

PONDICHERRY UNIVERSITY

(Central University)



SCHOOL OF ENGINEERING AND TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE
Master of Computer Applications

NAME : MUHAMMED RAIHAN P A

REG. NO. : 21352030

SEMESTER : IV

SUBJECT : MAIN PROJECT REPORT

GUIDED BY : Dr. M. Nandhini

CamBuzz - Social Media & Resource locator

By

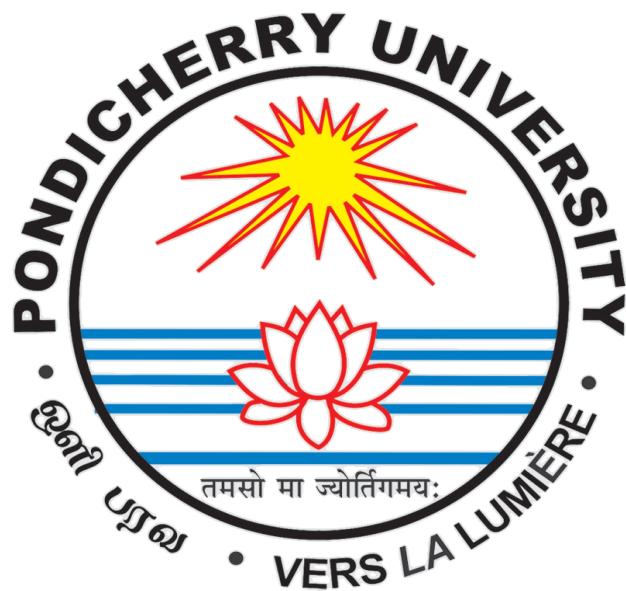
MUHAMMED RAIHAN P A

(Registration Number : 21352030)

Guided by **Dr. M. Nandhini**

**Project report submitted in partial fulfilment of the requirements
for the award of the degree of**

MASTER OF COMPUTER APPLICATIONS



**DEPARTMENT OF COMPUTER SCIENCE
SCHOOL OF ENGINEERING & TECHNOLOGY
PONDICHERRY UNIVERSITY**

May 2023

BONAFIDE CERTIFICATE

This is to certify that this project work entitled "**CamBuzz - Social Media & Resource locator**" is a bonafide record of work done by **Mr. MUHAMMED RAIHAN P A** (Reg. Number 21352030) in the partial fulfilment for the degree of Master of Computer Applications of Pondicherry University.

This work has not been submitted elsewhere for the award of any other degree to the best of our knowledge.

INTERNAL GUIDE

Dr. M. Nandhini
Associate Professor
Department of Computer Science
School of Engineering & Technology
Technology Pondicherry University
Pondicherry – 605 014

HEAD OF THE DEPARTMENT

Dr. S.Siva Sathya
Professor/HOD
Department of Computer Science
School of Engineering &
Pondicherry University
Pondicherry – 605 014

Submitted for the Viva-Voce Examination held on :

INTERNAL EXAMINER

EXTERNALEXAMINER

ACKNOWLEDGEMENT

I would like to extend my sincere gratitude and appreciation to all those who have contributed to the successful completion of your project. Without their support, your project would not have been possible.

Firstly, I would like to express your gratitude to your Project Guide **Dr. M. Nandhini, Associate Professor, Department of Computer Science, Pondicherry University, Pondicherry**, for her invaluable guidance and direction throughout the duration of the project. Her unwavering support and constant care have been instrumental in shaping your project and providing you with a new perspective to work with.

I also acknowledge the contribution of **Dr. S.Siva Sathya, Professor and Head, Department of Computer Science, Pondicherry University, Puducherry, and Dr. T. Chithralekha, Dean, School of Engineering & Technology, Pondicherry University, Puducherry**, for providing the necessary facilities and resources to work on your project.

I would like to express your gratitude towards all the faculty members of the Department of Computer Science and others who have contributed directly or indirectly to the project. Their insights and suggestions have played a significant role in the development of the project.

Furthermore, I would like to acknowledge the unwavering support and encouragement from my parents, which has helped me to overcome challenges and complete my project successfully.

Lastly, I believe that the blessings of God have enabled you to complete your project with flying colours. Once again, congratulations on your accomplishment, and I wish you all the best for your future endeavours.

(MUHAMMED RAIHAN P A)

SYNOPSIS

Cambuzz is an innovative social media website tailored specifically for the university community, including students and faculty members. Its primary objective is to create a central hub where individuals can easily share information, collaborate on projects, and communicate effectively. By providing a dedicated platform for the university environment, Cambuzz aims to enhance the overall experience of students and faculty members.

One key feature of Cambuzz is its campus resource locator. With this feature, students can quickly locate and access various resources on campus, such as computer laboratories, study spaces, and printing facilities. The platform utilizes a user-friendly map-based interface, allowing users to navigate through the campus and find the resources they need with ease. Real-time updates ensure that users have up-to-date information about the availability and status of these resources, making their campus life more efficient and productive.

Moreover, Cambuzz offers seamless connectivity and accessibility across different devices. Whether students are using their laptops, smartphones, or tablets, they can easily access Cambuzz and stay connected to their university community. This inclusivity ensures that users can engage and interact with the platform anytime, anywhere, enabling them to stay informed, connected, and engaged while on campus. By providing a unified platform for communication, collaboration, and resource discovery, Cambuzz streamlines the university experience for all members of the community. It fosters a sense of community and encourages meaningful connections among students and faculty members. With Cambuzz, the university community can effectively share knowledge, collaborate on projects, and discover valuable resources, ultimately enhancing the overall educational journey for everyone involved.

TABLE OF CONTENTS

TITLE	PAGE NO
ACKNOWLEDGEMENT	iv
SYNOPSIS	v
LIST OF FIGURES	viii
1. INTRODUCTION	1-3
1.1 ABOUT THE PROJECT	1
1.2 PLAN OF REPORT	2
2. PROBLEM DEFINITION & FEASIBILITY ANALYSIS	4-6
2.1 PROBLEM DEFINITION	4
2.2 EXISTING SYSTEM	4
2.3 PROPOSED SYSTEM	4
2.4. FEATURES	5
2.5. FEASIBILITY ANALYSIS	6
2.5.1 Operational Feasibility	6
2.5.2 Technical Feasibility	6
2.5.3 Economic Feasibility	6
3. SOFTWARE REQUIREMENT SPECIFICATION	7-9
3.1 SYSTEM REQUIREMENTS	7
3.1.1 Hardware Specification	7
3.1.2 Software Specification	7
3.2 USER REQUIREMENTS	8
3.2.1 Functional Requirement	8
3.2.2 Non Functional Requirement	8
3.2.3 Performance Requirement	8
3.3 EXPLANATION OF TECHNOLOGIES USED	9
3.3.1 Axios	9
3.3.2 Mutation	9
3.3.3 Google Maps API	9
3.3.4 Firebase Authentication	9
3.3.5 JWT	9

4. SYSTEM ANALYSIS	10-11
4.1 INTRODUCTION	10
4.2 DATA FLOW DIAGRAM	10
4.3 USE CASE DIAGRAM	11
5. SYSTEM DESIGN	12-23
5.1 ARCHITECTURE OF THE SYSTEM	12
5.2 DETAILED DIAGRAM	13
5.2.1 Detailed class diagram	14
5.2.2 Module-wise class diagram	14-15
5.2.3 Sequence diagram	16-18
5.3 MODULES DESCRIPTION	19-23
5.3.1 Register	19
5.3.2 Login	19
5.3.3 Social Media	20
5.3.4 Resources	21
5.3.5 Chats & PU Forum	22
5.3.6 Admin Portal	23
6. CODING, TESTING, AND IMPLEMENTATION	24-29
6.1 INTRODUCTION	24
6.2 IMPLEMENTATION	24-26
6.3 TESTING	27
6.3.1 Unit Testing	27
6.3.2 Validation Testing	27
6.3.3 Functional Testing	28
6.3.4 GUI Testing	29
7. CONCLUSION AND FUTURE WORK	30
7.1 CONCLUSION	30
7.2 FUTURE WORK	30
REFERENCES	31
APPENDICES	32
APPENDIX A: SAMPLE SCREENSHOTS	32-37

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
4.1	Data Flow Diagram	10
4.2	Use Case Diagram	11
5.1	Detailed Class Diagram	14
5.2	Module-wise Class Diagram -Navbar	14
5.3	Module-wise Class Diagram -Sidebar	15
5.4	Module-wise Class Diagram -Main Content	15
5.5	Module-wise Class Diagram -Main Content2	15
5.6	Sequence Diagram -User Login	16
5.7	Sequence Diagram -View Post	16
5.8	Sequence Diagram -View Profile	17
5.9	Sequence Diagram -User Follow/Unfollow	17
5.10	Sequence Diagram -Profile Update	17
5.11	Sequence Diagram -Share Post	18
5.12	Sequence Diagram -Comment	18
5.13	Sequence Diagram -Search	18
6.1	Screenshot- Login, Home (Mobile)	25
6.2	Screenshot- Resources (Desktop)	26
6.3	Screenshot- Profile (Desktop)	26
8.1	Screenshot- Chats - PU forum(Mobile)	32
8.2	Screenshot- Home (Desktop)	33
8.3	Screenshot- Resources (Mobile)	34
8.4	Screenshot- Resource Map (Desktop)	35
8.5	Screenshot- Search (Desktop)	36
8.6	Screenshot- Update Profile (Desktop)	37

CHAPTER 1

INTRODUCTION

1.1 ABOUT THE PROJECT

Social media platforms have become an integral part of modern-day communication and collaboration, with an ever-increasing number of people turning to these platforms to connect with others, share information and collaborate on various initiatives. Universities, with their large and diverse communities of students and faculty, are no exception to this trend. However, despite the abundance of social media platforms available today, there is still a need for a dedicated platform tailored to the unique needs of university communities.

In response to this need, this project aims to develop an advanced social media website for students and faculty of a university community. The platform will serve as a central hub for information sharing, collaboration, and communication, providing users with a wide range of features and functionalities tailored to their needs.

One of the key features of the platform will be the campus resource locator, which will help students find and use a variety of campus resources, including computer laboratories, study spaces, and printing facilities. The resource locator will be based on a map-based user interface and provide real-time updates, making it easy for students to locate and access the resources they need.

Overall, the platform will provide an easy-to-use, accessible, and uniform platform for communication, collaboration, and resource discovery, significantly improving the experience for everyone involved in the university community. This report will provide a detailed overview of the development process, the key features and functionalities of the platform, and the potential benefits for users.

1.2 PLAN OF REPORT

Chapter 1: Introduction and About the Project

This chapter serves as an introduction to the project and provides background information about it. It begins by presenting an overview of the project, explaining its purpose, and highlighting its significance. It then delves into the specifics of the project, discussing its origins, motivation, and the problem it aims to address. By combining the introduction and the "About the Project" section, this chapter sets the stage for the rest of the report, providing a comprehensive understanding of the project's context and objectives.

Chapter 2: Problem Definition and Feasibility Analysis

In this chapter, the problem statement is defined in detail. It articulates the specific issue or challenge that the project aims to solve. Additionally, the existing system is examined, outlining its limitations and shortcomings. The proposed system is then introduced, highlighting its advantages and improvements over the existing system. The chapter concludes with a feasibility analysis, evaluating the project's operational, technical, and economic feasibility to determine its viability.

