A
Project Report
On
College Connect: An Exclusive Social Media Platform

Prepared by
VEER PATEL (23DCS097)
FENIL SAVALIYA (23DCS115)
VASU VAGHASIA (23DCS138)

Under the guidance of
Dr. Parth Goel
Mrs. Kirti Makavana

Submitted to
Charotar University of Science & Technology
Degree of Bachelor of Technology
in Computer Science & Engineering
CSE204: Project-II
of 4rd Semester of B.Tech

Submitted at

अमृतं तु विद्या
CHARUSAT®
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
Faculty of Technology & Engineering, CHARUSAT
Devang Patel Institute of Advance Technology and Research (DEPSTAR)

At: Changa, Dist: Anand – 388421
APPRIL 2025

CERTIFICATE

This is to certify that the report entitled "**College Connect: An Exclusive Social Media** " is a bonafied work carried out by **Veer Patel (23DCS097)** under the guidance and supervision of **Dr. Parth Goel** for the subject **Project-II (CSE204)** of 4rd Semester of Bachelor of Technology in **Computer Science & Engineering** at Devang Patel Institute of Advance Technology and Research (DEPSTAR), Faculty of Technology & Engineering (FTE) – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Under the supervision of,

Dr. Parth Goel
Assistant Professor
Dept.of Computer Science & Engineering
DEPSTAR, CHARUSAT-Changa.

Dr. Chirag Patel
Head of Department,
Dept. of Computer Science & Engineering
DEPSTAR, CHARUSAT-Changa.

Devang Patel Institute of Advance Technology and Research (DEPSTAR)
Faculty of Technology & Engineering, CHARUSAT
At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat

CERTIFICATE

This is to certify that the report entitled "**College Connect: An Exclusive Social Media** " is a bonafied work carried out by **Savaliya Fenil(23DCS115)** under the guidance and supervision of **Dr. Parth Goel** for the subject **Project-II (CSE204)** of 4rd Semester of Bachelor of Technology in **Computer Science & Engineering** at Devang Patel Institute of Advance Technology and Research (DEPSTAR), Faculty of Technology & Engineering (FTE) – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Under the supervision of,

Dr. Parth Goel
Assistant Professor
Dept.of Computer Science & Engineering
DEPSTAR, CHARUSAT-Changa.

Dr. Chirag Patel
Head of Department,
Dept. of Computer Science & Engineering
DEPSTAR, CHARUSAT-Changa.

Devang Patel Institute of Advance Technology and Research (DEPSTAR)
Faculty of Technology & Engineering, CHARUSAT
At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CERTIFICATE

This is to certify that the report entitled "**College Connect: An Exclusive Social Media Platform** " is a bonafied work carried out by **Vasu vaghasia(23DCS138)** under the guidance and supervision of **Prof. Kirti Makavana** for the subject **Project-II (CSE204)** of 4rd Semester of Bachelor of Technology in **Computer Science & Engineering** at Devang Patel Institute of Advance Technology and Research (DEPSTAR), Faculty of Technology & Engineering (FTE) – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Under the supervision of,

Prof. Kirti Makavana
Assistant Professor
Dept. of Computer Science & Engineering
DEPSTAR, CHARUSAT-Changa.

Dr. Chirag Patel
Head of Department,
Dept. of Computer Science & Engineering
DEPSTAR, CHARUSAT-Changa.

Devang Patel Institute of Advance Technology and Research (DEPSTAR)
Faculty of Technology & Engineering, CHARUSAT
At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat

DECLARATION BY THE CANDIDATE/CANDIDATES

We/I hereby declare that the project report entitled "**College Connect: An Exclusive Social Media** " submitted by us/me to Devang Patel Institute of Advance Technology and Research (DEPSTAR), Changa in partial fulfilment of the requirements for the award of the degree of **B.Tech Computer Science & Engineering**, from the Department of Computer Science & Engineering, DEPSTAR, FTE is a record of bonafide CSE204 Project-I carried out by us/me under the guidance of **Dr. Parth Goel**. We/I further declare that the work carried out and documented in this project report has not been submitted anywhere else either in part or in full and it is the original work, for the award of any other degree or diploma in this institute or any other institute or university.

Signature of the candidate
(Veer Patel – 23DCS097)

Signature of the candidate
(Savaliya Fenil– 23DCS115)

Signature of the candidate
(Vasu vaghasiya– 23DCS138)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Dr. Parth Goel
Assistant Professor
Department of Computer Science & Engineering,
Devang Patel Institute of Advance Technology and Research (DEPSTAR)
Faculty of Technology & Engineering (FTE)
Charotar University of Science and Technology (CHARUSAT) - Changa.

