ConnectX

A Project Report submitted in partial fulfilment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

(Hons.)

by

Ananya Agrawal (2115800002)

Supervised by:

Dr. RamManohar Nisarg

Department of Computer Engineering & Applications

Institute of Engineering & Technology

GLA University

Mathura- 281406, INDIA

May 2024

CONTENTS

Declaration

We hereby declare that the project work entitled **"*ConnectX*"** submitted to the GLA University, is a record of an original work done by us groupmates under the guidance of ***Dr. RamManohar Nisarg*** and this project work is submitted in the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science & Engineering. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

(Signature of Mentor)

Acknowledgment

It gives us a great sense of pleasure to present the report of the BTech(H) project undertaken during the **BTech III Year.** This project is going to be an acknowledgment of the inspiration, drive, and technical assistance that will be contributed to it by many individuals.

We owe a special debt of gratitude to, ***DR. RAM MANOHAR SIR*** for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towardsa constructive goal, and for *his* constant support and guidance to our work. *His* sincerity, thoroughness, andperseverance have been a constant source of inspiration for us. We believe that he will shower us with all his extensively experienced ideas and insightful comments at different stages of the project & also teach usabout the latest industry-oriented technologies.

We also do not like miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and cooperation.

GROUP MEMBER NAME:ANANYA AGRAWAL

2115800002(02)

Abstract

The Internet has rapidly evolved from just a social platform to a social network that is used to share content, ideas, and information. As a result of social networking, people's communication styles have changed. It affects almost every aspect of our lives: education, communication, employment, politics, health care, public relations, and personal productivity. A telecommunications service (SNS) is An online platform used to build and develop relation between peoples. It provides ways to Users to interact online with people with similar interests, either for romantic or social purposes. Emails, instant messages, online comments, wiki, digital photographs, videos, and blog post submissions are all possible. It also gives people with disabilities the possibility to express their mind and ideas in public. Social networks serve two roles as content companies and purchasers. They choose which users get access to his information. A profile is generated with answers to questions, consisting of age, area, hobbies, and many others. Some web sites permit users to upload pics, upload multimedia content material or alternate the profile and sound of profiles, weblog posts, feedback, hyperlinks, and sharing Contact listing. Users can choose who can view, edit, and add to their friend list, and so on, to preserve their privacy on social media platforms. Social networking has modified the way people communicate, percentage data, and engages with people. It enables people to communicate as well as interact with each another over the internet. As the recognition of social media grows, new technology grows to be increasingly famous.

1. Introduction

1.1 Overview and Motivation

The ConnectX project embarks on a journey to revolutionize online community platforms, driven by the aspiration to foster engagement and cultivate meaningful interactions among users. Inspired by the success stories of renowned discussion forums and message boards, our endeavor seeks to introduce a modern and feature-rich platform tailored to the evolving needs of digital communities.

At the heart of our project lies a profound motivation to bridge the gap between virtual interactions and genuine connections. In an era dominated by digital communication, the significance of fostering authenticrelationships and nurturing vibrant communities cannot be overstated. With this motivation in mind, we embarked on the ambitious task of conceptualizing and developing the ConnectX, aiming to provide users with a platform where they can engage, share, and connect with like-minded individuals from aroundthe globe.

1.2 Objective

Define Project Requirements-

The first step in our project journey is to define the requirements that will guide the development process. This involves gathering insights from stakeholders, understanding user needs, and outlining the functionality and features that the ConnectX will offer. Key components of defining project requirements include identifying user roles, specifying core functionalities such as thread creation, commenting, and communitymanagement, determining technical specifications, and establishing project timelines and milestones. By clearly defining project requirements, we ensure alignment between stakeholders' expectations and the finalproduct, laying a solid foundation for the development process.

Frontend Development with React.js-

In React.js frontend development endeavors for the ConnectX, we're committed to creating a compelling user interface and seamless user experience. Our approach begins with translating the design specifications outlined in Figma into functional and visually appealing components. Each aspect, from buttons to forms, is meticulously crafted to align with the intended design language. Ensuring responsiveness across various devices and screen sizes is a core focus, achieved through the implementation of responsive design principles and CSS media queries. Leveraging React's capabilities, we prioritize server-side rendering to optimize page loading speeds and improve search engine visibility. Additionally, we structure our application for intuitive navigation using React's routing capabilities, organizing pages and components to facilitate seamless user journeys. Performance optimization is paramount, and we employ React's best practices to minimize load times and enhance overall responsiveness. Key user interaction features, such as thread creation and commenting, are seamlessly integrated using React's state management and component lifecycle methods. By combining these elements, we aim to deliver a visually appealing, intuitive, and high-performing frontend that enhances user engagement and satisfaction within the ConnectX ecosystem.

Authentication-

Authentication plays a pivotal role in the operation of a social media app, acting as the gatekeeper to ensure that only authorized users gain access to the platform's features and content. The process typically begins with user registration, where individuals provide personal details such as their name, email address, and password. Verification steps may follow, where the app confirms the authenticity of the provided information, often by sending a verification code to the user's email or phone number. Once registered, users can log in using their credentials or through alternative methods like single sign-on with other platforms. Upon successful login, the app generates authentication tokens, such as JSON Web Tokens (JWTs), which serve as proof of the user's identity. These tokens are sent with each subsequent request to the server, facilitating seamless authentication without the need for repetitive logins. Session management mechanisms keep track of user sessions, managing their duration and expiration to ensure security and privacy..

Backend Development using Express.js-

Backend development using Express.js forms the backbone of our project, enabling the implementation of server-side logic, data management, and API integration. Express.js, a lightweight and flexible Node.js framework, provides a scalable and efficient platform for building backend services. Our backend development efforts will focus on creating RESTful APIs to handle CRUD operations for users, threads, comments, communities, and notifications. We will also implement middleware for request handling, error management, and logging to ensure smooth operation of the backend infrastructure. By leveraging Express.js, we aim to build a reliable and performant backend system that supports the functionality and scalability requirements of the ConnectX.

Features to Added

Responsive Design for All Devices: Ensure that the ConnectX application is easily accessible and user-friendly across various devices, including smartphones, tablets, and desktops.

Enable users to find others: With similar preferences and experiences. This facilitates meaningful connections and builds a more engaging community within your application.

User-Friendly Chat Features: Implement intuitive chat features to facilitate seamless communication and interaction between users.

Simplified User Feedback System: Create a straightforward and user-friendly feedback system to encourage user input and suggestions.

Enable Friend Requests: Implement a friend request system to allow users to send and receive friend requests, facilitating connections between users.

Implement User Authentication: Develop a secure authentication system to allow users to sign up, log in, and manage their accounts securely.

Security Measures: Implement basic security practices such as input validation, password hashing, and protecting against common web vulnerabilities.

Database Connectivity: Establish a connection between the application and a MongoDB database using Mongoose, allowing seamless storage and retrieval of user data.

Testing & Development-

Software testing is vital for social media apps to ensure quality, functionality, and reliability. It enhances user experience by identifying and fixing bugs and glitches. Security is a top priority, and testing uncovers vulnerabilities to protect user data. Performance optimization ensures smooth operation under high loads, and compatibility testing ensures seamless functioning across devices. Testing instills trust, enhancing the platform's reputation. Common methodologies include unit testing, integration testing, functional testing, performance testing, security testing, and usability testing. Best practices include testing early and often, using automation, replicating real-world scenarios, cross-platform testing, collaborative testing, regression testing, and performance monitoring.

.

1.3 Summary of Similar Applications

Social media has become an integral part of modern communication, connecting people globally and facilitating various forms of interaction. This survey aims to explore a selection of existing social media applications, analyzing their features, target demographics, and unique selling propositions.

Facebook:

- Founded in 2004 by Mark Zuckerberg, Facebook is one of the earliest and most widely used social media platforms.
- Features include user profiles, news feeds, groups, events, and messaging services.
- Target audience: Broad demographic, particularly popular among adults aged 18-65+.
- Unique selling proposition: Extensive user base, diverse content sharing options, and robust advertising capabilities.

Instagram:

- Launched in 2010, Instagram focuses on photo and video sharing, allowing users to apply filters and edit their content.
- Key features: Feed, stories, IGTV, reels, and direct messaging.
- Target audience: Predominantly younger demographics, particularly teenagers and young adults.
- Unique selling proposition: Emphasis on visual content, influencer marketing, and engagement through likes and comments.

Twitter:

- Founded in 2006, Twitter is a microblogging platform enabling users to share short messages known as tweets.
- Features include retweets, replies, hashtags, and multimedia attachments.
- Target audience: Varied, but skewed towards professionals, journalists, and influencers.
- Unique selling proposition: Real-time information dissemination, public conversation platform, and trending topics discovery.