Chapter 3: Software Requirement Specification

This chapter combines the Software Requirement Specification and Explanation of Technologies Used sections. It outlines the software requirements for the project, including hardware and software specifications. Additionally, it provides a concise explanation of the technologies employed, such as Axios, Mutation, Google Maps API, Firebase Authentication, and JWT (JSON Web Tokens). By merging these sections, this chapter presents a comprehensive overview of the project's technical requirements and the technologies that facilitate its implementation.

Chapter 4: System Analysis

This chapter focuses on the analysis of the system. It begins by explaining the concept of system analysis and the methodology employed for analysis. The chapter then presents the data flow diagram, illustrating the movement of data within the system and showcasing the system's functionality. It also includes a use case diagram, highlighting the system's various use cases and interactions with its users.

Chapter 5: System Design

The system design chapter provides insights into the architecture and design of the project. It describes the overall system architecture, detailing its components and their relationships. A comprehensive system diagram is provided to visualize the system's structure. Additionally, a detailed class diagram is included to showcase the various classes and their associations. The chapter also features module-wise class diagrams and sequence diagrams, illustrating the interactions between different system components during specific processes.

Chapter 6: Conclusion and Future Work

The conclusion chapter summarizes the key findings and outcomes of the project. It recaps the main points discussed throughout the report, highlighting the accomplishments and contributions of the project. The chapter presents the conclusions drawn from the project's implementation and discusses the potential for future work and enhancements to further improve the system's functionality and address any remaining challenges.

References and Appendices

This chapter includes two sections, References and Appendices, which provide supplementary information and supporting materials for the project. The References section lists all the sources and references used throughout the report, ensuring proper citation and acknowledging the contributions of external works. The Appendices section contains additional resources such as charts, graphs, tables, diagrams, screenshots, or any other relevant documentation that provides further insights and enhances the understanding of the project. These sections collectively offer readers a comprehensive collection of resources and additional materials to explore and deepen their understanding of the project.

CHAPTER 2

PROBLEM DEFINITION & FEASIBILITY ANALYSIS

2.1 PROBLEM DEFINITION

In today's world, social media platforms have become an integral part of daily life. They are a vital tool for communication, collaboration and information sharing. Social media has also been widely adopted in university communities to facilitate communication and collaboration among students and faculty members. However, the current social media platforms are not designed to meet the specific needs of a university community. These platforms lack the specialised features required to cater to the specific needs of students, faculty members, and other individuals associated with a university. This results in an inefficient and unsatisfactory experience for users.

2.2 EXISTING SYSTEM

The current social media platforms lack specialised features and functionalities required to cater to the specific needs of a university community. They do not provide a specialised environment for students, faculty members, and other campus-related individuals to interact and share information about the university. As a result, users experience an inefficient and unsatisfactory social media experience.

Moreover, navigating and locating resources on a university campus can be difficult due to the wide range of resources and services spread across different locations. This can lead to frustration and ultimately impact the quality of the academic experience.

2.2 PROPOSED SYSTEM

To address the issues faced by the existing social media platforms, an advanced social media website, Cambuzz, has been developed. Cambuzz is designed to serve as a one-stop-shop for all campus-related information and resources, providing a specialised environment for students, faculty members, and other campus-related individuals to interact and share information about the university.

Cambuzz offers specialised features and functionalities that cater to the specific needs of the university community, such as campus resource locators, event notifications, discussion forums, and much more. By offering these specialised features, Cambuzz provides an efficient and satisfying social media experience for users, ultimately improving the quality of their academic experience.

2.3 FEATURES

1. Social Media Platform: Cambuzz provides a social media platform for students and faculty members to interact and connect with each other. This platform enables users to share information, create groups, and participate in discussions.
2. Academic Activities Coordination: Cambuzz enables faculty members to coordinate academic activities and share information related to their courses. This platform provides a centralised location for students to access course information and communicate with their professors.
3. PU Forum: Cambuzz offers a forum for the Pondicherry University community to discuss campus-related issues. This platform enables students, faculty members, and staff to share their thoughts, ideas, and opinions on a range of topics.
4. Chat: Cambuzz provides a chat feature that enables students and faculty members to interact with each other in real-time. This feature facilitates communication and collaboration within the university community.
5. Resource Locator: Cambuzz offers a campus resource locator that enables students to find resources such as food, toilets, printers, and other services on campus. This feature provides a convenient and efficient way for students to access the resources they need.
6. Events: Cambuzz enables users to create, discover and RSVP to events happening on campus. Users can filter by category, date, and location to find events of interest.
7. Notifications: Cambuzz sends notifications to users about upcoming events, deadlines, and other important information to ensure they are up to date with what's happening on campus.

2.4 FEASIBILITY ANALYSIS

Overall, Cambuzz has high feasibility in terms of operational, technical, and economic aspects, making it a viable project that will bring significant benefits to the university community.

2.4.1 Operational Feasibility:

Cambuzz is operationally feasible as it addresses the key challenges faced by students and faculty members in accessing and sharing information. With its comprehensive features, it will make communication, coordination, and resource discovery easy for users. The platform will also enhance collaboration between students and faculty members, leading to increased productivity and a better overall experience.

2.4.2 Technical Feasibility:

Cambuzz is technically feasible as it utilises modern technologies and programming languages to develop its web application. The platform will be designed with user-friendly interfaces, and the website will be accessible from various devices, including mobile phones and computers. The implementation of cloud-based architecture and cybersecurity measures will ensure that the platform is secure and can handle high traffic volumes.

2.4.2 Economic Feasibility:

Cambuzz is economically feasible as it offers cost-effective solutions to improve communication and collaboration within the university community. The initial investment required for the development of the platform will be high, but the benefits it will provide will outweigh the costs. The platform will help reduce administrative costs, save time, and increase efficiency for students and faculty members. The revenue generated from advertising, marketplace and promotional services will help offset the initial investment and provide sustainable revenue streams. Overall, the platform's benefits and revenue generation capabilities make it economically feasible.

CHAPTER 3

SOFTWARE REQUIREMENT SPECIFICATION

3.1 SYSTEM REQUIREMENTS

3.1.1 Software Specification

- Operating System: Windows 10 or above, macOS, or Linux
- Web Server: Node server
- Web Development Framework: MERN stack (with MySql)
- Programming Languages: ReactJs, NodeJs, Express Js, HTML, SCSS
- Relational Database Management System: MySQL
- Version Control System: Git

3.1.2 Hardware Specification

- Processor: Intel Core i5 or above or equivalent AMD processor
- RAM: 4GB or above
- Storage: 256GB SSD or above
- Internet Connection: Broadband connection with a minimum speed of 5 Mbps
- Display: Recommended 13-inch screen with a resolution of 1920x1080

3.2 USER REQUIREMENTS

3.2.1 Functional Requirement

1. The web application must provide a user-friendly interface for students, faculty, and staff to access and share information.
2. The platform should allow users to create and join groups based on interests, majors, or academic programs.
3. The web application must provide a searchable campus resource locator for students to find and access resources.
4. The platform should support real-time messaging and chat features to facilitate communication and collaboration between users.
5. The web application should allow faculty to create and manage course materials, assignments, and assessments.

3.2.2 Non-Functional Requirement

1. The web application must be compatible with various browsers and devices.
2. The platform should support secure login and authentication for all users.
3. The web application must comply with data protection laws and privacy policies.
4. The platform should have a simple and intuitive navigation system.
5. The web application should support multiple languages for users who are not fluent in English.

3.2.3 Performance Requirement

1. The web application should load quickly and provide a responsive user interface.
2. The platform should be able to handle a large number of concurrent users without slowing down or crashing.
3. The web application must provide accurate and up-to-date information on campus resources and events.
4. The platform should be able to handle large amounts of data, including multimedia content such as images and videos.
5. The web application should be able to handle peak loads during registration periods, exam periods, and other high-traffic times.

3.3 EXPLANATION OF TECHNOLOGIES USED

3.3.1 Axios

Axios [2] is a popular JavaScript library used for making HTTP requests from web browsers or Node.js. It provides an easy-to-use interface for performing AJAX requests, handling request and response data, and handling errors. In CamBuzz client and admin communicate with the server using axios.

3.3.2 Mutation

Mutation [3] is a state management library for React that allows developers to manage state in a predictable and efficient way. It provides a simple and easy-to-use API for updating and accessing state data, and it integrates well with other popular React libraries such as Redux and MobX. User posts and stories and profile details are loaded using this without the need of reloading the webpage.

3.3.3 Google Maps API

Google Maps API is a set of APIs [7] provided by Google that allows developers to integrate Google Maps functionality into their web applications. It provides a wide range of features such as mapping, location search, and route planning. Cambuzz uses maps api and directions api to get the location and directions to the resources.

3.3.4 Firebase Authentication

Firebase Authentication [8] is a service provided by Google that allows developers to easily add user authentication to their web applications. It provides a simple and secure way for users to sign up, log in, and manage their user accounts. User email verification in cambuzz is succeeded through this.

3.3.5 JWT

JSON Web Tokens (JWT) [5] are a standard for representing and transmitting secure authentication and authorization data between parties. JWTs [4] are used to verify the identity of users and to authorize access to protected resources in web applications. Each user is provided with the corresponding token for the type of user.

CHAPTER 4

SYSTEM ANALYSIS

4.1 INTRODUCTION

Cambuzz is an advanced social media website designed to provide a specialised environment for students, faculty, and other campus-related individuals to interact and share information about the university. The platform is equipped with a variety of features such as social media profiles, academic coordination tools, campus event forums, chat rooms, and a resource locator to find various resources on campus. Cambuzz uses the React framework and Node.js to run the server and client, and integrates technologies such as Axios, Mutation, Google Maps API, Firebase Authentication, and JWT for efficient and secure data transmission. The system has undergone thorough feasibility analysis and has been designed to meet all functional, non-functional, and performance requirements. Cambuzz aims to significantly improve the university experience for all stakeholders by providing a unified platform for communication, collaboration, and resource discovery.