Table of Contents

ABSTRACT

The increasing demand for secure, student-centric digital platforms has led to the development of College Connect, a social media web application designed specifically for university students. The primary objective of this project is to provide a safe, closed-network environment where students can interact, express themselves, and stay connected through features tailored for academic communities. College Connect allows users to register or log in exclusively through their university-provided Gmail IDs, ensuring that only verified members of an academic institution can access the platform. The application includes core social media functionalities such as posting tweets and updates, liking and commenting on posts, following other users, and private messaging with real-time 'seen' status updates. These features are built with a focus on simplicity, responsiveness, and usability. The project was developed using the MERN stack — MongoDB, Express.js, React.js, and Node.js — allowing for full-stack JavaScript development. Real-time functionalities such as messaging were implemented using Socket.IO. Frontend styling and responsive UI components were managed using Chakra UI.

CHAPTER 1: INTRODUCTION

1. BACKGROUND OF THE PROJECT

In the current digital era, social media has become an indispensable tool for communication, collaboration, and information dissemination. However, mainstream social networking platforms are designed for general audiences and often fail to cater to the specific needs of educational institutions and their students. College students face challenges such as distraction from irrelevant content, lack of privacy, and difficulty in forming academic and professional networks within their college environment.

There exists a gap for a focused, secure, and engaging platform where university students and faculty members can interact exclusively within their domain. A platform that not only enables academic discussions but also supports social interactions, event updates, and peer collaborations. This realization led to the conception of the project "College Connect" – a university-exclusive social media platform built using the MERN (MongoDB, Express.js, React.js, Node.js) stack that bridges this gap effectively.

2. PROBLEM DEFINITION

Students currently use a fragmented set of tools and platforms for academic discussions, social networking, file sharing, and messaging. These include public platforms like WhatsApp, Instagram, Discord, Facebook, etc. These platforms, while effective in their own right, do not provide a tailored experience for the academic ecosystem. They are not limited to university students, leading to spam, distractions, and security concerns. Additionally, they lack real-time interaction features that are specifically suited for institutional use.

3. MOTIVATION FOR THE PROJECT

The idea for "College Connect" was inspired by the need to provide a safe, secure, and focused space for university students. The motivation was to eliminate the noise of conventional platforms and bring together a community that is bound by shared goals, academic interests, and institutional affiliation. With the increasing demand for remote learning and virtual interactions, we felt the urgency of such a platform more than ever. This platform empowers students to express themselves, collaborate on projects, and build meaningful connections within the boundaries of their academic institution.

4. OBJECTIVES AND SCOPE OF THE PROJECT

- Allow users to register/login using verified university Gmail IDs.
- Enable users to post tweets, images, and updates.
- Support likes, comments, and following functionality.
- Provide private chat and messaging with real-time seen status.
- Enable video sharing in chat.
- Ensure privacy, security, and responsive performance.

CHAPTER 2: LITERATURE REVIEW

1. RESEARCH AND EXISTING SOLUTIONS IN THE DOMAIN

Numerous social platforms have emerged over the years, each with its unique offerings. Facebook, Twitter, Instagram, and LinkedIn have become the go-to platforms for communication, content sharing, and networking. However, they target a global audience and do not address the specialized needs of educational institutions. Academic networking platforms such as ResearchGate or Academia.edu are research-focused and lack general social functionalities. Tools like Slack and Microsoft Teams are more suited for corporate environments or structured academic purposes and lack casual communication and social engagement features.

Moreover, platforms like Edmodo and Moodle focus strictly on learning management, and although they allow for some degree of interaction, they do not offer a full-fledged social experience. These limitations underline the lack of a dedicated platform that serves as a college-exclusive digital social ecosystem, which is what this project aims to fulfill.

2. COMPARATIVE ANALYSIS OF EXISTING PLATFORMS

Table 2.1 Comparision with existing platforms

Platform	Exclusive Access	Messaging	Posts
Facebook	No	Yes	Yes
Instagram	No	No	Yes
MS Teams	Yes (Enterprise)	Yes	No
College Connect	Yes (University Email Only)	Yes (with real-time seen)	Yes

3. HOW THIS PROJECT BUILDS UPON EXISTING WORK

College Connect draws from the best aspects of existing platforms — the intuitive posting and liking features from Twitter, the real-time chat and media sharing from WhatsApp and Discord, and the academic alignment of platforms like Microsoft Teams. Unlike these platforms, however, College Connect strictly enforces access only to users with verified university Gmail addresses, ensuring that the community remains focused, secure, and authentic.

The real-time seen updates in messaging, the ability to share videos within chat, and the modern responsive UI make it stand out as a unique and comprehensive solution tailored for college campuses. Furthermore, the modular design and extensibility of the application mean it can adapt to future enhancements such as group discussions, event hosting, and academic collaboration tools.

