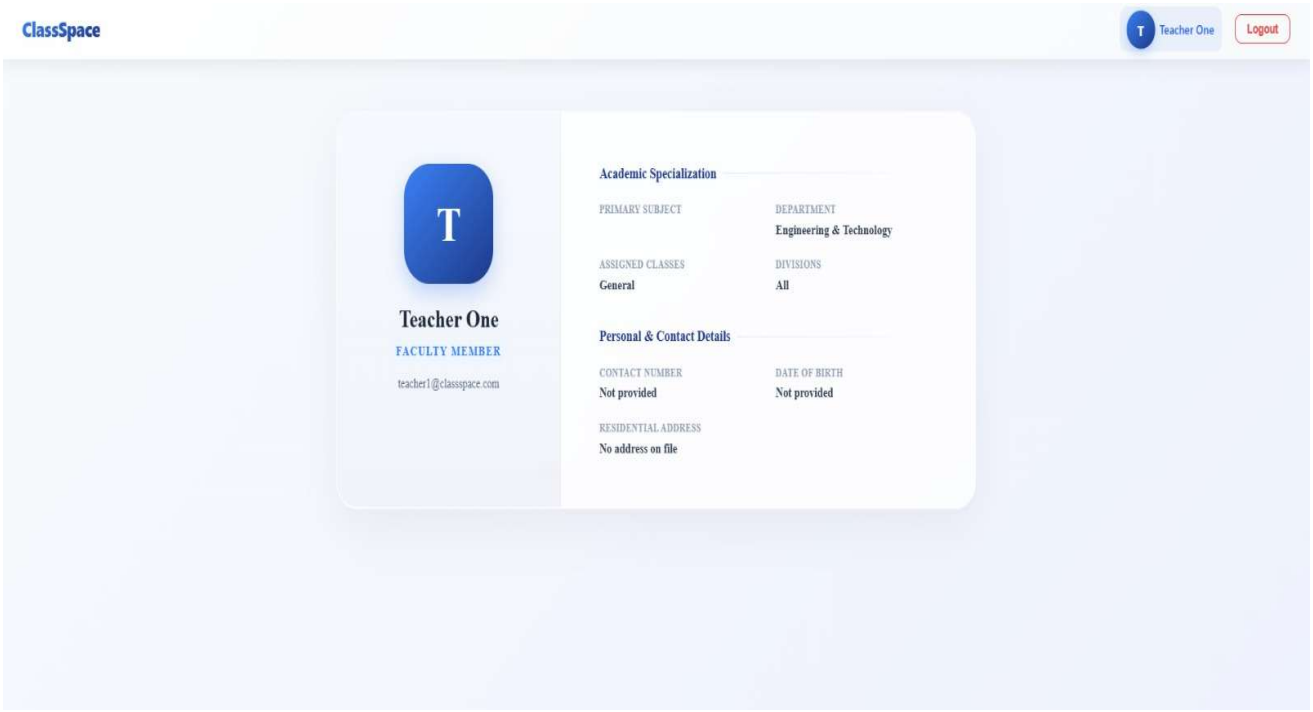
Register New Faculty



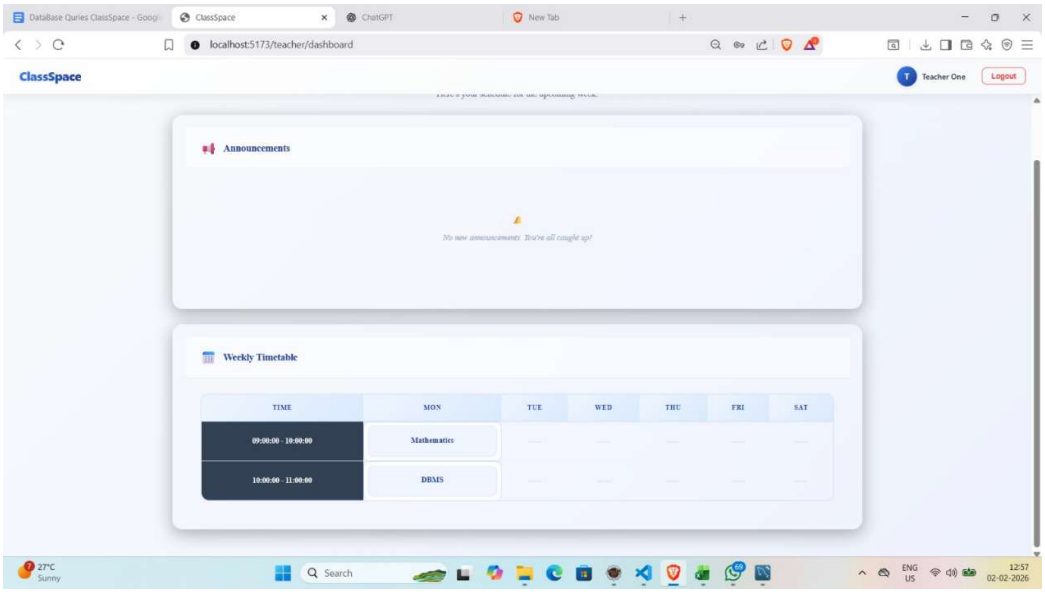
Create Official Announcement



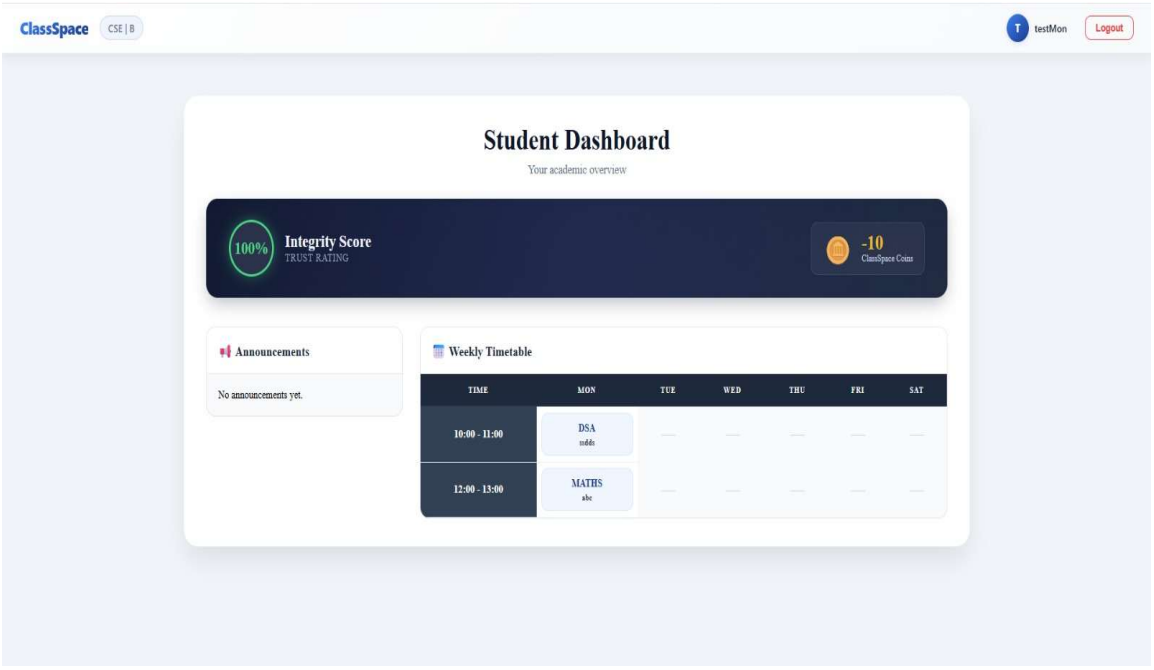
Bulk Student Upload



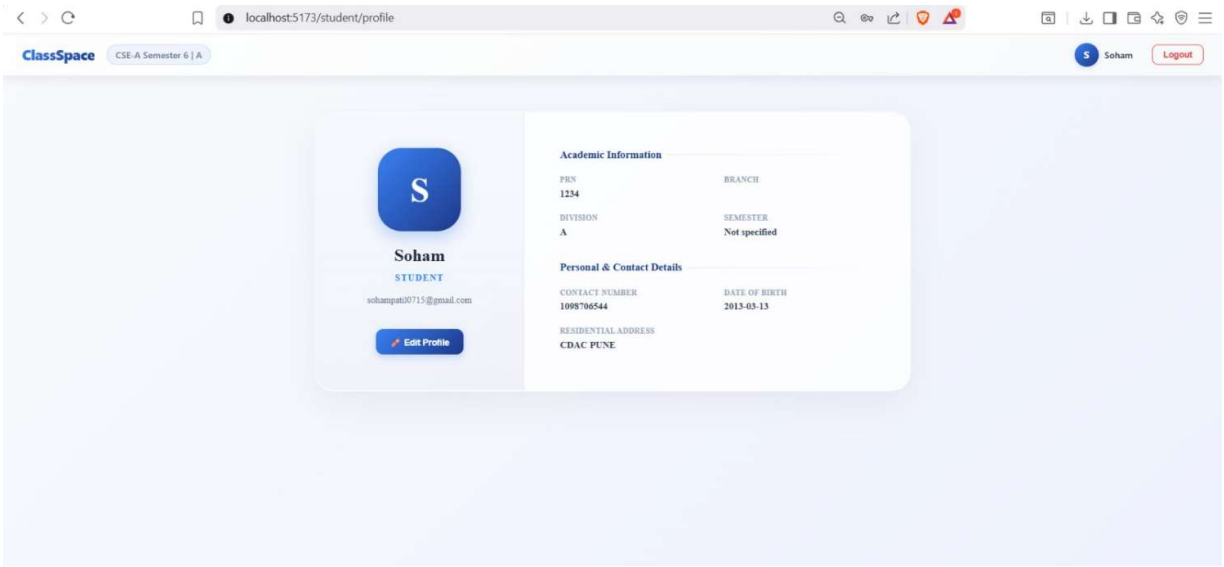
Teacher Profile



Teacher DashBoard

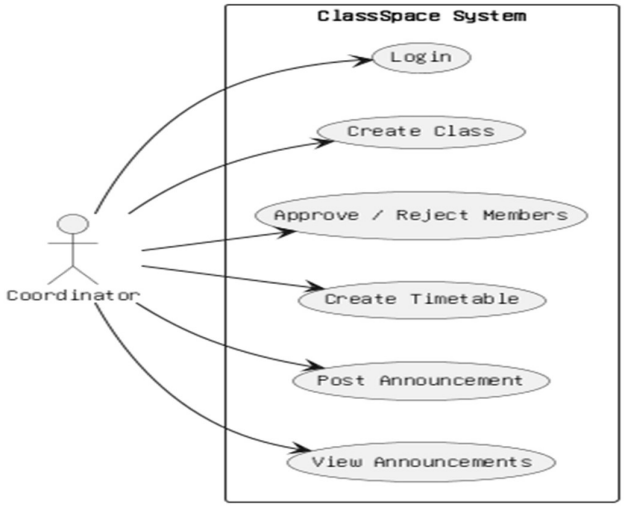


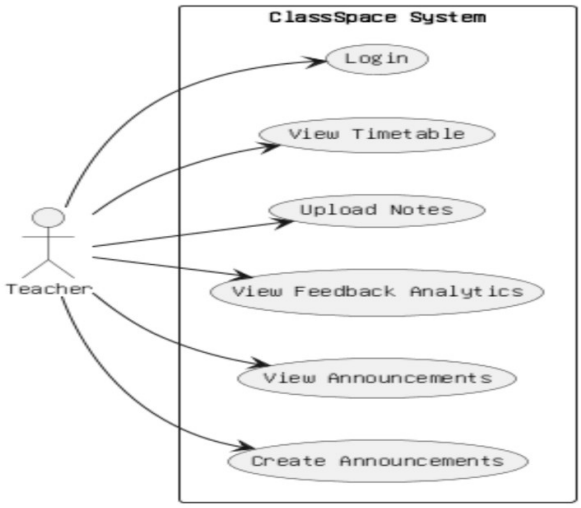
Student Dashboard

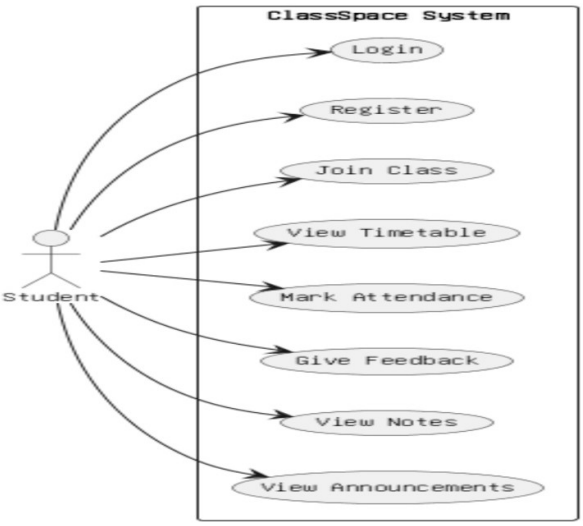


Student Profile

A) Use Case Diagram







FUTURE SCOPE:

- Mobile Application Development
  The system can be extended by developing dedicated mobile applications for Android and iOS platforms. This will improve accessibility and allow students and teachers to interact with ClassSpace anytime and anywhere.

- Advanced Analytics and Reporting
  Future enhancements can include advanced analytics dashboards for coordinators and teachers, providing insights into attendance trends, student performance, feedback patterns, and lecture effectiveness.