4.2 DATA FLOW DIAGRAM

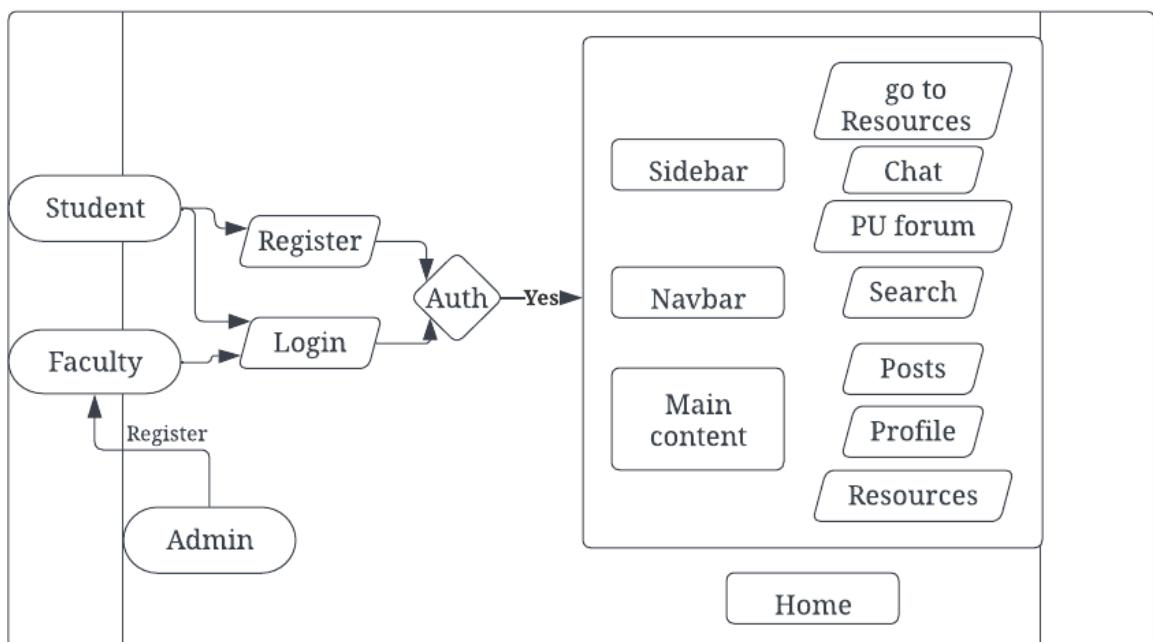


Figure 4.1 DFD

4.3 USE CASE DIAGRAM

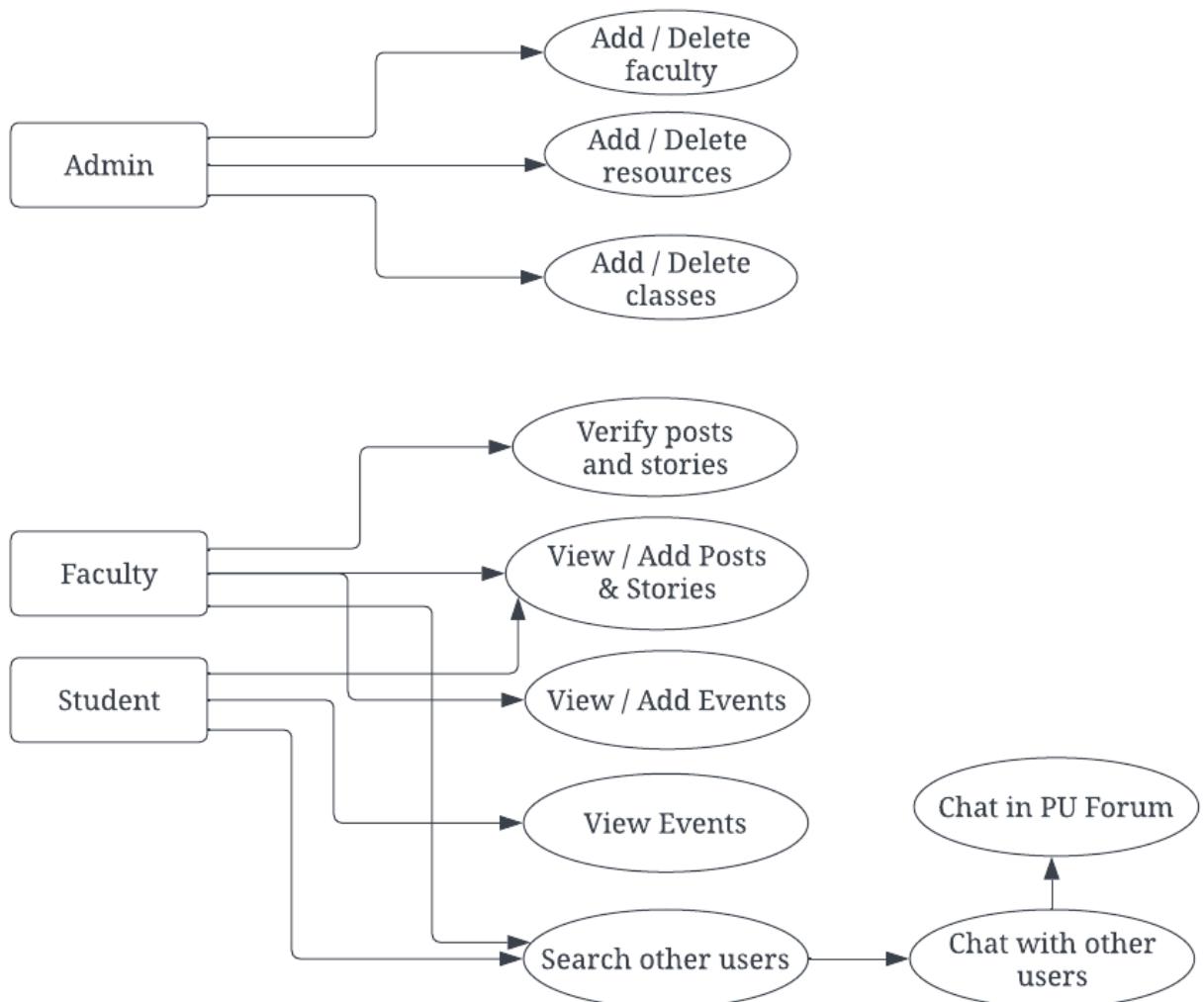


Figure 4.2 Use Case Diagram

CHAPTER 5

SYSTEM DESIGN

5.1 ARCHITECTURE OF THE SYSTEM

1. **Front-end:** The website's front-end would be built using React, which is a popular JavaScript library for building user interfaces. React provides a modular and scalable approach to building user interfaces and can handle complex state management. The front-end would also use HTML, SCSS, and JavaScript to create the website's layout and styling. Axios library is used to link with the api.
2. **Back-end:** The website's back-end would be built using Node.js and Express, which is a fast and lightweight web application framework for Node.js. The back-end would be responsible for handling user authentication, database interactions, and API endpoints. The back-end would use MySQL as the primary database to store user details, posts, and resources. Files would be stored on the client server itself. [6]
3. **Authentication:** The website would use a secure authentication mechanism such as JSON Web Tokens (JWT) to authenticate users and protect sensitive data. Users would be 2 types, namely student and faculty. Email verification would be done by firebase authentication.
4. **Database:** The website would use MySQL as the primary database to store user details, posts, and resources. A popular open-source relational database management system that is used to store and manage data for web applications.
5. **UI components:** The website's UI would be divided into three main sections: navbar, sidebar, and main content section. The navbar would contain buttons to go to the home page, change the theme to dark and light, view the user profile, and search bar. The sidebar would contain a section that navigates to the resources page and a section for chatting between users. And a forum where people can discuss and ask questions will also be available in the chats section. Main content section would be showing posts from different users which the current user is following and also the posts of the current user.
6. **Resources:** The website would use Google Maps [1] API to display the location of resources when a user clicks on a specific resource. The website would also provide search functionality to enable users to find specific resources. Users can update new changes regarding the resources if any and faculties would verify the changes and reflect on the website.

5.2 DETAILED DIAGRAM

The Navbar and Sidebar classes are important components of the user interface of the Cambuzz website. These classes are responsible for displaying the navigation menu and sidebar on the website, respectively. The Navbar class is responsible for showing the logo of Cambuzz and the menu options such as home, profile, notifications, resources, and logout. The Sidebar class shows additional options for users such as editing their profile, creating posts, and browsing different categories of resources. The Main Page class is responsible for displaying the user's posts, user profile, and available resources. This class acts as the main dashboard of the website. It displays the user's profile picture, cover picture, and other relevant information about the user. The user can also browse through their own posts and posts made by others on this page.

The User class represents the user's attributes that are stored in the database, such as username, password, email, profile picture, cover picture, website, and posts/resources. The User class interacts with the database to retrieve and update the user's information. The Post class represents each post made by users of the website. The attributes of each post include the Post ID, username of the poster, image, description, comments, and likes. The Post class interacts with the database to store, retrieve, and modify posts made by users. The Resource class represents the available resources on campus. The attributes of each resource include the ID, name, address, and location. Resources are further categorised based on their type and location. The Resource class interacts with the Google Maps API to display the location of the resources on the map. Each resource is also associated with a set of location coordinates and availability status, which is also stored in the database.

(Refer figure 5.1)