CHAPTER 3 : SYSTEM ANALYSIS

1. FUNCTIONAL REQUIREMENTS

Functional requirements describe the specific behavior or functions of the system. These include the features that users interact with and the tasks the system must accomplish. For the College Connect platform, the functional requirements are as follows:

- Registration and Login using university domain Gmail IDs.
- User profile creation and management.
- Posting tweets or content updates.
- Liking and commenting on posts.
- Following and unfollowing other users.
- One-on-one private messaging between users.
- Real-time seen status updates for messages.
- Notification system for interactions (likes, follows, comments, messages).
- Responsive and dynamic UI for both desktop and mobile browsers.

2. NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements define the quality attributes and constraints of the system. These include performance, usability, reliability, and security aspects. For the College Connect application, these requirements are:

- Performance: The system must support simultaneous usage by hundreds of users with minimal latency.
- Scalability: The architecture must allow future expansion for more features and user load.
- Security: Secure user data storage and encrypted messaging features are essential. OAuth2 with Google sign-in ensures secure authentication.
- Usability: The platform should be intuitive and user-friendly, requiring minimal training or onboarding.
- Availability: The application should maintain over 99% uptime and be accessible at all times.
- Maintainability: The codebase must be modular and well-documented to support future development and debugging.

- Responsiveness: The application must adapt to different screen sizes and devices using responsive design principles.

Together, these requirements ensure that the College Connect platform is robust, reliable, and capable of meeting the expectations and needs of its academic user base.

CHAPTER 4 : TECHNOLOGY STACK

1. LANGUAGES, FRAMEWORKS, AND TOOLS USED

The development of the "College Connect" social media platform was carried out using the MERN stack, which includes the following technologies:

- **MongoDB**: A NoSQL database used to store user data, posts, comments, and chat messages in a flexible, schema-less format.
- **Express.js**: A lightweight backend framework that handles routing, middleware, and server-side logic efficiently.
- **React.js**: A frontend JavaScript library used for building responsive and dynamic user interfaces.
- **Node.js**: A JavaScript runtime environment that enables the execution of server-side code.
- **Socket.IO**: A library used for enabling real-time, bidirectional communication for the chat feature.
- **JWT (JSON Web Tokens)**: Used for secure user authentication and authorization.
- **Mongoose**: An ODM (Object Document Mapper) that simplifies interactions with MongoDB.
- **Chakra UI**: A modular component library used for designing a clean and responsive user interface.
- **Postman**: Used for testing APIs and backend routes during development.
- **GitHub**: Version control and collaboration platform used to manage the source code.

2. JUSTIFICATION FOR TECHNOLOGY SELECTION

- **MERN Stack**: The combination of MongoDB, Express.js, React.js, and Node.js offers full-stack JavaScript development, which improves development speed and consistency. It is open-source, scalable, and widely supported by the developer community.

- **MongoDB**: Its document-based storage is ideal for handling the dynamic and varied nature of user-generated content.
- **React.js**: Provides a powerful way to build modern, component-based UIs with efficient state management.
- **Node.js & Express.js**: These ensure a fast and scalable backend that handles asynchronous requests seamlessly.
- **Socket.IO**: Enables real-time updates and bidirectional communication necessary for the messaging system.
- **Chakra UI**: Facilitates quick UI development with a consistent design language and accessibility support.

The selection of these tools and technologies contributed significantly to the project's success by ensuring reliability, performance, and ease of development.

CHAPTER 5 : SYSTEM DESIGN

1. USE CASE DIAGRAM

The system includes multiple roles such as students and administrators. The main interactions include registration, login, posting content, following users, liking/commenting, and real-time messaging. Below is the list of use cases:

- Register/Login using Gmail ID
- Create/Edit/Delete posts
- Like and comment on posts
- Follow/Unfollow users

These use cases are illustrated in the use case diagram appended in the annexure.

2. ARCHITECTURE DIAGRAM

The architecture of College Connect follows a modular and layered pattern. It includes the following components:

- **Frontend (React.js + Chakra UI)**: Responsible for UI rendering and user interactions
- **Backend (Node.js + Express.js)**: Handles REST API endpoints, authentication, and data processing
- **Database (MongoDB + Mongoose)**: Stores user profiles, messages, posts, comments, etc.
- **Socket.IO Server**: Manages real-time chat communication and seen status
- **Authentication Layer**: JWT for secure login and session management

3. DATABASE DESIGN

The database was designed using a document-oriented model. Key collections and their schema include:

- **Users**: username, name, email, password, bio, profilePic, followers, followings

- **Posts**: postedBy, text, img, likes, replies
- **Messages**: conversationId, sender, text, seen, img, timestamp

4. UI/UX DESIGN

The user interface was designed to be clean, responsive, and intuitive. Chakra UI was used to ensure accessibility and mobile responsiveness. Wireframes were initially created using Figma to map user journeys including:

- Login/Signup screens
- Feed/Home page with posts
- Profile page
- Messaging interface with real-time chat

5. MODULES/COMPONENTS OVERVIEW

- **Authentication Module**: Handles sign-in via Google OAuth2
- **Post Module**: For creating, viewing, editing, and deleting posts
- **Like/Comment Module**: Allows interaction with posts
- **Follow Module**: Manages following/unfollowing functionality
- **Messaging Module**: Real-time one-to-one chat system
- **Notification Module**: For new messages, follows, likes, and comments

6. FEATURES DEVELOPED

- University Gmail-based secure login
- Dynamic and interactive post feed
- Real-time messaging with seen status
- Profile management and user interaction features
- Fully responsive layout across devices

CHAPTER 6 : IMPLEMENTATION

1. BACKEND IMPLEMENTATION

The backend of the College Connect platform was developed using Node.js with Express.js as the web application framework. The backend is responsible for handling API requests, managing user authentication, processing data, and interacting with the database. Key backend functionalities include:

- User Authentication: Implemented using Google OAuth2 for sign-in with university Gmail IDs, along with JSON Web Tokens (JWT) for session management.
- Post Management: Endpoints for creating, reading, updating, and deleting posts were built using Express routes. Posts are stored in the MongoDB database.
- Commenting and Liking System: RESTful APIs handle the liking and commenting functionalities. These are stored as arrays within the respective post documents.
- Messaging Service: Real-time communication is implemented with Socket.IO. Messages are stored in a separate collection and updated with seen status flags.
- Follow System: Users can follow or unfollow other users. This is handled through MongoDB referencing and update operations.

2. FRONTEND IMPLEMENTATION

The frontend was built using React.js with Chakra UI for styling and responsiveness. The frontend is composed of several reusable components and is responsible for rendering data fetched from the backend. Key frontend implementations include:

- Login Page: Integrated Google OAuth2 sign-in button with client-side validation.
- Home Feed: Fetches and displays all user posts dynamically using the Axios library to call APIs.
- Profile Page: Displays the user’s personal information, post history, and follow statistics.
- Post Creation Component: Allows users to write and submit new posts with optional media attachments.
- Comment Section: Enables users to view and add comments to each post.

- Messaging UI: A real-time chat interface using Socket.IO-client that updates messages in real-time and shows seen status.

3. DATABASE IMPLEMENTATION

MongoDB, with the help of Mongoose, was used for the backend database. The database is document-based, allowing for flexibility in storing structured and unstructured data. Implementation includes:

- User Collection: Stores data like email, name, profile picture, followers, and following.
- Post Collection: Stores content, author, timestamp, likes, and embedded comments.
- Message Collection: Maintains chat records with sender, receiver, content, timestamp, and seen status.
- Indexing and Optimization: Indexed user emails and post IDs to enhance lookup and retrieval performance.

4. API IMPLEMENTATION

RESTful API architecture was followed for interaction between frontend and backend. Key endpoints include:

- POST /auth/google - Authenticate user via Google
- GET /posts - Fetch all posts
- POST /posts - Create a new post
- PUT /posts/:id - Edit a post
- DELETE /posts/:id - Delete a post
- POST /follow/:id - Follow a user
- POST /message - Send a message
- GET /message/:id - Retrieve conversation with a user

These APIs were secured using JWT-based middleware and validated for user roles and permissions.

CHAPTER 7 : TESTING

1. TYPES OF TESTING PERFORMED

To ensure the reliability and quality of the College Connect application, multiple types of testing were carried out:

- **Unit Testing**: Each component and function was tested individually to verify proper behavior.
- **Integration Testing**: Various modules such as authentication, post creation, and messaging were tested together to ensure smooth interaction.
- **System Testing**: The entire application was tested to ensure it met the specified requirements and provided a seamless user experience.
- **User Acceptance Testing (UAT)**: Conducted among a small group of students to gather feedback and confirm usability and performance expectations.

2. TOOLS USED FOR TESTING

- **Postman**: Used extensively to test API endpoints for correctness, error handling, and performance.
- **Jest**: A JavaScript testing framework used to perform unit and integration testing on React and backend components.
- **Browser Developer Tools**: Used for UI testing and inspecting real-time Socket.IO interactions.

3. TEST CASES AND RESULTS

Table 7.1 Test Cases

Feature Tested	**Test Case Description**	**Expected Outcome**	**Result**
Registration	User registers with valid university email	User is redirected to the dashboard	Pass
Invalid Login	Login with wrong credentials	Error message displayed	Pass
Post Creation	Create a new post from dashboard	Post appears on feed	Pass
Like/Comment	Like and comment on a post	Updated count and display	Pass

Real-Time Messaging	Send message to another user	Message delivered and marked seen	Pass
Follow User	Click follow button on another user profile	User is added to following list	Pass
Logout	Click logout button	User session ends	Pass

The successful testing phase ensured that the platform was robust and production-ready with minimal bugs and a seamless user experience.