LinkedIn:

- Established in 2003, LinkedIn is a professional networking platform designed for career development and business connections.
- Features include user profiles, job postings, company pages, and professional content sharing.
- Target audience: Professionals, job seekers, recruiters, and businesses.
- Unique selling proposition: Focus on professional networking, skill endorsements, and industry-specific content.

2. Software Requirement Analysis

In this pivotal chapter, we embark on a detailed exploration of the software requirements for the ConnectX project. Understanding these requirements is fundamental to the success of the project, as they serve as the blueprint for the development process. We delve into various aspects, including technical feasibility, functional requirements, non-functional requirements, and use cases, to ensure a comprehensive understanding of project needs and objectives.

2.1 Functional Requirements :-

- **User Registration and Authentication :-** Users should be able to create accounts, log in securely, and manage their profiles.
- **Profile Creation :-** Users should have the ability to create detailed profiles, including personal information, interests, and preferences.
- **Messaging System :-** Provide a messaging system for users to communicate with each other securely within the application.
- **Privacy Controls :-** Implement robust privacy controls to allow users to manage their privacy settings. Users should be able to control who can send them messages.

2.2 Non-Functional Requirements :-

- **Security :-** Ensure the security of user data through encryption, secure authentication mechanisms, and adherence to data protection regulations.
- **Usability :-** Design an intuitive and user-friendly interface that is easy to navigate and accessible to users of all ages and technological proficiency levels.
- **Compatibility :-** Ensure compatibility with various devices, operating systems, and web browsers**.**
- **Accessibility :-** Ensures that all sections of users can access our application.
- **Performance :-** Optimize application performance to ensure fast response times, minimal latency, and scalability.

2.3 Technical Requirements :-

- **Backend Framework :-** Choose a suitable backend framework for developing the application’s server-side logic, database management, and API development.
- **Frontend Framework :-** Select a frontend framework for building the user

interface of the application, ensuring responsiveness and cross-browser compatibility.

- **Database :-** Choose a reliable and scalable database solution for storing user data, preferences, messages, and event information securely.
- **Messaging Service :-** Integrate a messaging service or API for real-time communication between users within the application.

2.4 Technical Feasibility :-

- **MERN Stack :-** The MERN stack (MongoDB, Express.js, React.js, Node.js) is well-suited for developing dynamic and interactive web applications. MongoDB provides a flexible and scalable NoSQL database solution, while Express.js and Node.js enable server-side logic and API development. React.js facilitates the creation of responsive and user-friendly frontend interfaces.

- **Socket.IO :-** Socket.IO is a JavaScript library that enables real-time, bidirectional communication between web clients and servers. It is ideal for implementing features like instant messaging and live updates in the ConnectX application.

- **Server-Side Rendering :-** The MERN stack offers performance optimizations such as server-side rendering (SSR) with React.js, caching mechanisms with Express.js, and asynchronous I/O operations with Node.js. These optimizations help minimize latency, improve page load times, and enhance overall application performance.

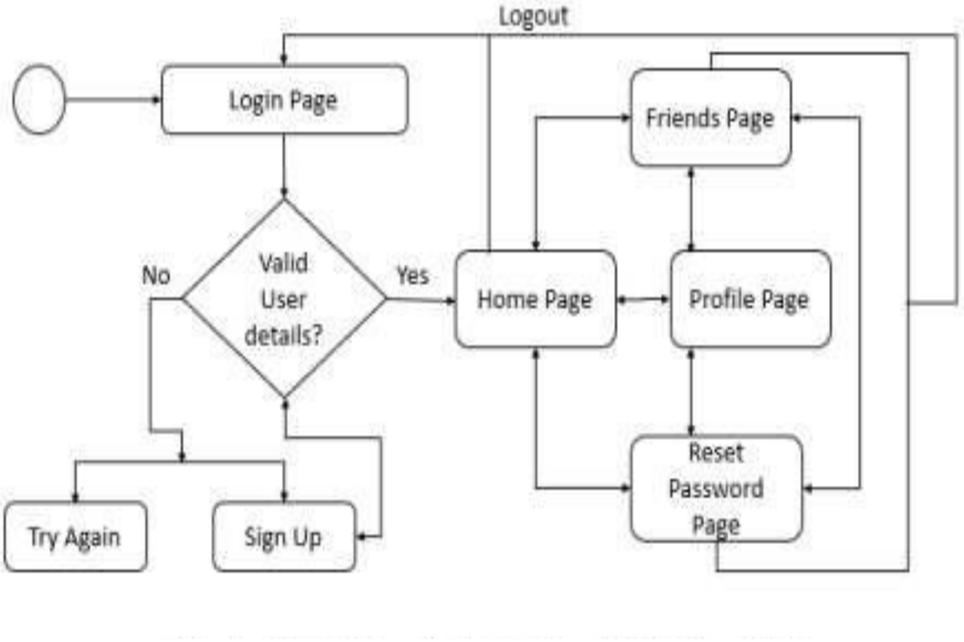
- **Security :-** The ConnectX application can implement security measures such as data encryption, input validation, authentication, and authorization using middleware and libraries available in the MERN stack ecosystem.Socket.IO supports secure WebSocket connections over HTTPS and enables implementation of authentication mechanisms and message encryption to ensure data privacy and integrity.

- **Resources :-** The MERN stack has a large and active community of developers, extensive documentation, and a wide range of libraries and frameworks available for building web applications. This ensures ample availability of resources, tutorials, and support for development and maintenance of the ConnectX application.

3. Implementation

3.1 Project Interface

The implementation of the ConnectX project involves several stages, encompassing frontend development, backend development, database management, and integration of various features and functionalities. Below is an overview of the implementation process:



- .**Frontend Development (React):**
- Set up a React project using **create-react-app** or a similar tool.
- Design and implement UI components for various features such as user authentication, profile management, news feed, posting content, commenting, liking, and notifications.
- Utilize React Router for client-side routing between different pages.
- Integrate third-party libraries like Material-UI or Ant Design for UI components and styling.
- Ensure responsiveness and accessibility across different devices and screen sizes.
- **2. Backend Development (Node.js with Express.js):**
- Create a Node.js project and set up an Express.js server.
- Use MongoDB as the database for storing user data, posts, comments, likes, and other relevant information.
- Define RESTful APIs to handle CRUD operations for users, posts, comments, etc.
- Implement user authentication using JWT tokens or session cookies.

- Set up middleware for error handling, logging, and authentication.
- **3. Database Management (MongoDB):**
- Set up a MongoDB Atlas cluster or install MongoDB locally.
- Define schemas for users, posts, comments, etc., using Mongoose ODM.
- Establish connections to the MongoDB database from the Express.js server.
- **4. Integration of Features:**
- Implement features such as user registration, login, logout, and profile management on the frontend.
- Allow users to create, edit, and delete posts with text, images, or videos.
- Enable users to comment on posts, like posts, and follow/unfollow other users.
- Implement real-time updates using WebSocket technology or polling for notifications, new posts, comments, etc.
- Utilize libraries like Axios or Fetch API for making HTTP requests to the backend APIs.
- **5. Testing and Deployment:**
- Write unit tests for backend APIs and frontend components using testing frameworks like Jest, Mocha, or React Testing Library.
- Deploy the backend server to a platform like Heroku or AWS Elastic Beanstalk.
- Deploy the frontend app to a CDN like Vercel or Netlify for static hosting.
- Set up CI/CD pipelines for automated testing and deployment.
- **6. Continuous Improvement:**
- Gather user feedback and analytics to identify areas for improvement and new feature requests.
- Iterate on the design and functionality based on feedback, ensuring a seamless user experience.
- Stay updated with the latest security patches and best practices for MERN stack development.
- Incorporate performance optimizations to enhance the app's speed and scalability.
-

3.2 Key Features

1. **Profile Creation and Customization:** Users can create personalized profiles, upload profile pictures, and customize their bios to reflect their personalities and interests.

2. **News Feed:** The app features a dynamic news feed algorithm that curates content based on user preferences, interests, and social connections, ensuring a tailored and engaging browsing experience.

3. **Multimedia Sharing:** ConnectX allows users to share photos, videos, links, and status updates with their followers and friends, encouraging creativity and self-expression.

4. **Real-time Messaging:** The app offers a seamless messaging interface, enabling users to engage in private or group conversations with friends, family, and acquaintances.

5. **Events and Announcements:** ConnectX includes features for organizing and promoting events, parties, and gatherings, as well as for sharing announcements and updates with followers.

6. **Like/Dislike Comments:** Users have the ability to like or dislike comments on posts, enabling them to express their opinions and engage with the content more interactively.

7. **Login and Logout:** ConnectX provides secure login and logout functionality, allowing users to access their accounts and log out securely when they are finished using the app.