5.2.1 Detailed Class Diagram

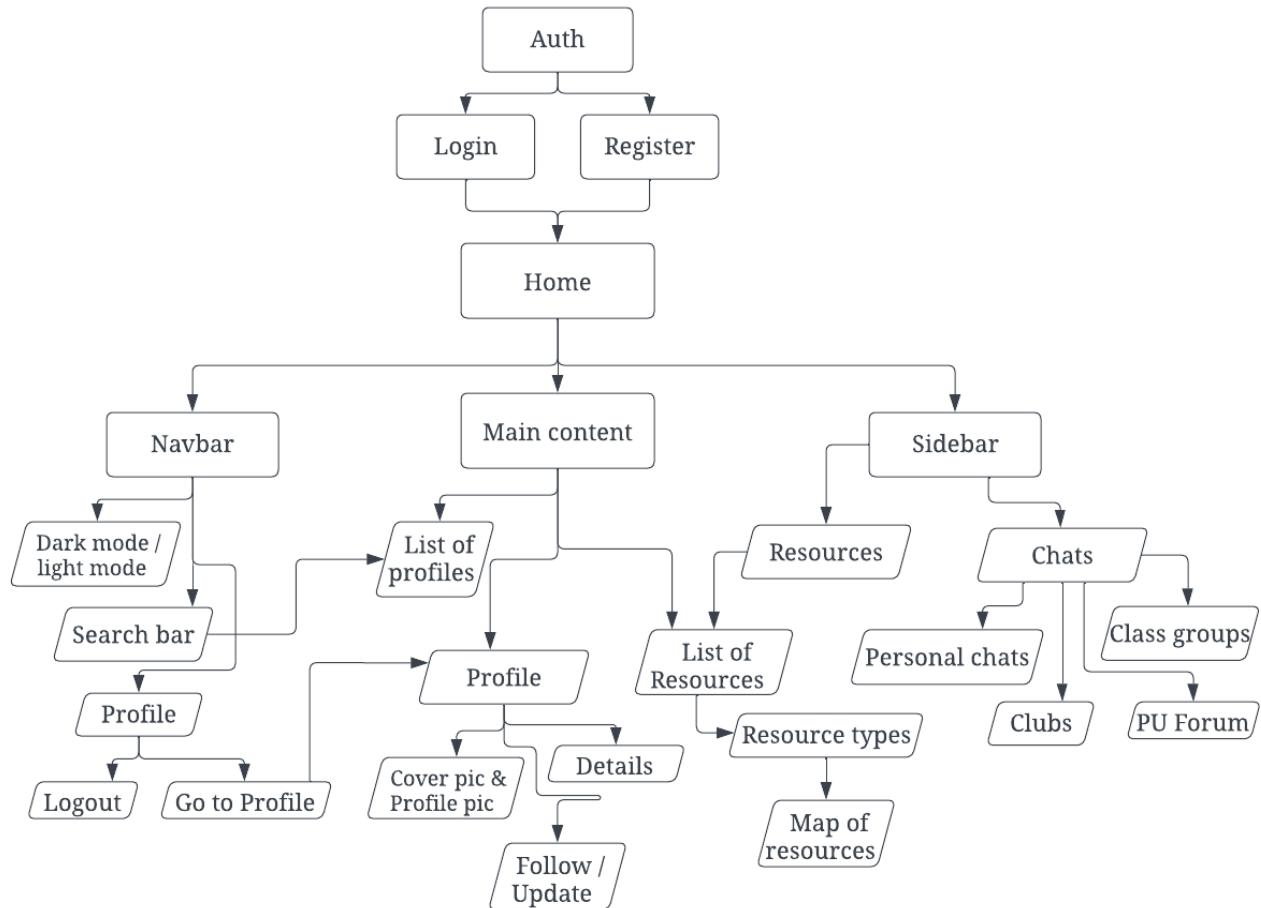


Figure 5.1 Detailed Class Diagram

5.2.2 Module-wise Class Diagram

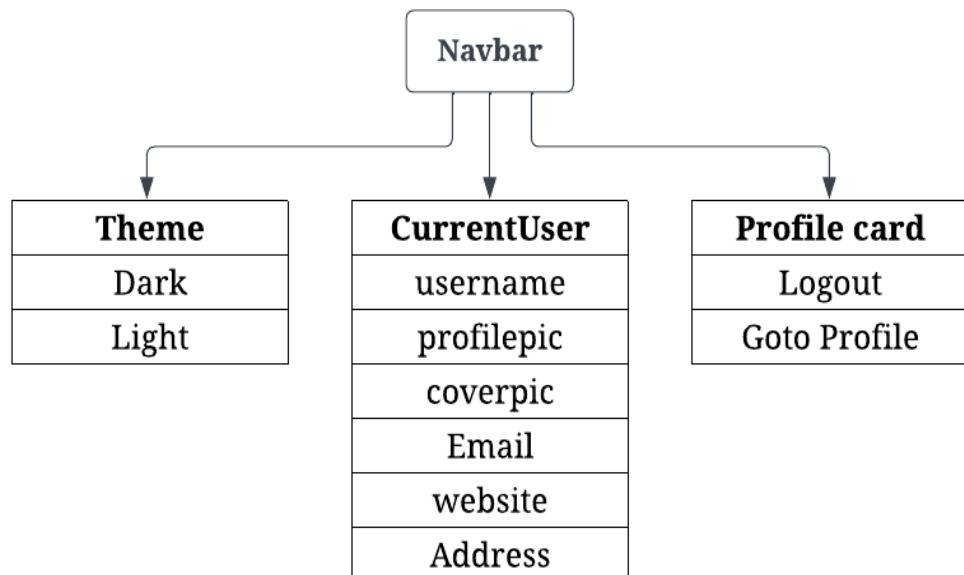


Figure 5.2 Module-wise Class Diagram -Navbar

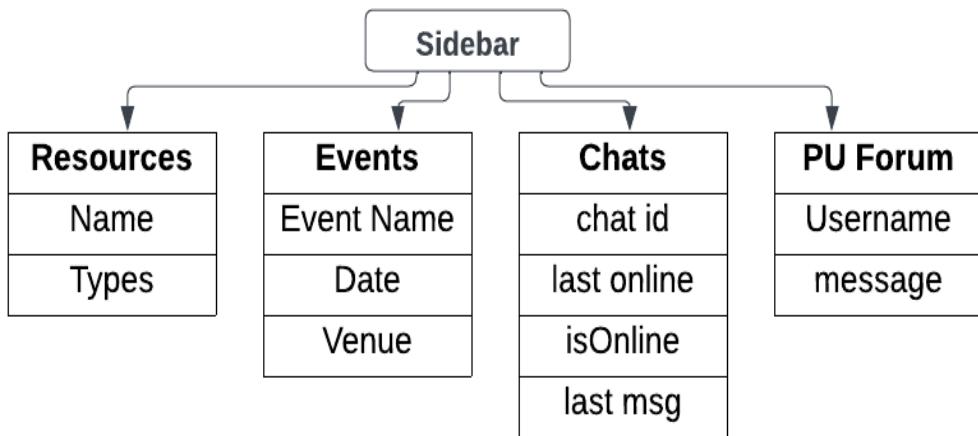


Figure 5.3 Module-wise Class Diagram -sidebar

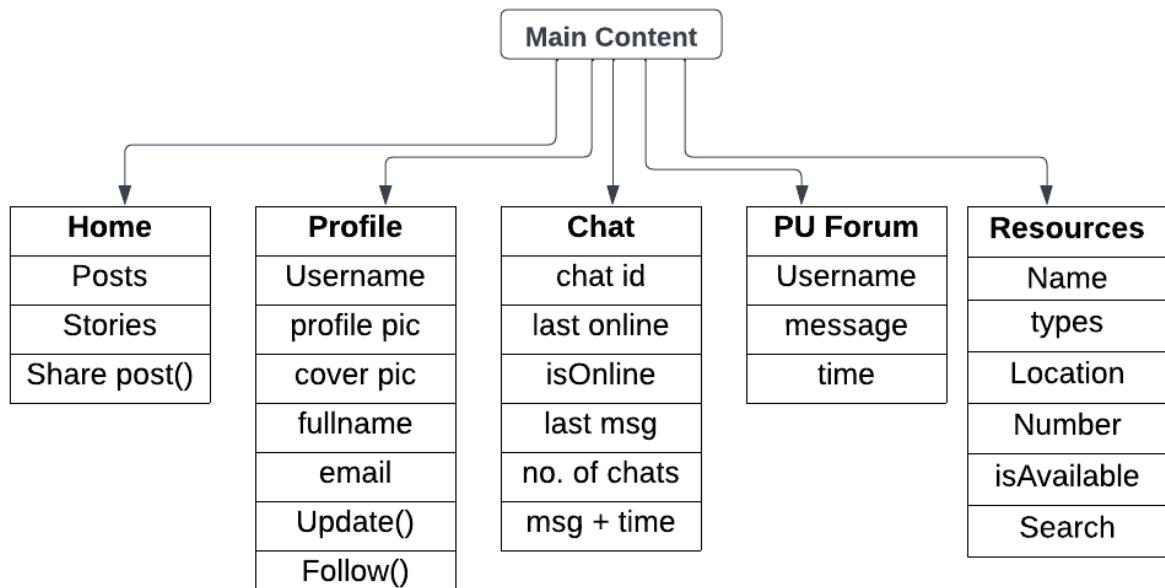


Figure 5.4 Module-wise Class Diagram -Main content

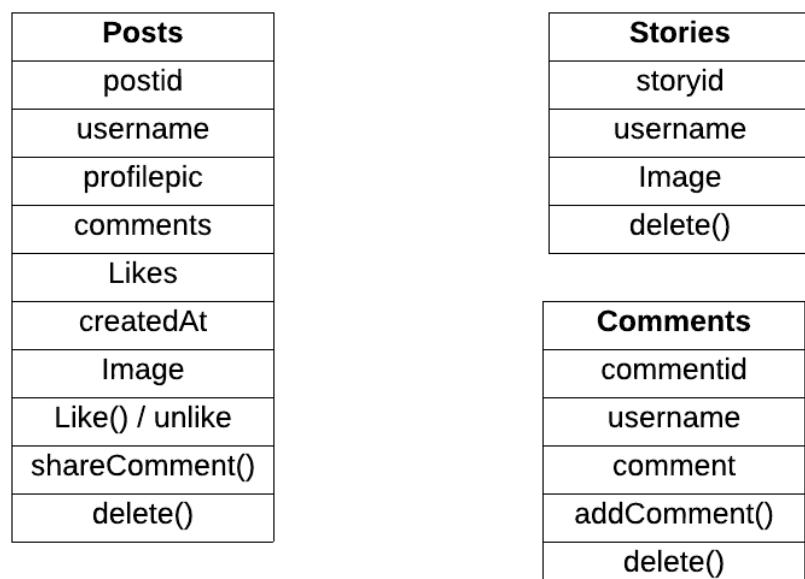


Figure 5.5 Module-wise Class Diagram -Main content2

5.2.3 Sequence Diagram

1. User login

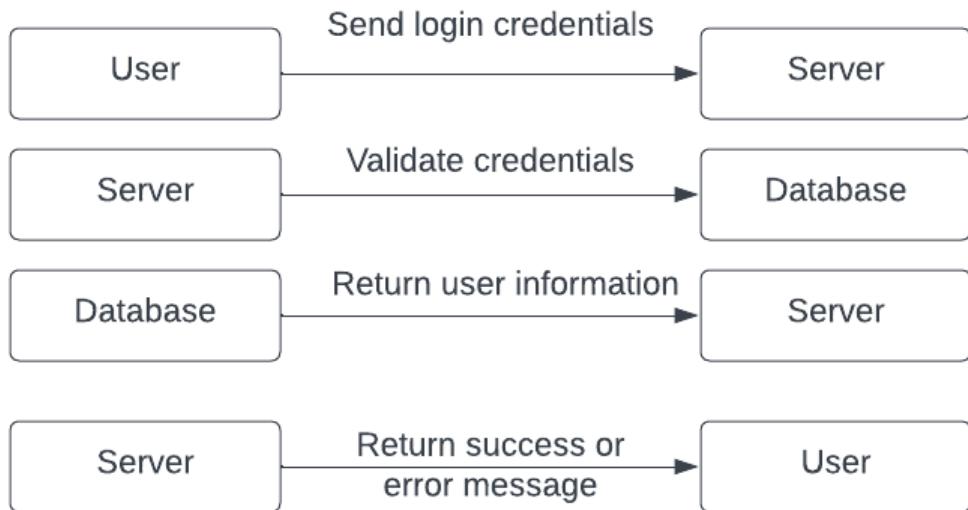


Figure 5.6 Sequence Diagram -User Login

2. View posts

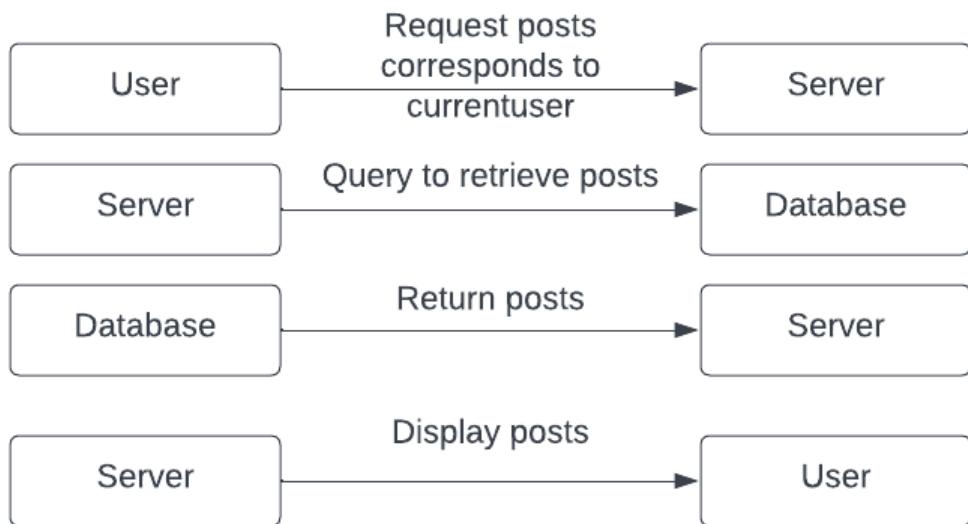


Figure 5.7 Sequence Diagram -View Posts

3. View profile

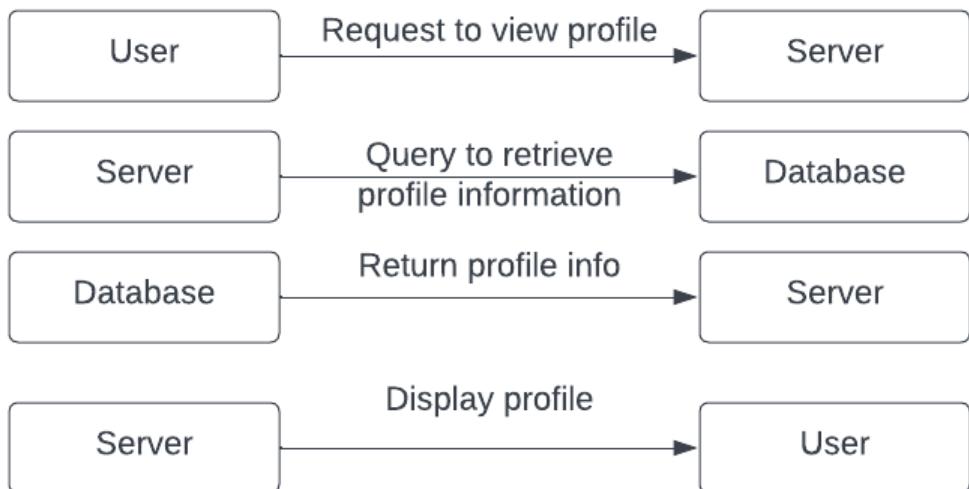


Figure 5.8 Sequence Diagram -View profile

4. User follow or unfollow another user

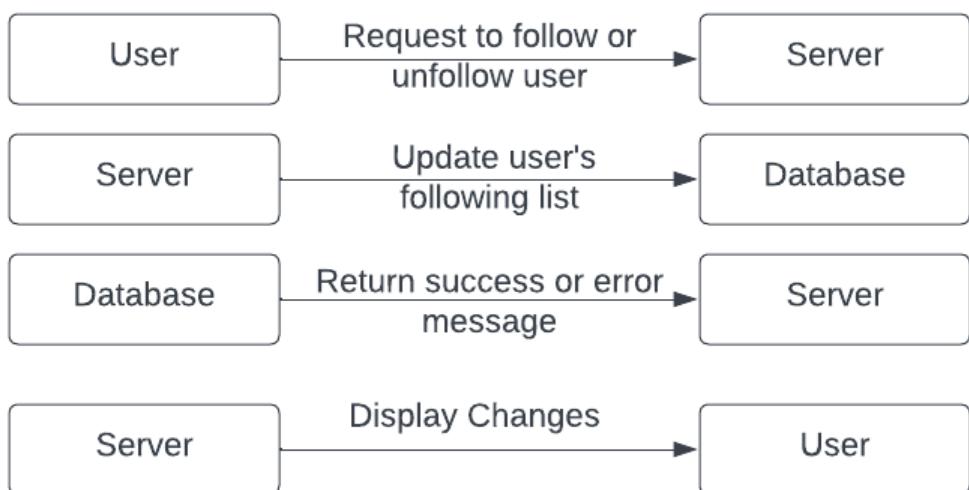


Figure 5.9 Sequence Diagram -User follow/unfollow

5. Profile update

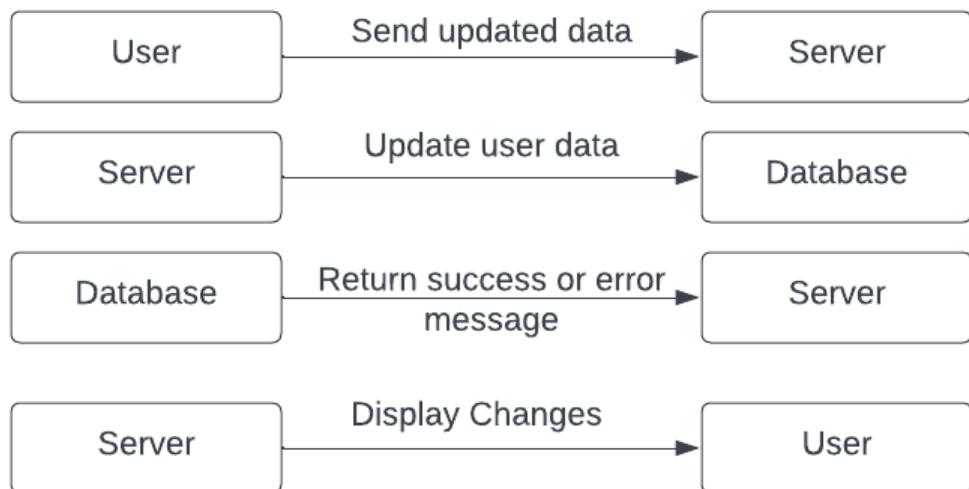


Figure 5.10 Sequence Diagram -Profile update

6. Share post

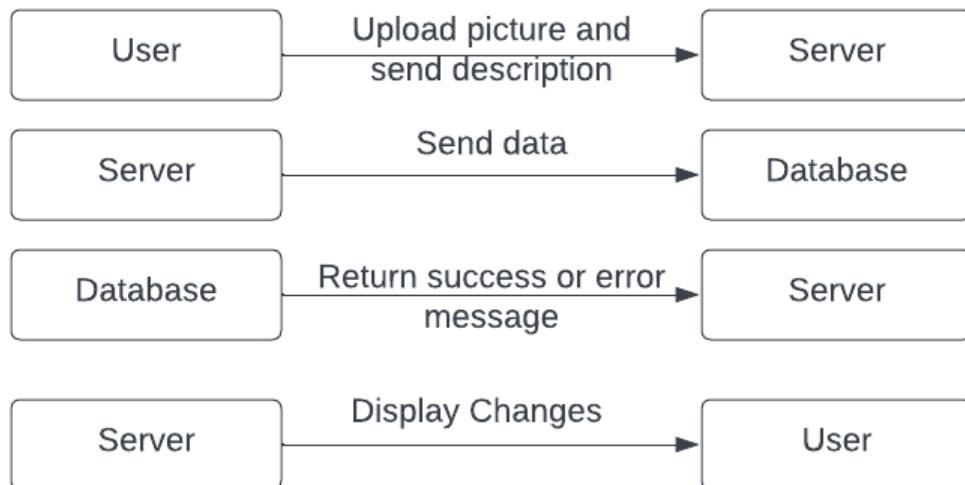


Figure 5.11 Sequence Diagram -Share post

7. Comment on Post

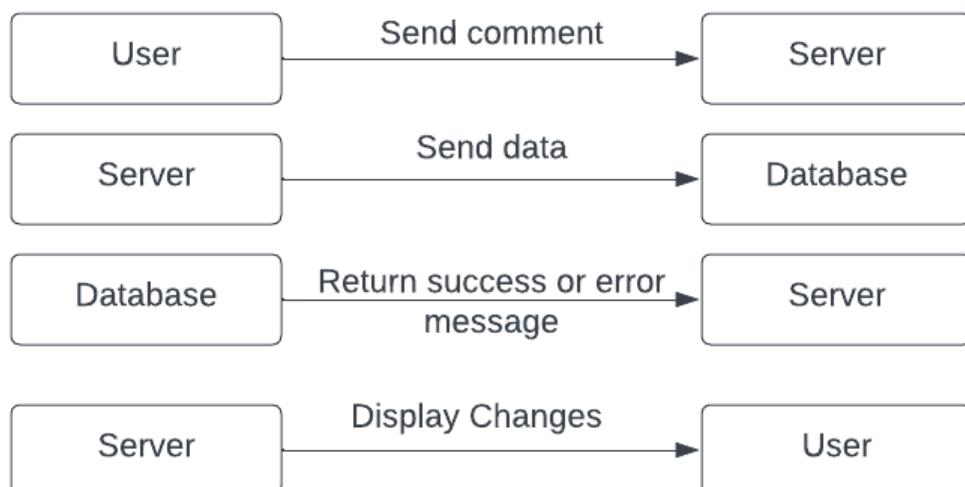


Figure 5.12 Sequence Diagram -Comment

8. Search

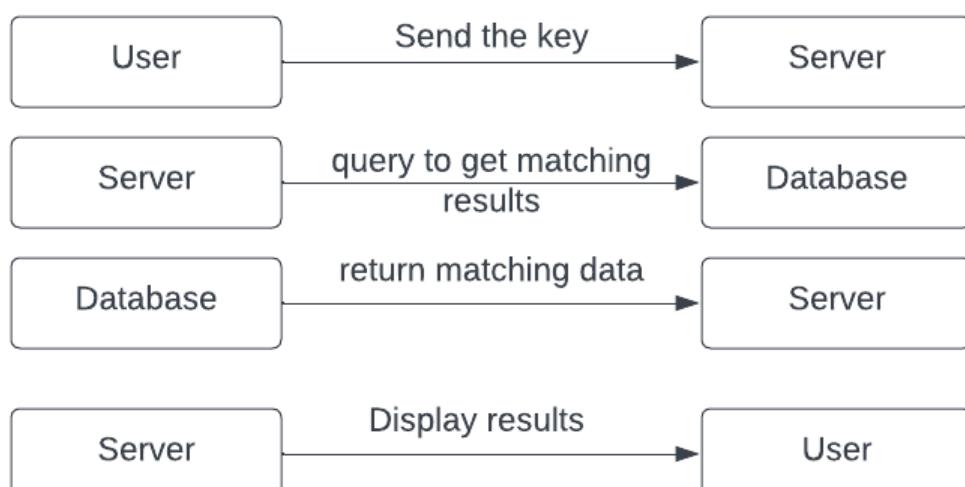


Figure 5.13 Sequence Diagram - Search

5.3 MODULES DESCRIPTION

5.3.1 Register

Registration is only for students. It begins by validating the provided email address to ensure it matches the expected format for a Pondicherry University student. If the email is valid, it queries the database to check if a user with the same username already exists. If a conflict is found, it returns an appropriate response with a status code of 409. If the username is available, the function proceeds to hash the provided password using bcrypt for secure storage. It then constructs an SQL query to insert the user's username, email, hashed password, and type into the users table. The function creates a new user account using the createUserWithEmailAndPassword method and sends a verification email to the user. If the user creation and email verification process is successful, the user data is inserted into the database, and a 200 status response is returned with a success message. In case of any errors during the process, the function returns the corresponding error message along with an appropriate status code. Overall, the register module ensures proper validation, secure password storage, user creation, and database integration for a smooth user registration process.

5.3.2 Login

Login is available for faculty and students. The login module is a server-side function responsible for handling user login requests and authentication. It begins by querying the database to retrieve user data based on the provided username. If the user is not found in the database, it returns a 404 status response with an appropriate error message. If the user is found, it checks if the user's type is "student" to ensure that only students can proceed with login. If the user's type is not "student", it returns a 404 status response with an error message indicating that the user is not a student.

To validate the user's credentials, the module compares the provided password with the stored hashed password using bcrypt. If the password is incorrect, it returns a 400 status response with an error message. On successful authentication, the module signs in the user using the signInWithEmailAndPassword method, passing the user's email and password.