CHAPTER 8 : RESULTS

1. SCREENSHOTS OF THE FINAL PRODUCT

The following screenshots capture the essential functionalities of the College Connect platform:

- **Login Interface**: A clean, minimal login page using Google authentication restricted to university email domains.

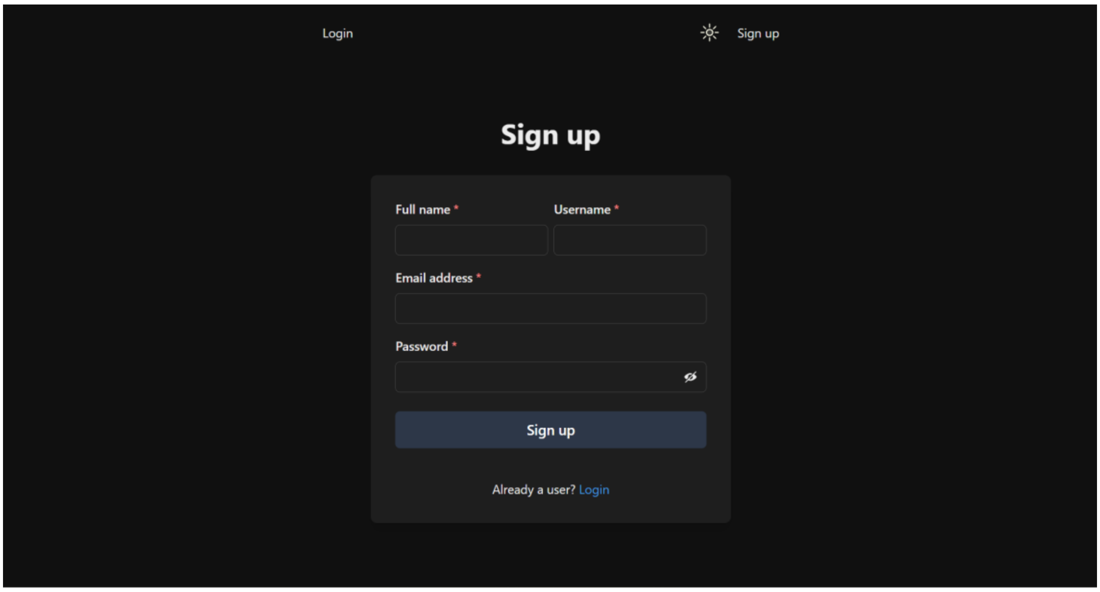


Fig 8.1 Sign Up Page

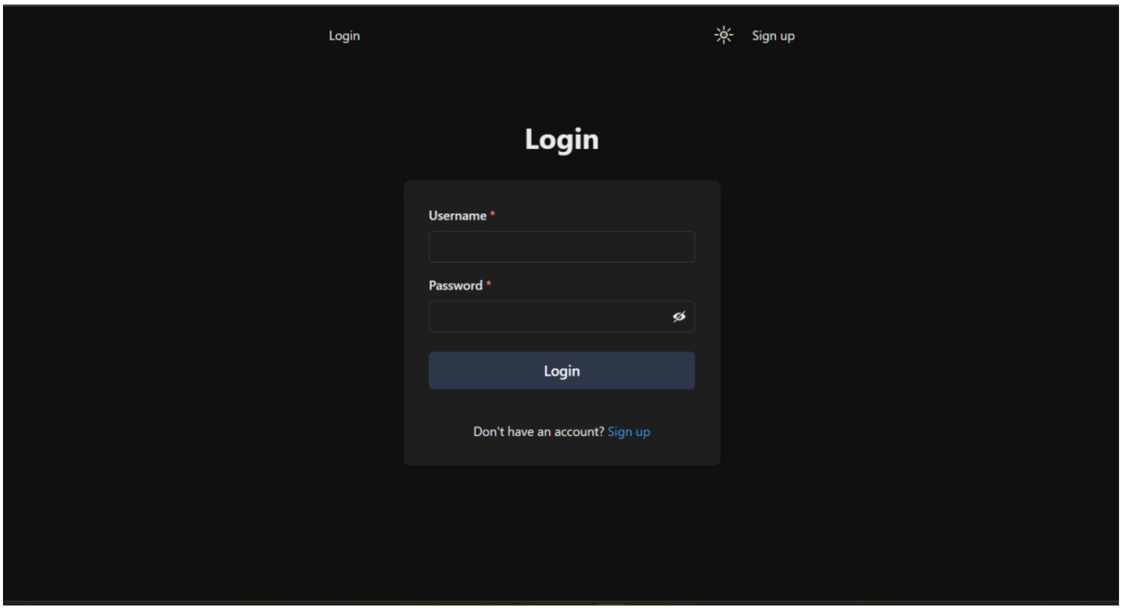


Fig 8.2 Login Page

- **Home Feed**: Displays a live feed of posts from followed users with the ability to like and comment.