8. **Privacy and Security:** The app prioritizes user privacy and security, implementing robust encryption protocols, privacy settings, and content moderation tools to safeguard user data and ensure a safe and respectful online environment.

9. **Accessibility:** ConnectX is designed to be accessible to users of all abilities, with features such as customizable text sizes, screen readers, and other accessibility options.

10. **Ads and Promotions:** Allows businesses and individuals to create targeted advertisements and sponsored content to reach specific audiences.

11. **Integration with Other Platforms:** Seamless integration with other social media platforms, allowing users to share content across multiple networks.

4. Flow Chart

4.1 Data Flow Diagram

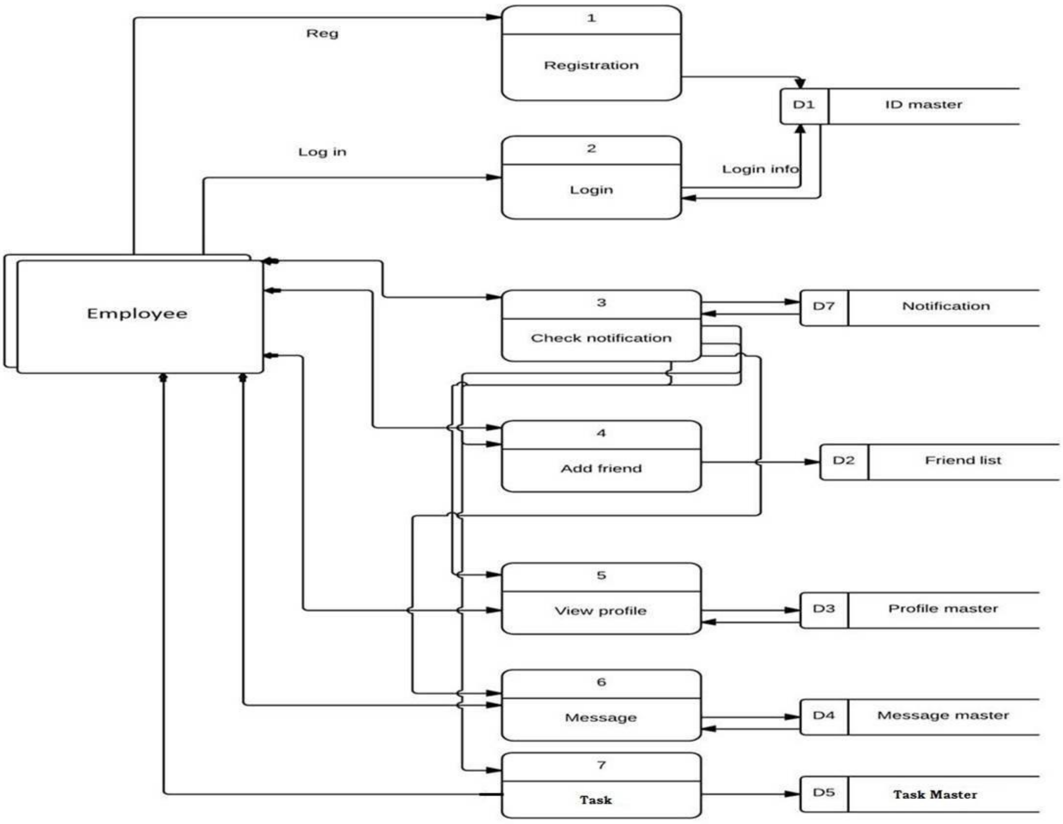


Figure 4.1: Data Flow Diagram

4.2 Use case Diagram

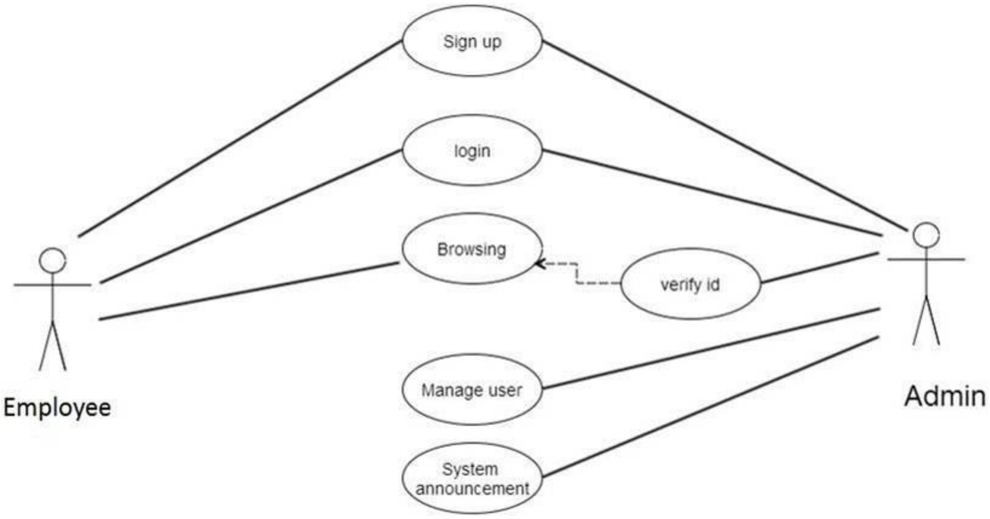


Figure 4.2: use case diagram

4.3 E-R Diagram

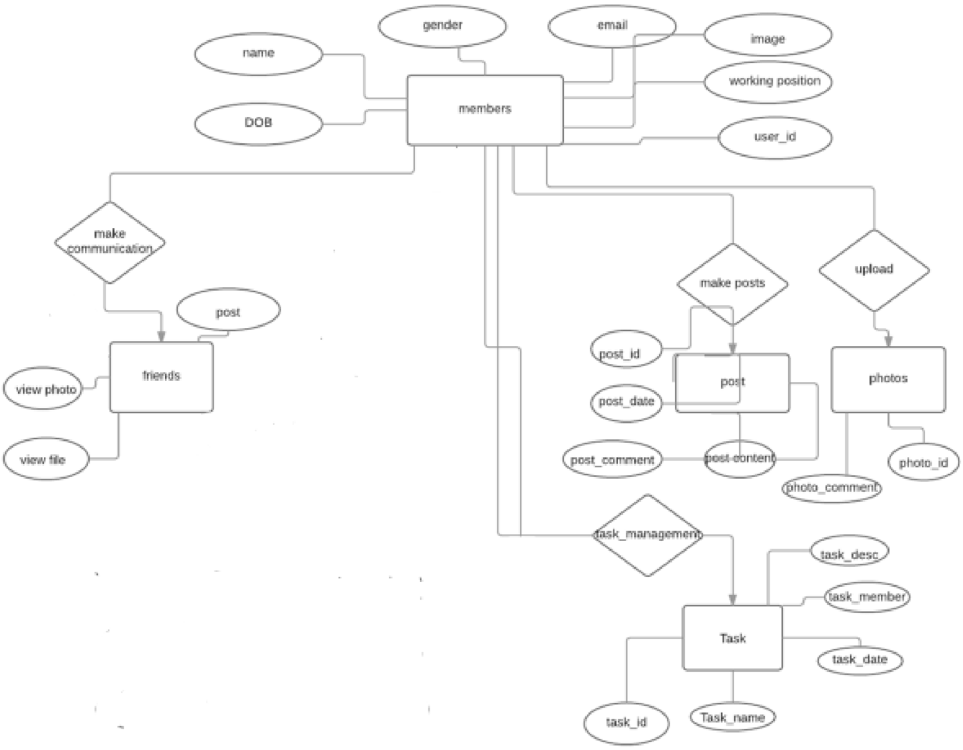
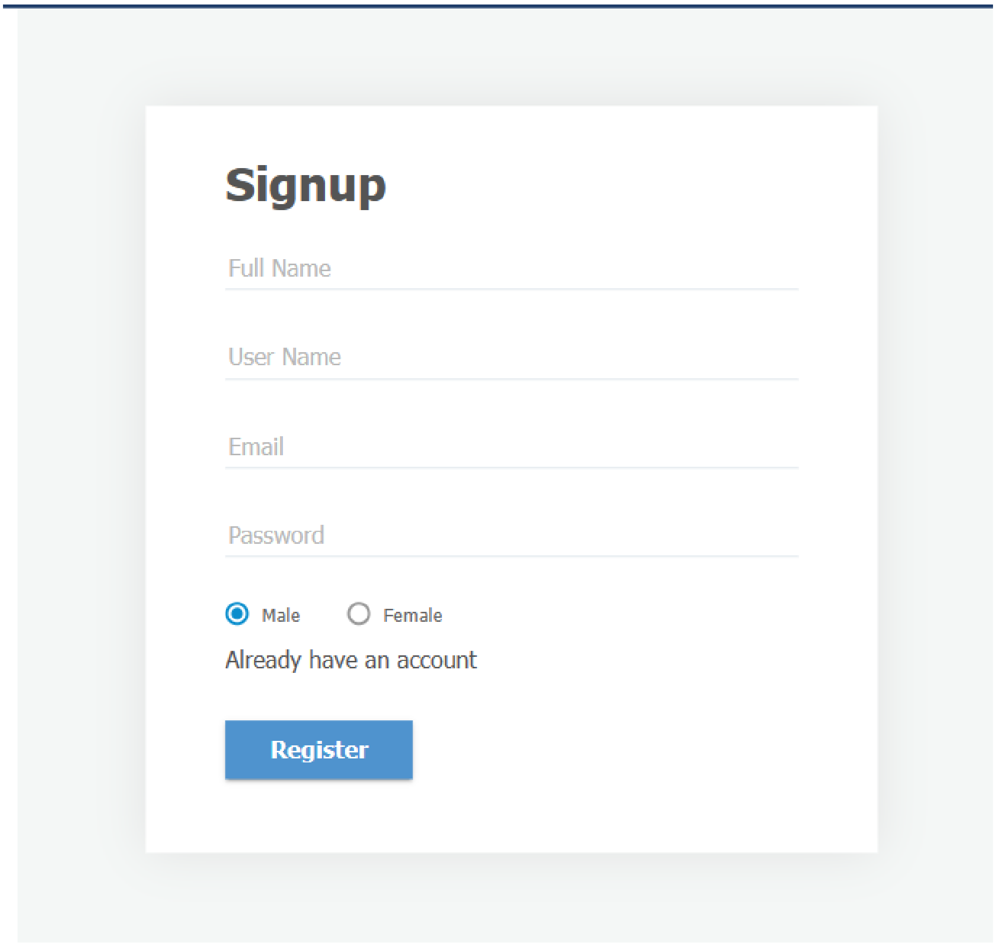