To enhance security, the module generates a JSON Web Token (JWT) that contains the user's username as the payload and signs it with a secret key. The response is then prepared by excluding the password field from the user data and including the emailVerified field obtained from the sign-in process. For convenient access control, the module sets the access token as an HTTP-only cookie in the response. Finally, a 200 status response is sent back to the client along with the user data (excluding the password) as a JSON response. Overall, the login module efficiently handles user authentication, password verification, and token generation, ensuring secure access to protected resources.

5.3.3 Social Media

The social media module in Cambuzz offers a comprehensive set of features to enhance user engagement and interaction. It primarily focuses on three key components: Posts, Stories, and Events. In the Home tab, the module provides a personalized experience by showcasing stories from users whom the current user follows. Stories are temporary photos or videos that allow users to share moments from their day. These stories are displayed at the top of the page and disappear after 24 hours, creating a sense of urgency and exclusivity. Users can view these stories and have a glimpse into the lives of the people they follow.

Alongside stories, the module presents a feed of posts from users whom the current user follows. Posts consist of a photo or image along with a description. This allows users to share their thoughts, experiences, or any interesting content with their followers. The feed presents a curated collection of posts, ensuring that users see updates from the accounts they are interested in. Users have the ability to engage with posts by liking and commenting on them. Liking a post shows appreciation or agreement, while commenting allows users to share their thoughts, ask questions, or engage in discussions with the post's author and other users. This interactive element fosters a sense of community and encourages social connections.

The Profile page provides a dedicated space for each user, showcasing their details and posts. Users can view the profile of other users to learn more about them and explore their posted content. This includes the ability to view their posts, providing a comprehensive overview of their activities and interests. Additionally, users can choose to follow or unfollow other users directly from their profiles, allowing them

to manage their social connections and control the content they see on their feed. Overall, the social media module in Cambuzz offers a dynamic and engaging platform for users to connect, share, and interact through posts, stories, and events. It provides a personalized experience tailored to the user's interests and preferences while fostering social connections and community engagement.

In addition to the Posts and Stories features, the social media module in Cambuzz also includes an Events component. Events are added by faculty members, providing a platform for them to share information about upcoming activities, workshops, seminars, or any other events relevant to the university community. Faculty members have the privilege to create and post events through the module. They can provide details such as the event title, date, time, location, description, and any additional relevant information. This allows faculty members to effectively communicate and disseminate information about important events happening within the university. Users, including students and other faculty members, can access the Events section to view the upcoming events. They can browse through the event listings, read the descriptions, and gather all the necessary details. This helps users stay informed about various activities and plan their participation accordingly.

5.3.4 Resources

The Resources module in Cambuzz provides a comprehensive directory of various resources available within the university campus. It aims to assist users in locating and accessing different facilities and amenities conveniently. The module includes categories such as water, food outlets, toilets, hostels, departments, and more, each offering detailed information about the respective resources.

Within each category, users can explore the available resources and access specific details. This may include the names, locations, opening hours, contact information, and any additional relevant information about each resource. Users can quickly find the nearest resource by sorting them based on their distance from the current location, ensuring convenience and efficiency. The integration of maps enhances the user experience by allowing users to visualize the location of each resource. Users can view individual resources on the map, enabling them to navigate to their desired destination effectively.

Additionally, an option to view all resources within a category on the map provides a comprehensive overview, enabling users to plan their routes or identify nearby facilities easily.

One specific category, such as Bus Stops, includes additional functionality. It not only provides information about the bus stop locations but also displays the bus timings from each stop. This assists users in planning their transportation within the campus, ensuring they have access to accurate and up-to-date bus schedules. The Resources module aims to simplify the process of finding and accessing various amenities and facilities within the university campus. It leverages categorization, distance sorting, mapping, and additional features like bus timings to provide a user-friendly and efficient resource directory. By centralizing information and offering convenient navigation, the module enhances the overall user experience and ensures that users can easily locate and utilize the resources available on campus.

5.3.5 Chats and PU Forum

The Chats module in Cambuzz enables users to engage in real-time conversations with each other. It serves as a platform for direct communication and interaction among users. Additionally, within the Chats module, there is a specific feature called PU Forum, which facilitates open discussions and information sharing among all Cambuzz users. The Chats module allows users to initiate one-on-one conversations with other users. They can exchange messages, share thoughts, ideas, and information privately. This feature promotes direct communication, enabling users to connect, collaborate, and build relationships within the Cambuzz community.

Within the Chats module, the PU Forum stands out as a distinct component. It serves as a public chat room or forum where all Cambuzz users can participate. Users have the opportunity to share messages, thoughts, and valuable information openly with other users. This feature fosters a sense of community and encourages discussions on various topics of interest to the Cambuzz user base. The PU Forum acts as a centralized hub for communication and information sharing, enabling users to engage in conversations with a wider audience. It allows users to seek advice, share experiences, ask questions, and contribute to the collective knowledge within the Cambuzz community.

Overall, the Chats module provides users with a platform for direct communication, while the PU Forum within the module serves as a public chat room for open discussions and information sharing among all Cambuzz users. This combination of one-on-one conversations and public forum fosters interaction, collaboration, and the exchange of ideas within the Cambuzz community.

5.3.6 Admin Portal

The Admin portal in Cambuzz is a dedicated platform designed for administrative users to manage various aspects of the system. It provides a range of modules that enable the admin to perform administrative tasks efficiently. Two essential modules within the Admin portal are Faculty Management and Class Management, along with Resource Management.

Faculty Management Module:

The Faculty Management module allows the admin to add, edit, and delete faculty members within the system. Admin can input relevant information such as name, contact details, qualifications, and any other necessary details pertaining to faculty members. This module provides the admin with the flexibility to manage faculty profiles, update information when required, and remove faculty members from the system as needed.

Class Management Module:

The Class Management module empowers the admin to add, edit, and delete classes within the system. Admin can create class profiles by specifying details like class name, course, department, and other relevant information. The module offers the admin the ability to modify class details when necessary and remove classes that are no longer active or relevant.

Resource Management Module:

The Resource Management module allows the admin to add, edit, and delete resources available within the Cambuzz system. Admin can categorize resources, such as water, food outlets, toilets, hostels, etc., and input detailed information about each resource, including location, availability, and other relevant details. This module facilitates efficient management of resources, ensuring that the information provided to users is accurate and up to date.

CHAPTER 6

CODING, TESTING, AND IMPLEMENTATION

6.1 INTRODUCTION

The coding, testing, and implementation phase of the Cambuzz system are crucial steps in the development process, as they involve the actual creation of the software product. This phase focuses on translating the software design into a fully operational system by writing code, conducting thorough testing to identify and resolve any errors, and deploying the system in the production environment. This chapter provides an overview of the implementation process, highlighting the technologies employed, the architectural design decisions made, and the testing methodologies utilized to ensure the system's functionality and reliability.

6.2 IMPLEMENTATION

The implementation of the Cambuzz system involves the utilization of MEAN stack (MongoDB, Express.js, Angular, Node.js) with MySQL as the database. Client-server communication is facilitated using the Multer library for storing posts, stories, profile pictures, and cover pictures. The Axios library is employed for HTTP requests between the client and server. React is chosen as the front-end framework for building the website's user interface, providing a modular and scalable approach. HTML, SCSS, and JavaScript are utilized for layout and styling. The React library Mutation is employed for efficient state management, enabling seamless updates and access to user posts, stories, and profile details without page reloading.

The Google Maps API is integrated to provide mapping, location search, and route planning functionalities. This enables users to view resource locations and obtain directions. Firebase Authentication is utilized for secure user authentication, allowing users to sign up, log in, and manage their accounts. Email verification is implemented through Firebase Authentication. JSON Web Tokens (JWTs) are employed as a standard for secure authentication and authorization between parties, verifying user identities and granting access to protected resources.

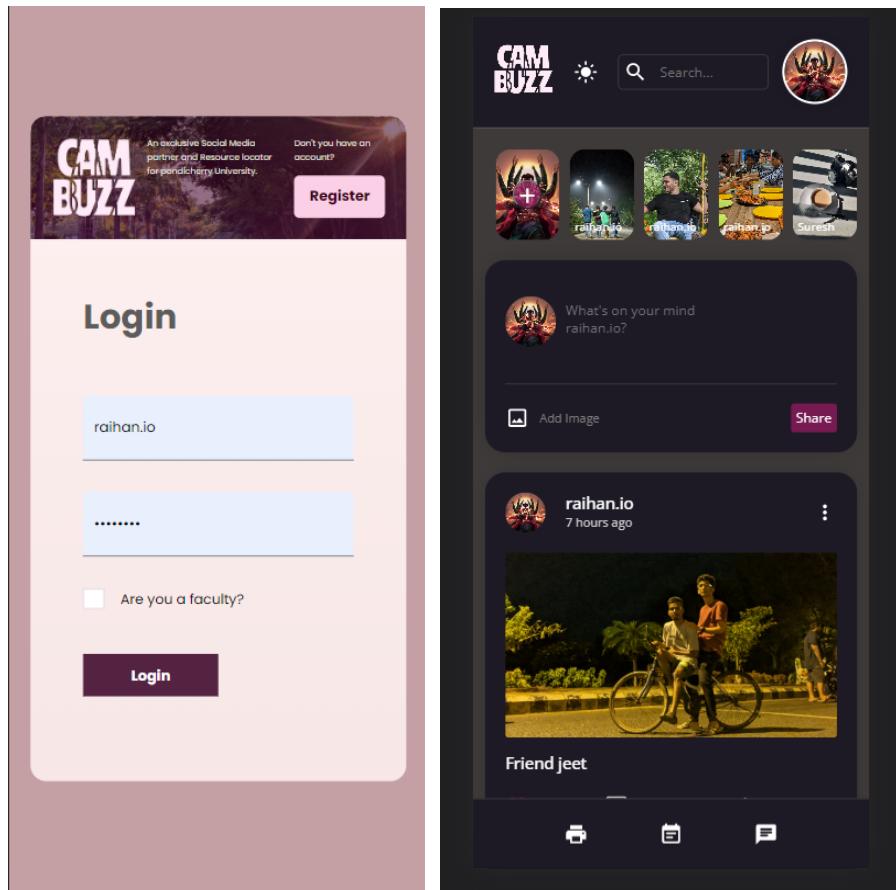


Figure 6.1 Login, Home (Mobile)

On the front-end, React, HTML, SCSS, and JavaScript are used. Axios handles communication with the API. On the back-end, Node.js and Express are employed to handle user authentication, database interactions, and API endpoints. MySQL serves as the primary database for storing user details, posts, and resources. Files are stored on the client server.

The registration process ensures validation of provided email addresses, checks for existing usernames in the database, and securely stores hashed passwords using bcrypt. User data is inserted into the MySQL database, and a verification email is sent using Firebase Authentication [8]. Success or error messages are returned accordingly. The login module handles user login requests and authentication. It queries the database for user data, verifies the user's credentials using bcrypt, and generates a JSON Web Token (JWT) for secure access to protected resources. The response includes the user's data (excluding the password) and sets the access token as an HTTP-only cookie. The social media module includes features such as posts, stories, and events. Users can share posts with images and descriptions, like and comment on posts, view personalized feeds, and explore profiles. Faculty members can create and post events, while users can view and access event details.