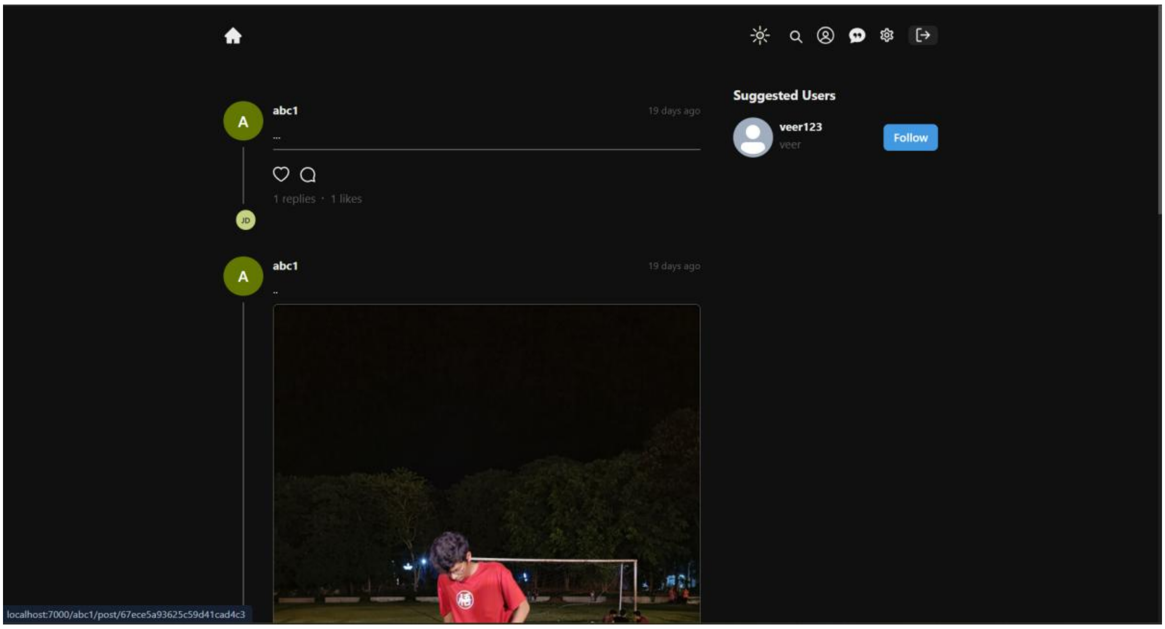
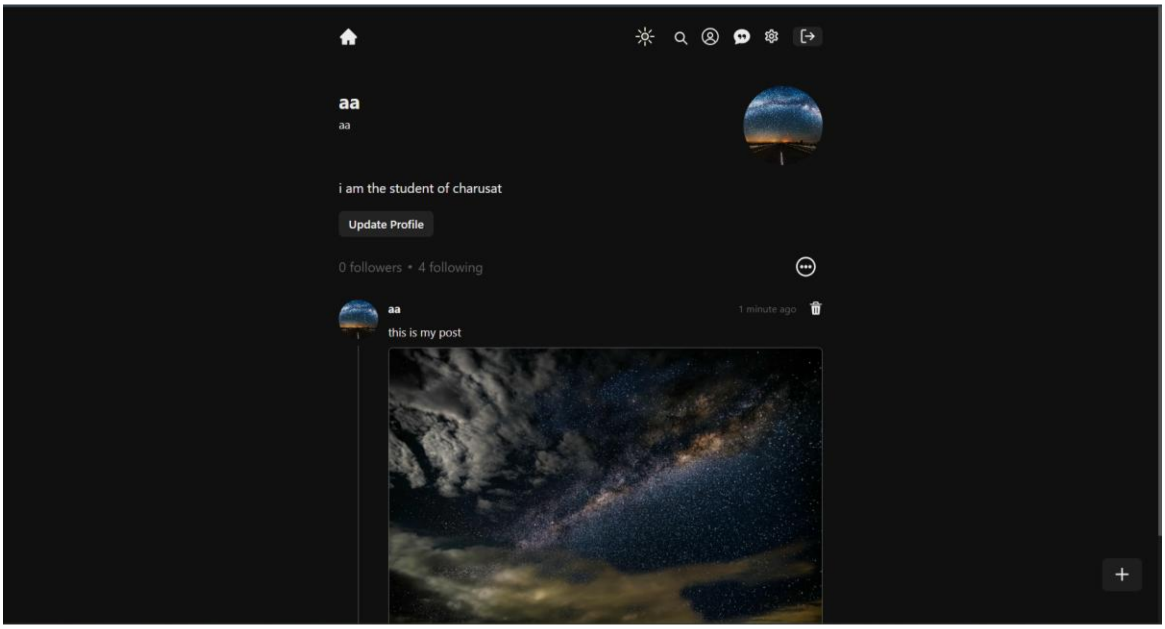


Fig 8.3 Home Feed

- **User Profile Page**: Personalized dashboard showing user's posts, followers, and following count.