Figure 4.3: E-R Diagram.

5. Images



Signup

Full Name

User Name

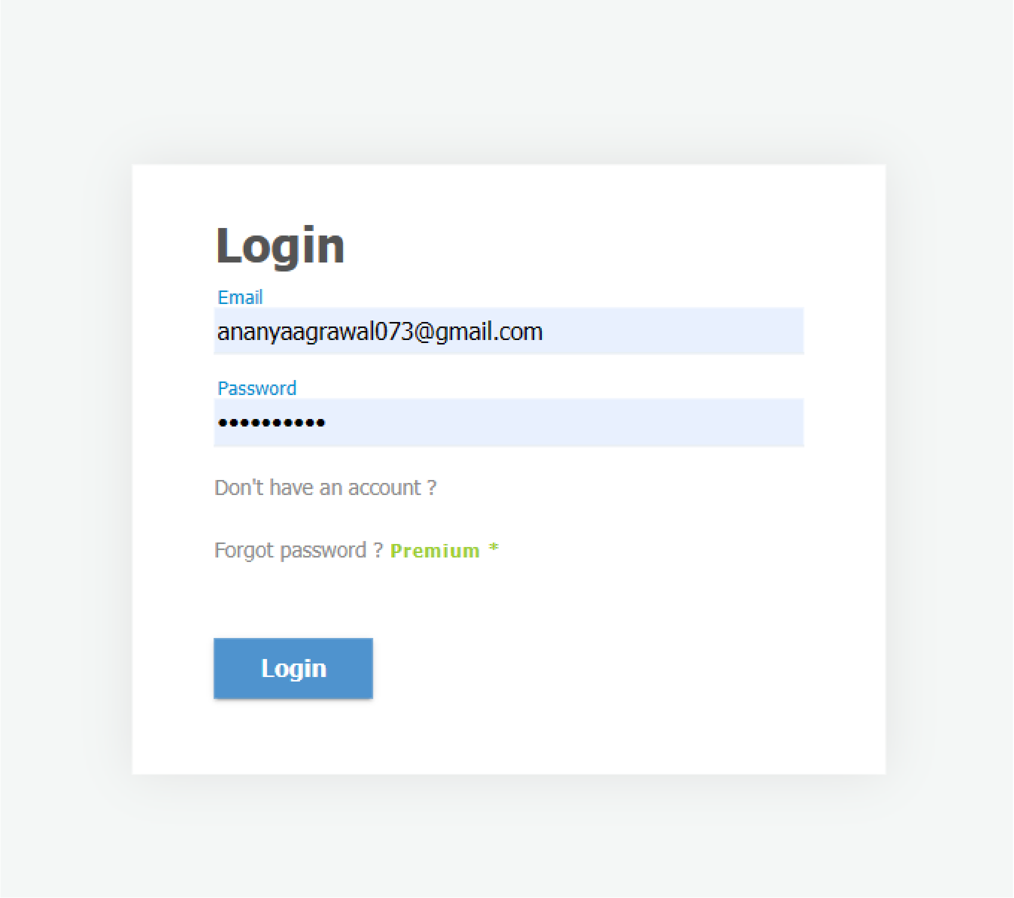
Email

Password

● Male ○ Female

Already have an account

Figure 1: Sign up page



Login

Email

ananyaagrawal073@gmail.com

Password

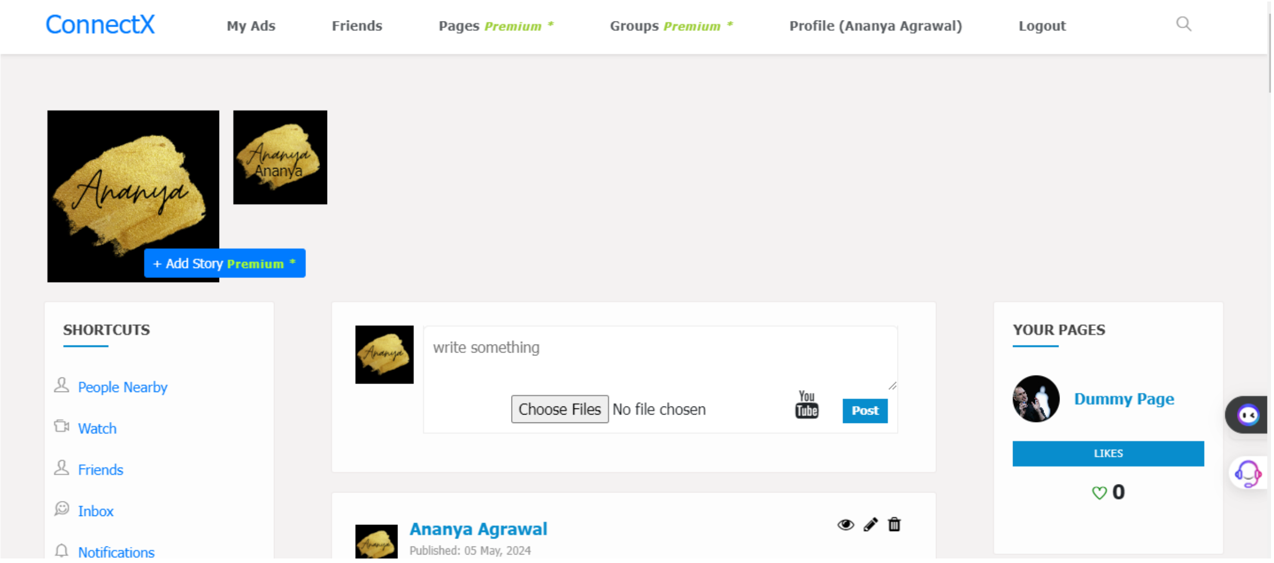
••••••••••

Don't have an account ?