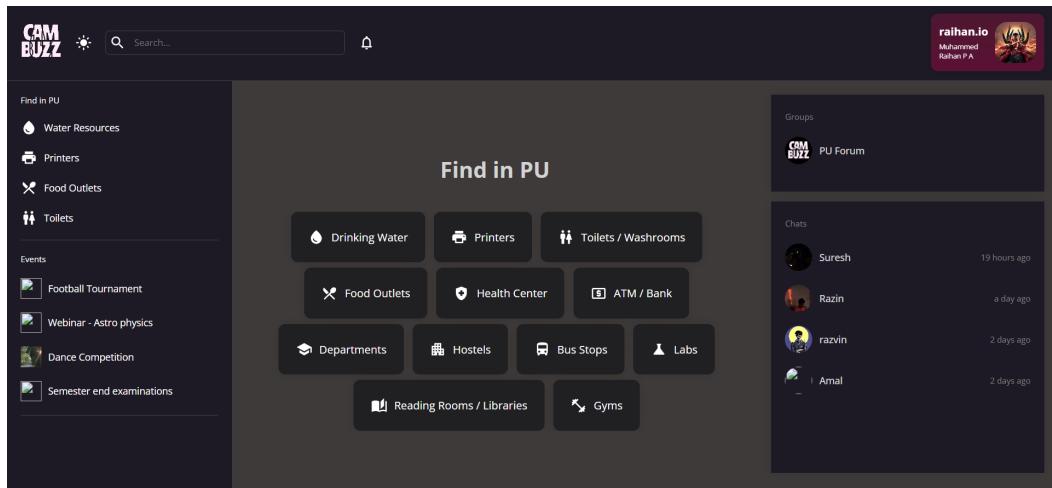


Figure 6.2 Resources (Desktop)

The resources module provides a comprehensive directory of resources available on campus. Users can search for specific resources, view details, sort by distance, and utilize maps for navigation. Bus stops also display bus timings. The chats module enables users to engage in one-on-one conversations and participate in the public PU Forum for open discussions and information sharing. The PU Forum acts as a centralized hub for communication within the Cambuzz community. The admin portal offers modules for faculty management, class management, and resource management. Admins can add, edit, and delete faculty profiles, classes, and resources, ensuring efficient management of the system.

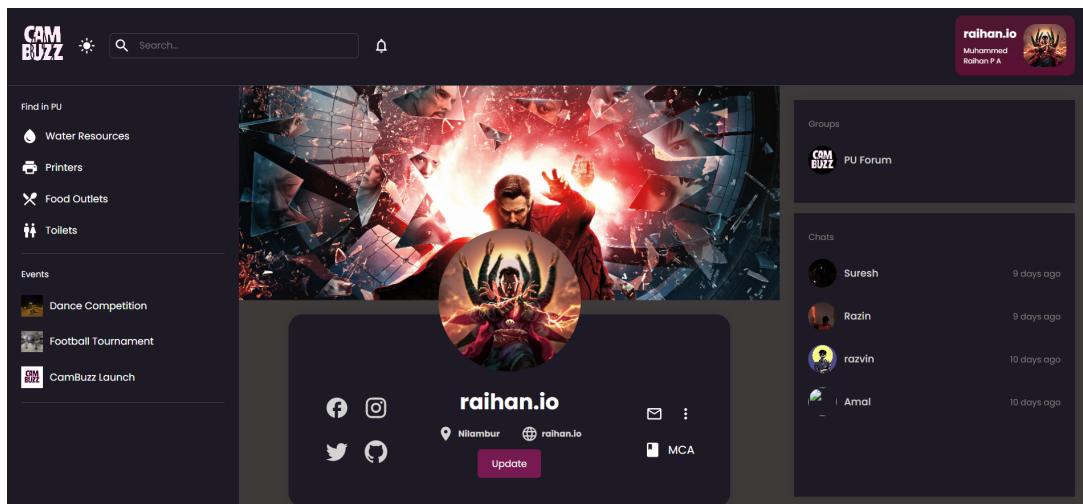


Figure 6.3 Profile (Desktop)

Overall, the implementation of the Cambuzz system involves the integration of various technologies, libraries, and APIs to deliver a comprehensive and engaging platform for users to connect, share, access resources, and communicate effectively.

6.3 TESTING

Testing is an essential process in software development aimed at identifying and rectifying errors to ensure the validity and reliability of the system. The Cambuzz system also underwent rigorous testing to validate its functionality and performance. The testing process consisted of several phases, including unit testing, validation testing, functional testing, and GUI testing.

6.3.1 Unit Testing

Unit testing focuses on testing individual units or components of the software system to ensure their proper functioning. In the case of Cambuzz, unit testing was conducted on various software components, such as the search algorithm, database queries, and user interface controls. The search algorithm was tested to verify the accuracy and relevance of search results based on given criteria. Database queries were examined to ensure correct retrieval and updating of content in the knowledge base, confirming the expected data and accurate updates. User interface controls, including buttons, text boxes, and menus, were tested to ensure their expected behavior in capturing and processing user input. Additionally, access controls were verified to restrict access and modification of content to privileged users only. Error handling was also assessed to ensure the system gracefully handles errors, displaying appropriate error messages without compromising data integrity. Unit testing validated the proper functioning of individual components within the Cambuzz system.

6.3.2 Validation Testing

Validation testing is an essential part of the software development process, which aims to ensure that the system meets the specified requirements and satisfies user needs. In the context of Cambuzz, validation testing was conducted to verify that the system effectively incorporates specific features, such as sorting resources based on the distance from the current location and restricting post requests to faculty members only. To validate the sorting of resources based on distance, various scenarios were tested. The system was tested by providing different locations and verifying that the resources displayed were indeed sorted based on their proximity to the current location. This validation ensures that users can easily find relevant resources based on their geographical proximity.

Furthermore, to validate the restriction of post requests to faculty members, specific test cases were designed. The system was tested by attempting to submit post requests using different user roles, such as faculty, students, and administrators. It was confirmed that only faculty members had the privilege to submit post requests, while other user roles were appropriately restricted from accessing this functionality. This validation ensures that the system maintains the necessary access controls and only authorized users can perform certain actions.

By conducting validation testing on these specific features, Cambuzz ensures that the system meets the requirements of its users and provides the desired functionality. This process helps in enhancing user satisfaction and ensuring the system operates as intended.

6.3.3 Functional Testing

Functional testing encompassed the examination of user registration and authentication processes in Cambuzz. Test cases were developed to validate the workflow for user registration, ensuring that individuals could successfully register and authenticate themselves within the system. This involved verifying that all necessary registration fields were accurately captured, user credentials were securely stored, and the authentication process allowed registered users to access the system without any issues.

Furthermore, the navigation flow and page integration of the system underwent rigorous testing to ensure a smooth and seamless user experience. Test cases were designed to verify the functionality of links and buttons throughout the application, ensuring they directed users to the intended pages and sections. The tests also focused on confirming that users could seamlessly navigate between different pages, such as accessing their profile, exploring resources, and engaging with various features, without encountering any errors or broken links.

By conducting functional testing on user registration and authentication as well as navigation and page integration, Cambuzz aimed to deliver a reliable and user-friendly experience to its users, ensuring that they could effortlessly register, log in, and navigate through the system's interface.

6.3.3 GUI Testing

GUI testing is a form of software testing that specifically examines the graphical user interface (GUI) of an application. In the case of Cambuzz, GUI testing was performed to ensure that the user interface met the desired standards of being user-friendly, intuitive, and aligned with the system requirements.

A notable feature that was specifically tested during GUI testing was the implementation of the dark and light themes using the Context API. This involved evaluating how well the application handled the dynamic switching between these themes. It was important to ensure that the transition between the dark and light themes was smooth, seamless, and visually appealing, providing users with an enhanced experience based on their preferences.

Another important aspect was testing the data input fields. It was crucial to ensure that these fields were easy to use and that the input validation mechanisms were functioning properly. This involved verifying that users could enter data without any difficulties and that any mandatory fields were appropriately marked and enforced. Furthermore, GUI testing examined the page layout to ensure consistency, visual appeal, and compliance with system requirements. Font size, color scheme, and overall layout were evaluated to ensure they were visually pleasing and provided a comfortable user experience. Additionally, GUI testing verified that Cambuzz adhered to any branding or design guidelines in place.

Finally, GUI testing included assessing the system's error handling capabilities. It was essential to verify that appropriate error messages were displayed when users entered incorrect information or encountered system errors. The error messages needed to be clear, concise, and provide actionable feedback to guide users in resolving any issues they encountered. Through rigorous GUI testing, Cambuzz ensured that its knowledge hub system offered an intuitive and user-friendly interface. This included implementing the dynamic switching between dark and light themes using the Context API, providing users with a visually appealing experience that aligned with their preferences. The focus on GUI testing ultimately led to improved user adoption, satisfaction, and an overall enhanced user experience.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSION

In conclusion, Cambuzz is an advanced social media website designed for the Pondicherry University community to improve communication, collaboration, and resource discovery. The platform offers features such as real-time updates, a thorough campus resource locator, and a user-friendly interface accessible from any device. The feasibility analysis revealed that the project is operationally, technically, and economically feasible. The software requirements specification outlined the necessary components for the development of the platform. The technologies used, including Axios, Mutation, Google Map API, Firebase Authentication, and JWT, are essential to the development of the platform's user interface, authentication, and resource location functionality. The system analysis highlighted the essential classes responsible for the user interface components, user attributes, and post/resource attributes. Overall, Cambuzz is a promising solution to enhance the university experience by promoting a centralised platform for students, faculty, and other campus-related individuals to connect and collaborate effectively.

7.2 FUTURE ENHANCEMENTS

1. Student Management System: This system would allow faculty to manage student attendance, grades, and other academic-related information.
2. Virtual Classroom: A virtual classroom feature could be added to Cambuzz to allow for online classes, webinars, and other virtual meetings.
3. Online Library: Cambuzz could integrate with the university's library system to allow students and faculty to access online resources, such as e-books, academic journals, and research papers.
4. Career Services: A feature could be added to Cambuzz to connect students with career services, such as campus placements, job postings, internships, and career counselling.
5. MarketPlace : This feature can be used to put up Vehicles, Books or Utility items for sale or rent.