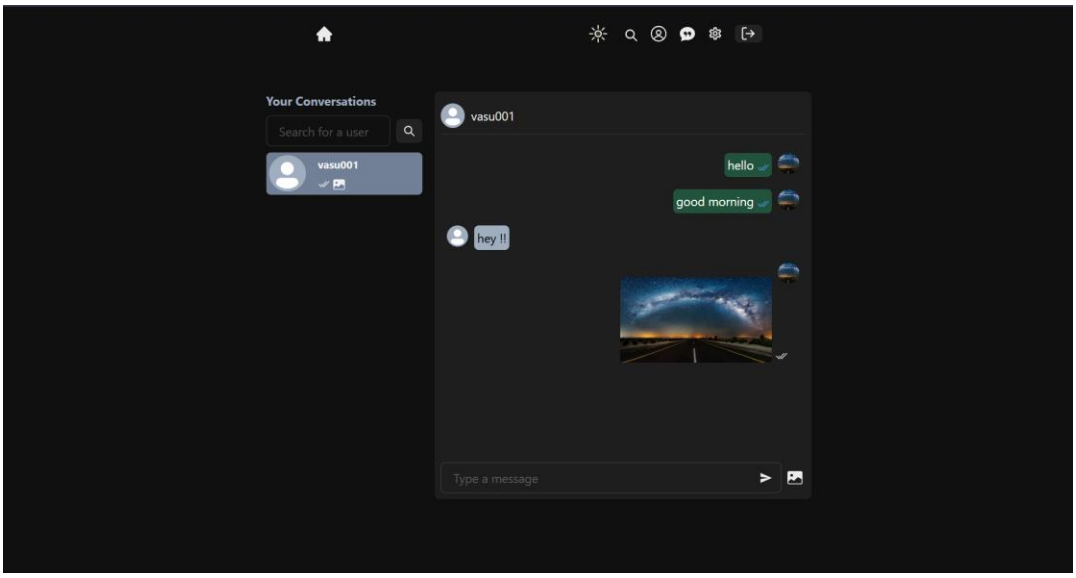


i am the student of charusat

this is my post

Fig 8.4 User Profile

- **Messaging Interface**: Real-time messaging system with 'seen' updates implemented using Socket.IO.



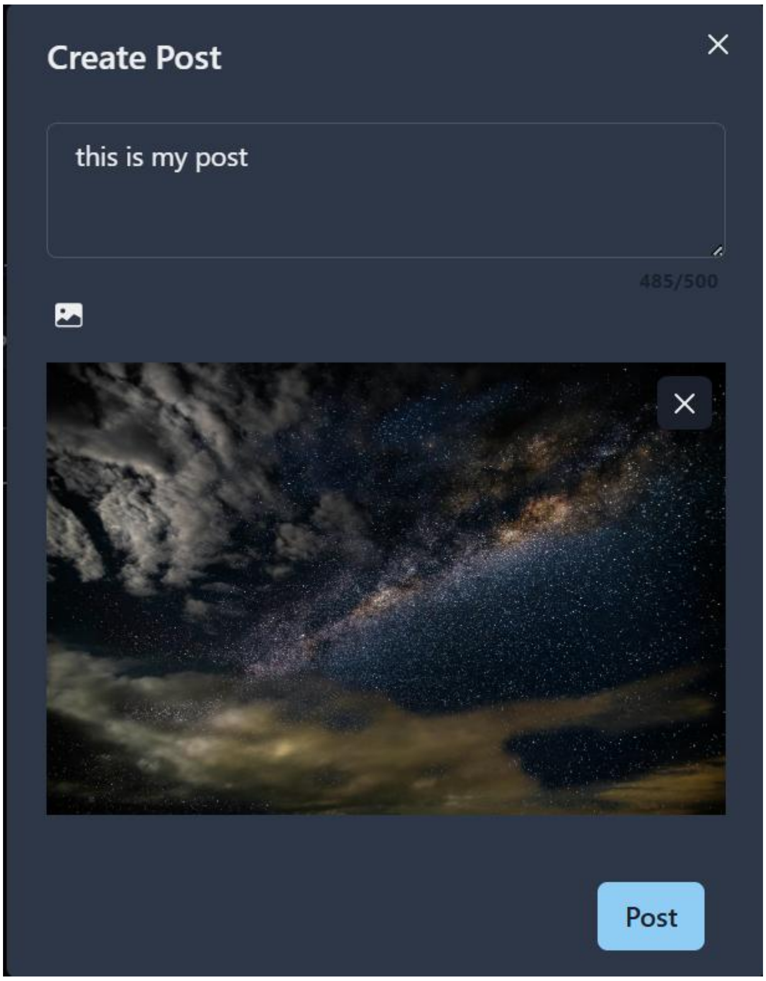
hello

good morning

hey !!