Forgot password ? Premium *

Login

Figure 2 Login Page



ConnectX

My Ads | Friends | Pages Premium * | Groups Premium * | Profile (Ananya Agrawal) | Logout

+ Add Story Premium *

SHORTCUTS

- People Nearby
- Watch
- Friends
- Inbox
- Notifications

write something

Choose Files No file chosen

Post

YOUR PAGES

Dummy Page

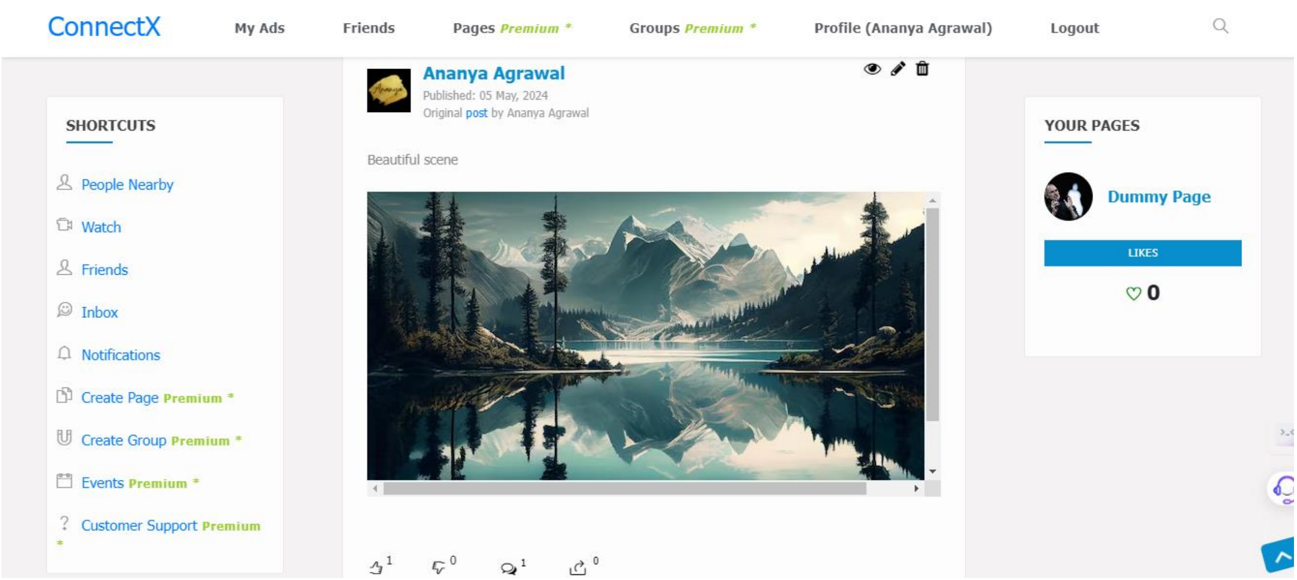
LIKES

♡ 0

Ananya Agrawal

Published: 05 May, 2024

Figure 3 Home Page



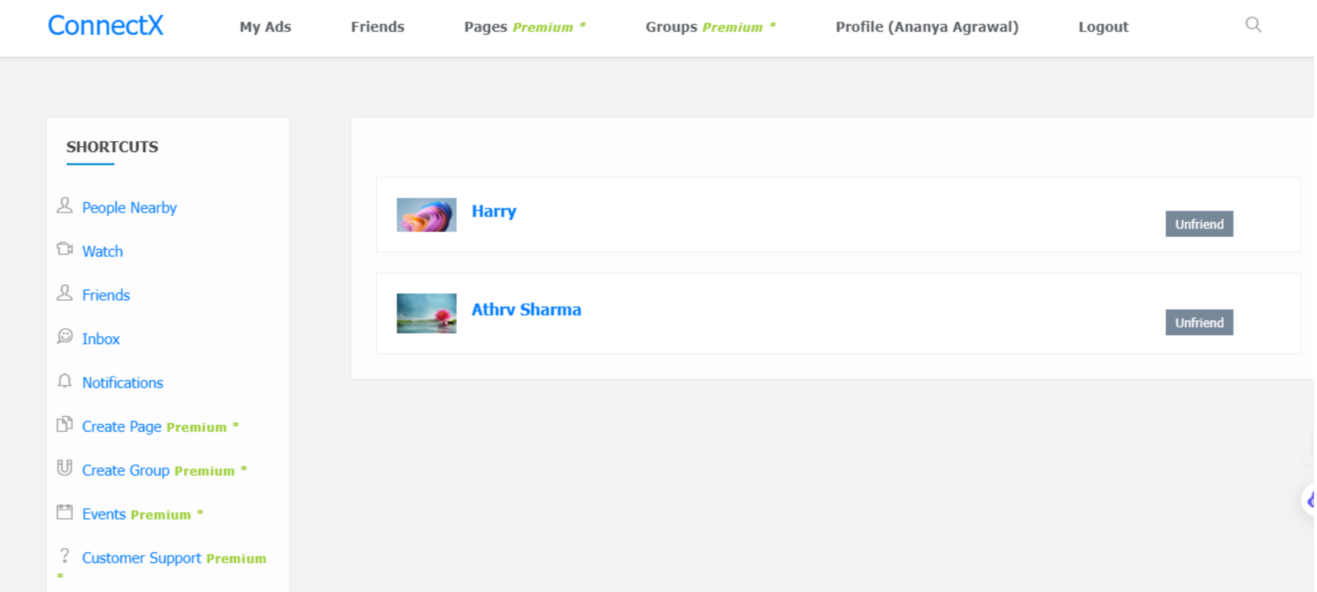
Ananya Agrawal

Published: 05 May, 2024

Original post by Ananya Agrawal

Beautiful scene

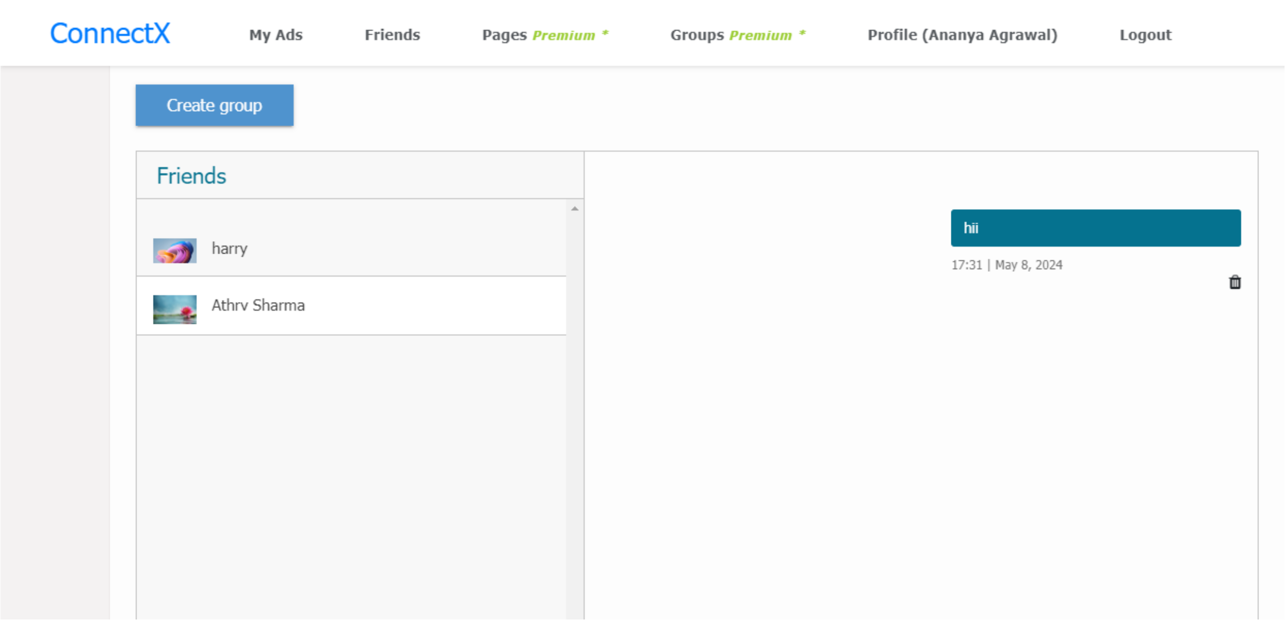
Figure 4 Home Page



Harry Unfriend

Athrv Sharma Unfriend

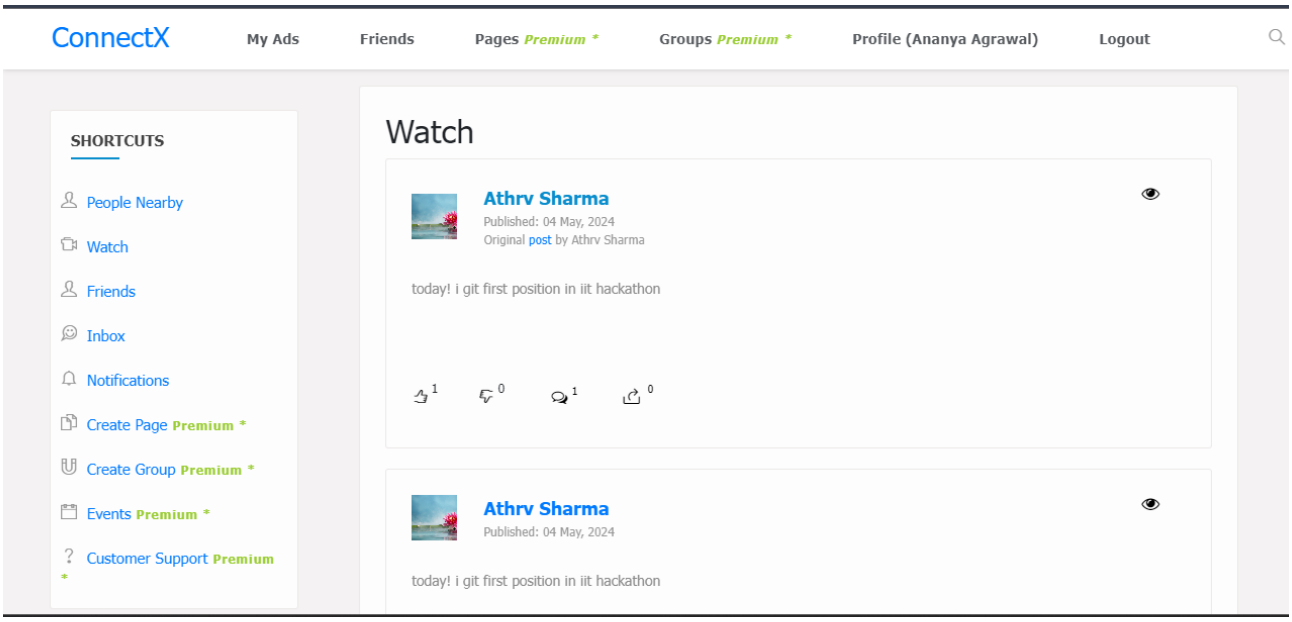
Figure 5: Friends Page



hii

17:31 | May 8, 2024

Figure 6: Chatting Area



Athrv Sharma
Published: 04 May, 2024
Original post by Athrv Sharma

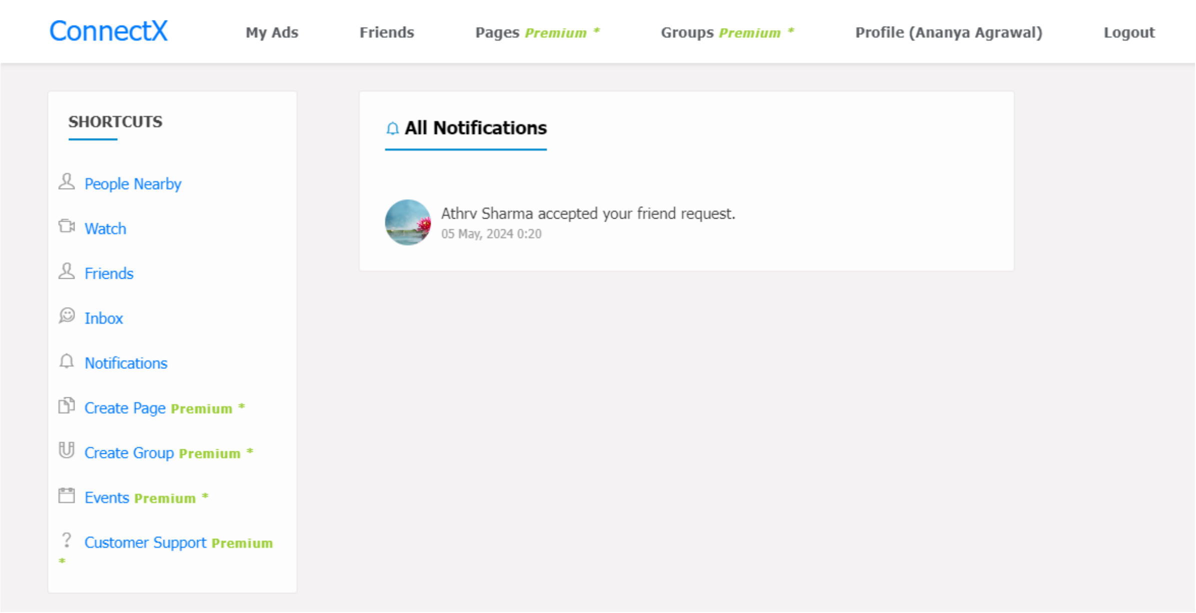
today! i git first position in iit hackathon