- AI-Based Attendance and Behavior Analysis
  Artificial intelligence techniques can be integrated to analyze attendance behavior and predict irregular patterns. This can help institutions identify at-risk students and take proactive measures.

- Real-Time Notifications and Alerts
  Push notifications and alert mechanisms can be introduced to inform users about class updates, timetable changes, announcements, and attendance status in real time.

- Cloud Deployment and Scalability
  Migrating the application to cloud platforms can improve scalability, availability, and performance. Cloud-based deployment will also support load balancing and automated backups.

- Enhanced Security and Compliance
  Additional security features such as multi-factor authentication and compliance with institutional data protection policies can be implemented to further strengthen system security.

7. Conclusion

The ClassSpace project successfully delivers a secure and role-based classroom management system that addresses the challenges of traditional academic coordination and monitoring. By integrating class creation, timetable management, lecture handling, attendance tracking, feedback collection, and announcement dissemination into a single platform, the system enhances efficiency, transparency, and accountability within educational institutions.

The application leverages modern technologies such as Spring Boot, RESTful web services, JWT-based security, React, and MySQL to provide a scalable, reliable, and user-friendly solution. Role-based access control ensures that coordinators, teachers, and students interact with the system according to their responsibilities, while the structured database design maintains data integrity and consistency. The inclusion of lecture-level attendance verification and integrity scoring further strengthens the reliability of attendance records and promotes honest participation.

Overall, ClassSpace demonstrates an effective application of software engineering principles and modern web technologies to solve real-world academic management problems. The system is designed with extensibility in mind, allowing future enhancements such as mobile application support, advanced analytics, and AI-based insights. With its modular architecture and scalable design, ClassSpace serves as a robust foundation for next-generation digital classroom management systems.

8. References

1. https://spring.io/projects/spring-boot
2. https://spring.io/projects/spring-data-jpa
3. https://restfulapi.net/
4. https://www.mysql.com/
5. https://spring.io/projects/spring-web
6. https://reactjs.org/