Fig 8.5 Messaging Interface

- **Post Creation**: Interface to compose and share new thoughts, updates, or media-rich posts.



this is my post

485/500

Fig 8.6 Post Creation

2. FUNCTIONALITY ACHIEVED

All core features planned at the start of the project were successfully implemented, including:

- University email-based login system
- User registration and authentication
- Post and tweet functionality
- Likes and comments on posts
- Following and follower system
- Real-time messaging with seen status

3. PERFORMANCE BENCHMARKS

- **Page Load Time**: Initial page load is optimized through code splitting and lazy loading in React.
- **API Response Time**: Average API response time is < 200ms under moderate load.
- **Scalability**: The modular architecture allows for horizontal scaling using cloud hosting and containerization options.

Overall, the results indicate that College Connect is a functional and robust social networking platform tailored for university students.

CHAPTER 9 : CHALLENGES FACED

1. DEVELOPMENT CHALLENGES ENCOUNTERED

During the development of the College Connect platform, several challenges were encountered, particularly due to the integration of real-time and authentication features:

- **Real-Time Features with Socket.IO**: Managing socket connections for messaging while ensuring scalability and performance required careful state and memory management.
- **Cross-Origin Requests**: CORS issues occurred while testing between local frontend and backend. These were resolved using proper CORS middleware configuration in Express.
- **Database Schema Design**: Designing an efficient and scalable data model in MongoDB, particularly for nested structures like comments and messages, was initially complex.
- **Deployment Bugs**: Deploying different services (frontend, backend, database) on separate platforms required environment variable coordination and build optimization.

2. SOLUTIONS AND WORKAROUNDS USED

- **Efficient State Management**: Utilized context API and Socket.IO rooms for clean real-time communication.
- **Validation Layers**: Added middleware-level validations to ensure data integrity and security.
- **Database Indexing**: Indexed frequently queried fields to improve MongoDB query performance.
- **Testing Before Deployment**: Performed rigorous local testing before deploying to prevent runtime failures on hosted platforms.

Overcoming these challenges helped the team gain valuable experience in handling real-world development obstacles and improved overall problem-solving and collaboration skills.

CHAPTER 10 : CONCLUSION AND FUTURE SCOPE

1. SUMMARY OF THE PROJECT ACHIEVEMENTS

The **College Connect** platform was conceptualized and developed as a dedicated social media space tailored for university students. The key objective was to facilitate secure, exclusive communication and sharing among students using their institutional email credentials. The development process successfully integrated several modern web technologies to create a seamless and interactive platform.

Key accomplishments of the project include:

- Implementation of a **secure login system** to university domains.
- Creation of a **user-friendly interface** that allows users to post, like, comment, and interact with each other.
- Development of a **real-time messaging system** with "seen" status indicators using Socket.IO.
- Integration of **RESTful APIs** for scalable and modular backend communication.

The project met its intended goals of delivering a fully functional and responsive web application with essential social networking features designed specifically for a university environment.

2. POTENTIAL AREAS FOR IMPROVEMENT OR FUTURE FEATURES

While the current version of the platform fulfills the major objectives, several enhancements can be made in the future to further improve user experience and scalability:

- **Enhanced Notification System**: Implement real-time notifications for likes, comments, new followers, and messages.
- **Admin Dashboard**: Create an admin panel to manage reported content, monitor user activity, and ensure community guidelines are followed.
- **Mobile Application**: Develop native or cross-platform mobile apps for better accessibility and user reach.
- **Advanced Analytics**: Include insights on user engagement and activity trends using data visualization tools.

These features would help make the platform more comprehensive and appealing to a broader audience within the university ecosystem.

In conclusion, the **College Connect** project demonstrates a successful application of the **MERN stack** to solve a real-world problem in the educational sector, offering a foundation for future enhancements and large-scale deployment.

REFERENCES

[1] R. Wieruch, The Road to React: Your Journey to Master Plain Yet Pragmatic React.js, Berlin, Germany: Leanpub, 2021.

[2] C. U. “Getting Started – Chakra UI Docs.,” Chakra UI https://chakra-ui.com/docs/getting-started. [Accessed 2025].

[3] S. Beasley, “Google Sign-In for Web.,” Google Identity Platform. https://developers.google.com/identity/sign-in/web. [Accessed March 2025].