1 0 1 0

Athrv Sharma
Published: 04 May, 2024

today! i git first position in iit hackathon

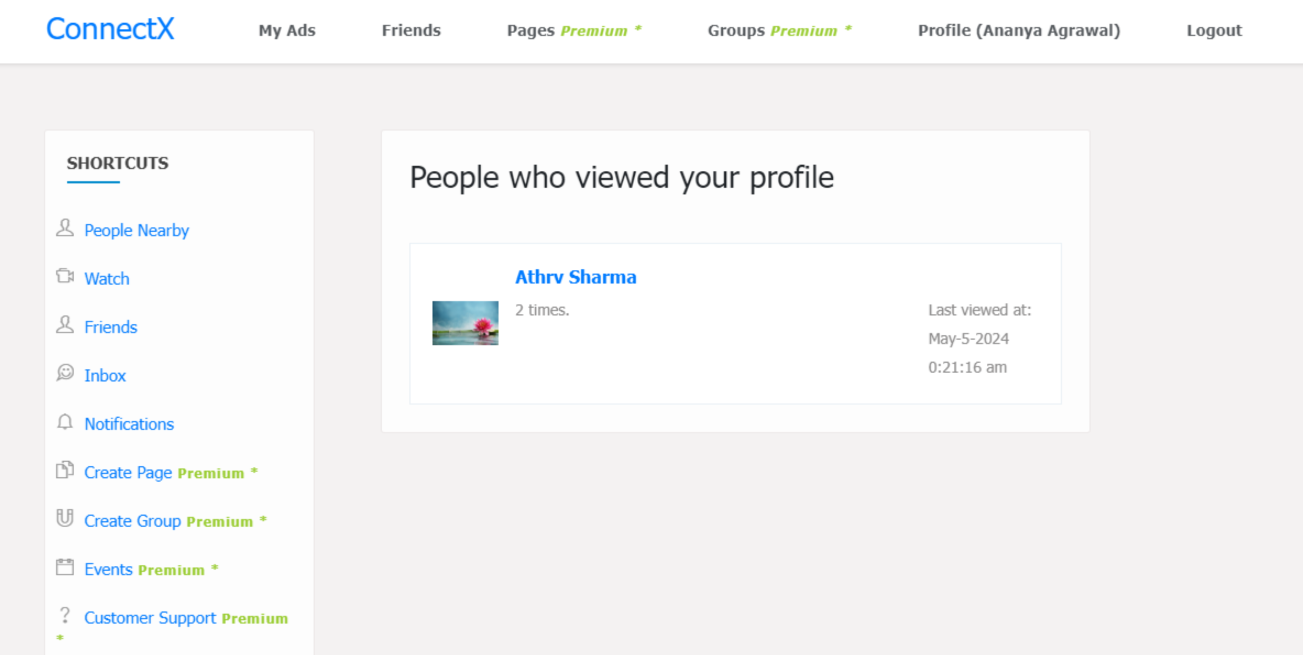
Figure 7:Watch Section



All Notifications

Athrv Sharma accepted your friend request.
05 May, 2024 0:20

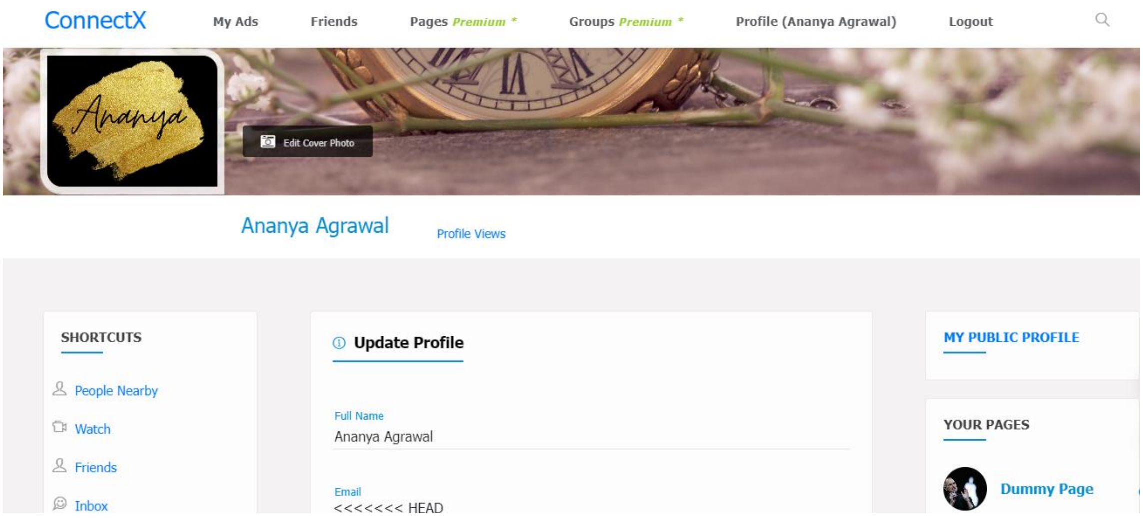
Figure 8:Notification Section



People who viewed your profile

Athrv Sharma
2 times.
Last viewed at:
May-5-2024
0:21:16 am

Figure 9:Profile View Section



Ananya Agrawal

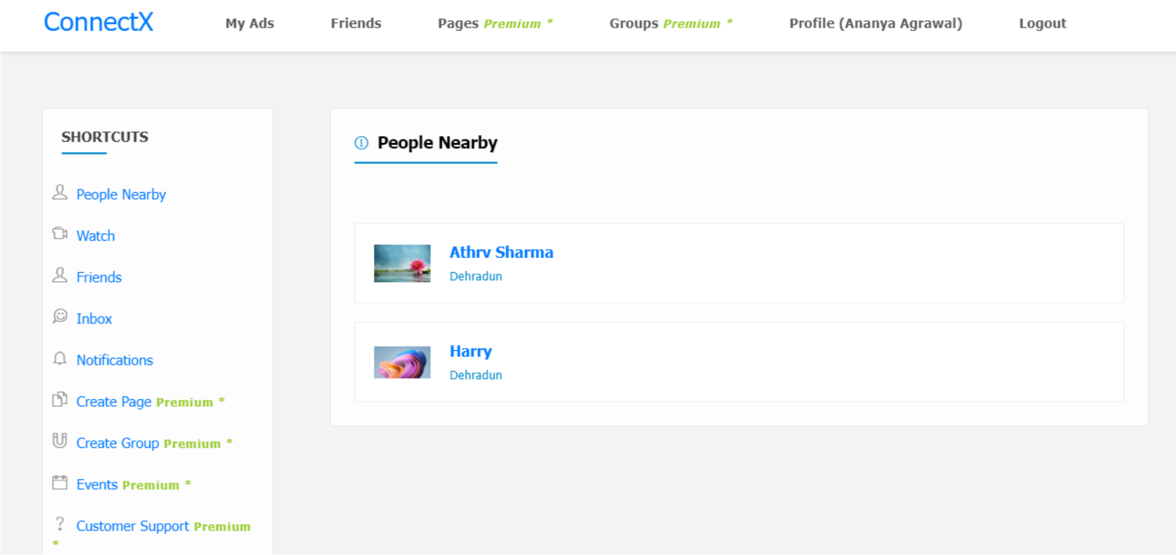
Profile Views

Update Profile

Full Name
Ananya Agrawal

Email
<<<<<<< HEAD

Figure 10:Profile section

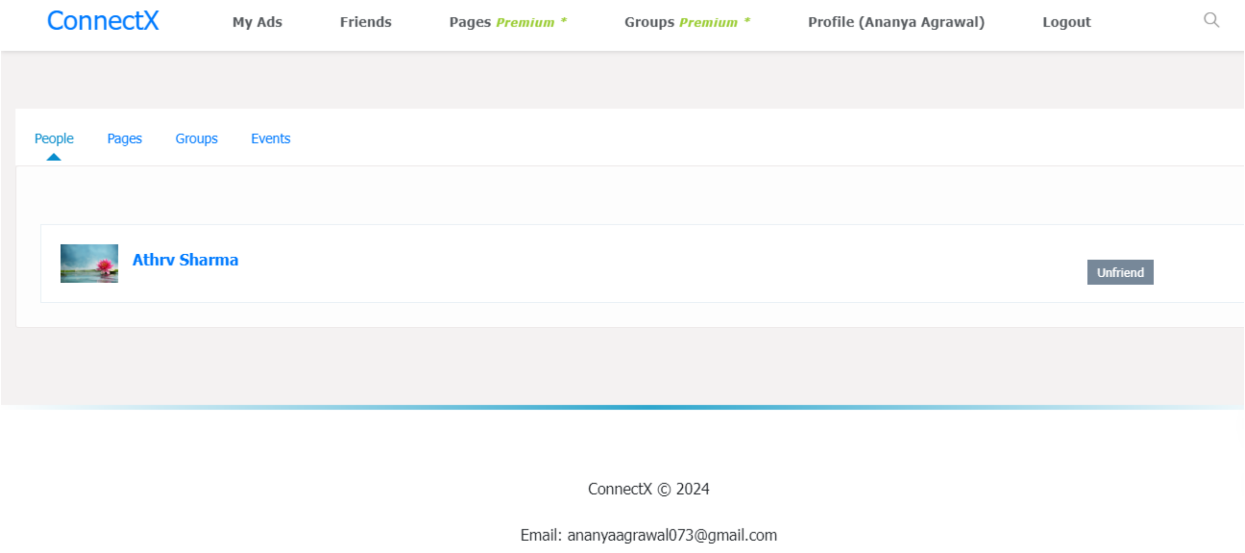


People Nearby

Athrv Sharma
Dehradun

Harry
Dehradun

Figure 11:People Nearby Section

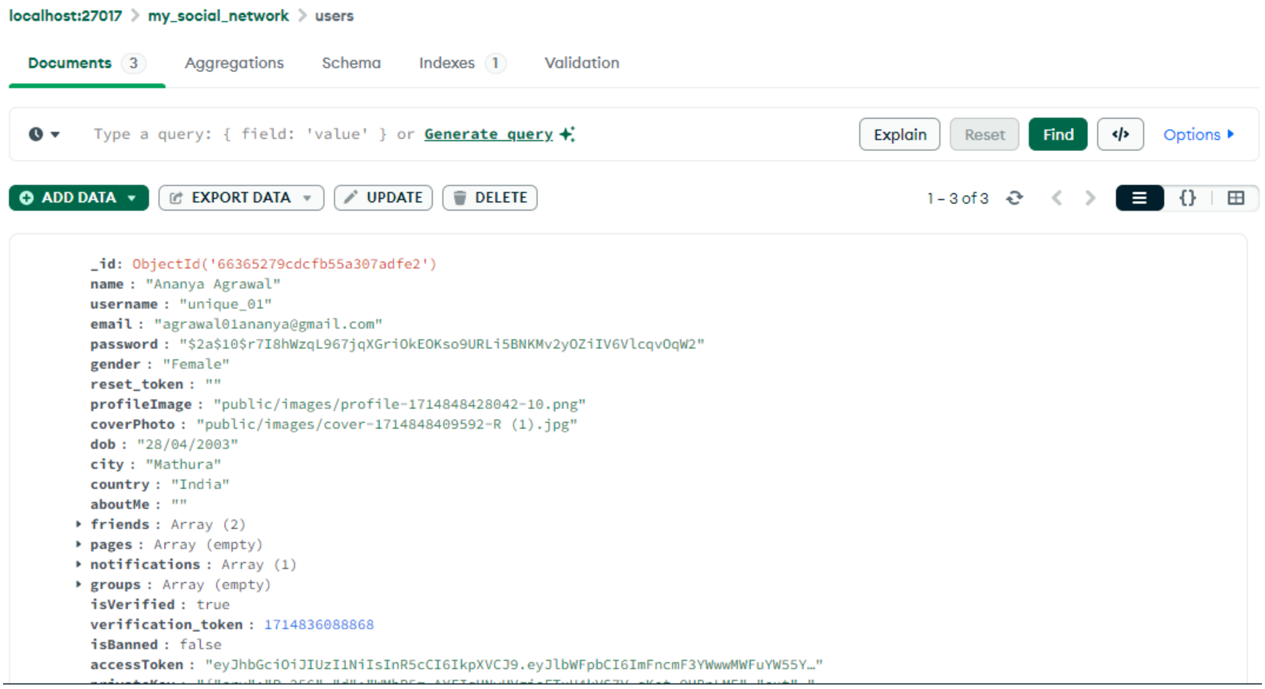


Athrv Sharma

ConnectX © 2024

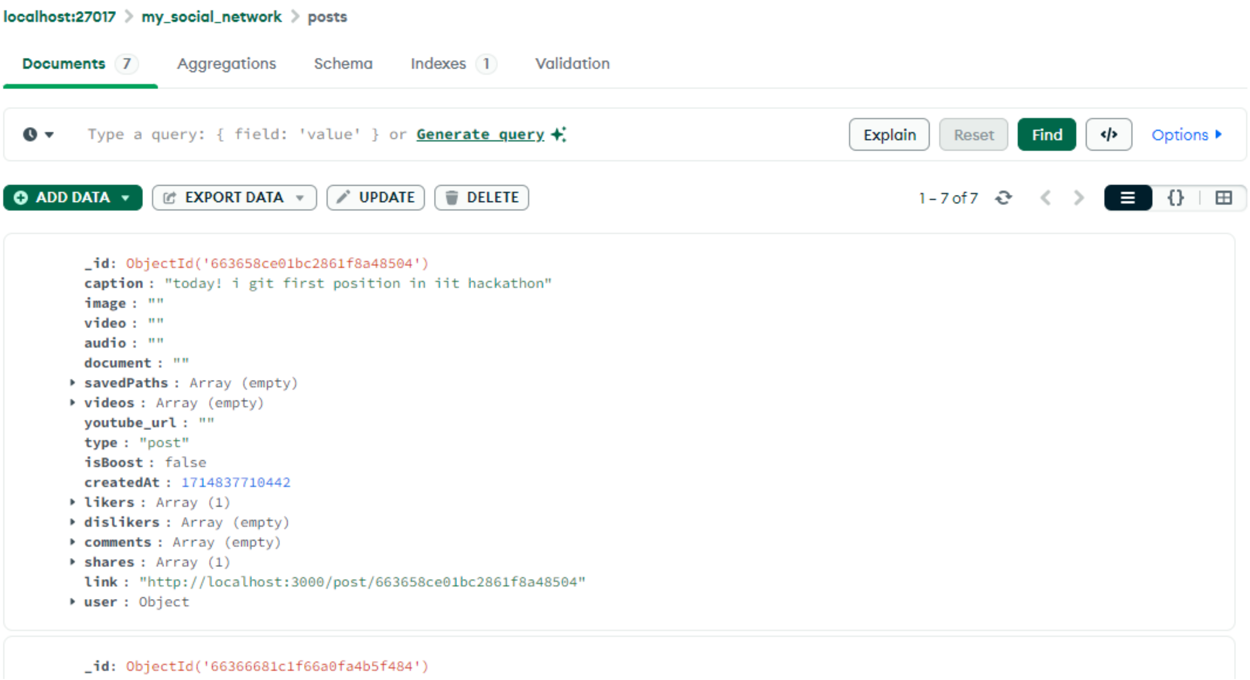
Email: ananyaagrawal073@gmail.com

Figure 12:Search Page



```
_id: ObjectId('66365279cdcfb55a307adfe2')
name : "Ananya Agrawal"
username : "unique_01"
email : "agrawal01ananya@gmail.com"
password : "$2a$10$r7I8hWzqL967jqXGriOkEOKso9URLi5BNKMv2yOZiIV6VlcqvOqW2"
gender : "Female"
reset_token : ""
profileImage : "public/images/profile-1714848428042-10.png"
coverPhoto : "public/images/cover-1714848409592-R (1).jpg"
dob : "28/04/2003"
city : "Mathura"
country : "India"
aboutMe : ""
friends : Array (2)
pages : Array (empty)
notifications : Array (1)
groups : Array (empty)
isVerified : true
verification_token : 1714836088868
isBanned : false
accessToken : "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlbWFpbCI6ImFncmF3YWwwMWFuYW55Y…"
```

Figure 13:Users Data



```
_id: ObjectId('663658ce01bc2861f8a48504')
caption : "today! i git first position in iit hackathon"
image : ""
video : ""
audio : ""
document : ""
savedPaths : Array (empty)
videos : Array (empty)
youtube_url : ""
type : "post"
isBoost : false
createdAt : 1714837710442
likers : Array (1)
dislikers : Array (empty)
comments : Array (empty)
shares : Array (1)
link : "http://localhost:3000/post/663658ce01bc2861f8a48504"
user : Object

_id: ObjectId('66366681c1f66a0fa4b5f484')
```

Figure 14:Post Details

6. Software Testing

Software testing marks a critical phase in the software development lifecycle – software testing. This chapter is dedicated to exploring the various testing methodologies, strategies, and techniques employed to validate the functionality, reliability, and performance of the ConnectX.

Testing is an essential aspect of software development, serving as a mechanism to identify and mitigate defects, errors, and vulnerabilities in the application. Effective testing ensures that the ConnectX meetsthe specified requirements, performs as expected under various conditions, and delivers a seamless user experience.

In Software Testing , we delve into the following key aspects of software testing for the ConnectX:

1. **Functional Testing:** Functional testing focuses on verifying that each feature and functionality of the ConnectX performs as intended according to the specified requirements. This includes testing individual components, user interactions, and system behaviors to ensure that they meet user expectations and business objectives.

2. **Usability Testing:** Usability testing evaluates the ease of use, intuitiveness, and overall user experience of the ConnectX This involves gathering feedback from real users through user interviews, surveys, and usability tests to identify any usability issues or pain points and make iterative improvements to the application's design and functionality.

3. **Performance Testing:** Performance testing assesses the responsiveness, scalability, and stability of the ConnectX under various load conditions. This includes stress testing, load testing, and endurance testingto determine how the application performs under normal usage, peak usage, and prolonged usage scenarios, ensuring optimal performance and reliability.

4. **Compatibility Testing:** Compatibility testing ensures that the ConnectX functions correctly across different devices, browsers, and operating systems. This involves testing the application on various platforms and configurations to identify any compatibility issues or inconsistencies and optimize the application for a seamless user experience across all environments.

Through a comprehensive testing approach encompassing functional testing, usability testing, performance testing, compatibility testing, and security testing, we aim to ensure that the ConnectX meets the highest standards of quality, reliability, and security. By identifying and addressing any issues or deficiencies early in the development process, we strive to deliver an application that exceeds user expectations and delivers exceptional value.

7. Conclusion

In conclusion, the development of a social media application using the MERN (MongoDB, Express.js, React.js, Node.js) stack offers a versatile and efficient platform for creating dynamic and interactive user experiences. Through MongoDB's flexibility in handling large volumes of data, Express.js's robustness in building server-side applications, React.js's ability to create engaging user interfaces, and Node.js's scalability in managing server-side operations, the MERN stack provides a comprehensive solution for developing modern social media platforms.

By leveraging the power of MERN, developers can streamline the development process, enhance performance, and ensure seamless integration across different layers of the application. Moreover, the modular architecture of the MERN stack allows for easy maintenance and updates, facilitating the continuous evolution of the social media app to meet changing user needs and technological advancements.

Incorporating features such as user authentication, real-time updates, multimedia content sharing, and social networking functionalities, the MERN-based social media app can cater to a wide range of user preferences and interactions. Furthermore, with the availability of numerous third-party libraries, frameworks, and APIs, developers can extend the app's functionality and integration with external services, enriching the overall user experience.

In essence, the MERN stack provides a solid foundation for building innovative and scalable social media applications that empower users to connect, share, and engage in meaningful ways. By harnessing the collective strengths of MongoDB, Express.js, React.js, and Node.js, developers can create compelling social media experiences that resonate with users and drive community engagement in the digital age.

8. Summary

The MERN (MongoDB, Express.js, React.js, Node.js) stack emerges as a comprehensive solution for crafting contemporary social media applications. By harnessing MongoDB's prowess in managing vast volumes of data, Express.js's robustness in constructing server-side applications, React.js's knack for crafting immersive user interfaces, and Node.js's scalability in handling server-side operations, the MERN stack furnishes developers with a potent toolkit for creating platforms that resonate with today's digital consumers.

This amalgamation of technologies not only expedites the development process but also elevates performance levels, ensuring seamless integration across various layers of the application. From user authentication mechanisms to real-time data updates, multimedia content sharing functionalities to social networking features, MERN-powered social media apps boast a plethora of capabilities to cater to diverse user preferences and interaction patterns.

Furthermore, the modular architecture inherent in the MERN stack facilitates streamlined maintenance and enables effortless updates, empowering developers to adapt swiftly to evolving user needs and technological advancements. Leveraging a plethora of third-party libraries, frameworks, and APIs further enriches the functionality and integration potential of these applications, fostering enhanced user experiences and community engagement.

In essence, the MERN stack equips developers with the tools necessary to craft innovative, scalable social media platforms that serve as catalysts for meaningful user connections and vibrant online communities. As the digital landscape continues to evolve, MERN stands as a beacon of versatility and efficiency, driving the development of next-generation social media experiences.

9. References

- https://en.wikipedia.org/wiki/Enterprise_social_networking

  https://www.lucidchart.com/?utm_source=google&utm_medium=

  cpc&utm_campaign=lucidchart_othercountries&gclid=CI_ c8Pzmk9ACFcgXaAodei8AtQ
- http://mashable.com/2013/06/14/enterprise-social-networks/ #kQqgGRbfPkqM
- http://www.zyncro.com/en/overview/benefits-uses-enterprise-social-networks

  http://www.ragan.com/Main/Articles/9_benefits_of_an_internal_ social_network_45790.aspx

Books: Konshin, K., 2018. *Next. Js Quick Start Guide: Server-side rendering done right*. Packt Publishing Ltd.

Al Salmi, H., Comparative CSS frameworks.

Websites: https://legacy.reactjs.org/docs/getting-started.html

https://expressjs.com/

Faculty Guidelines

Dr. Ram Manohar Nisarg

GitHub Repository Link

https://github.com/Ananya01Agrawal/ConnectX-