REFERENCES

1. <https://developers.google.com/maps/documentation/javascript>
2. <https://axios-http.com/docs/intro>
3. <https://tanstack.com/query/latest/docs/react/overview>
4. <https://www.permify.co/post/jwt-authentication-in-react>
5. <https://www.npmjs.com/package/react-jwt>
6. https://developer.mozilla.org/en-US/docs/Web/API/URL/createObjectURL_static
7. <https://react-google-maps-api-docs.netlify.app>
8. <https://firebase.blog/posts/2017/02/email-verification-in-firebase-auth>

APPENDICES

APPENDIX A: SAMPLE SCREEN SHOTS

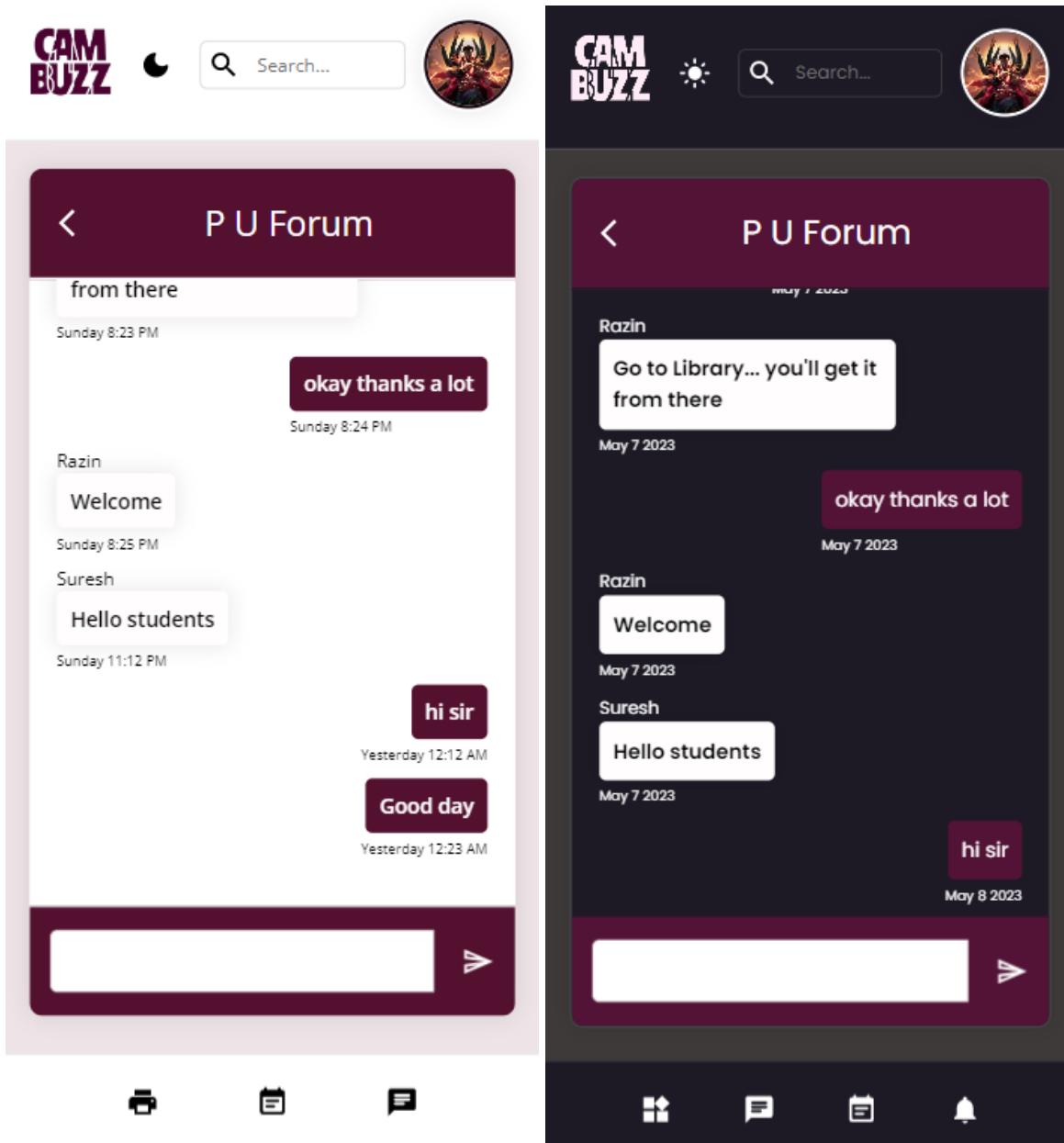


Figure 8.1 Chats - PU forum(Mobile)

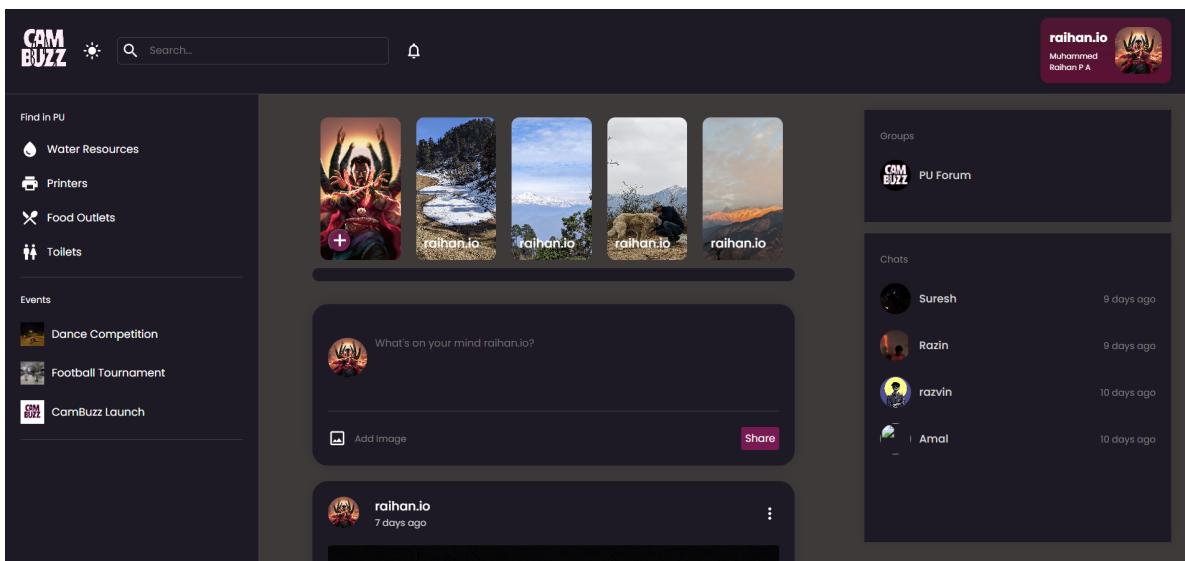
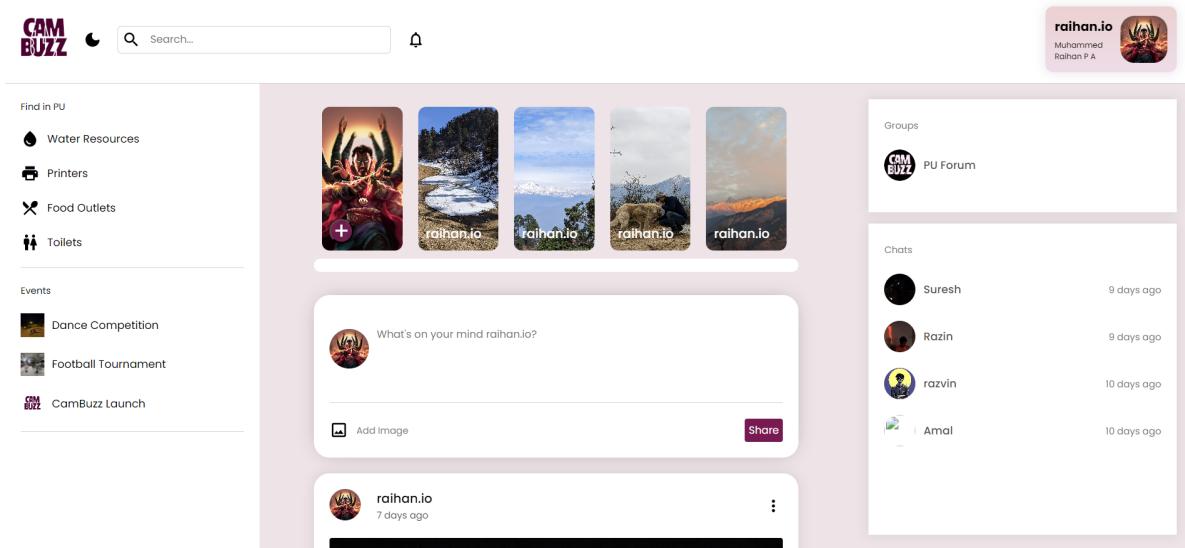


Figure 8.2 Home (Desktop)

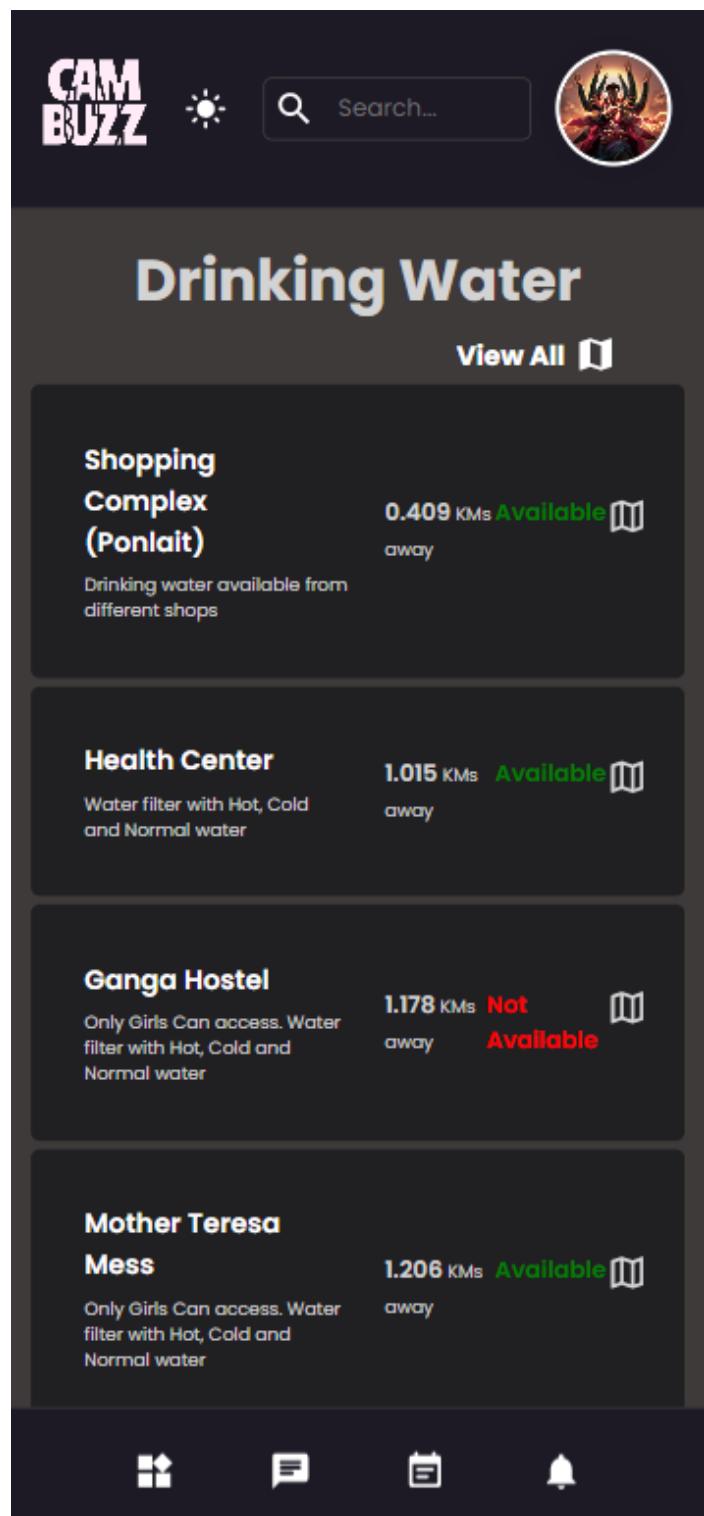


Figure 8.3 Resources(Mobile)

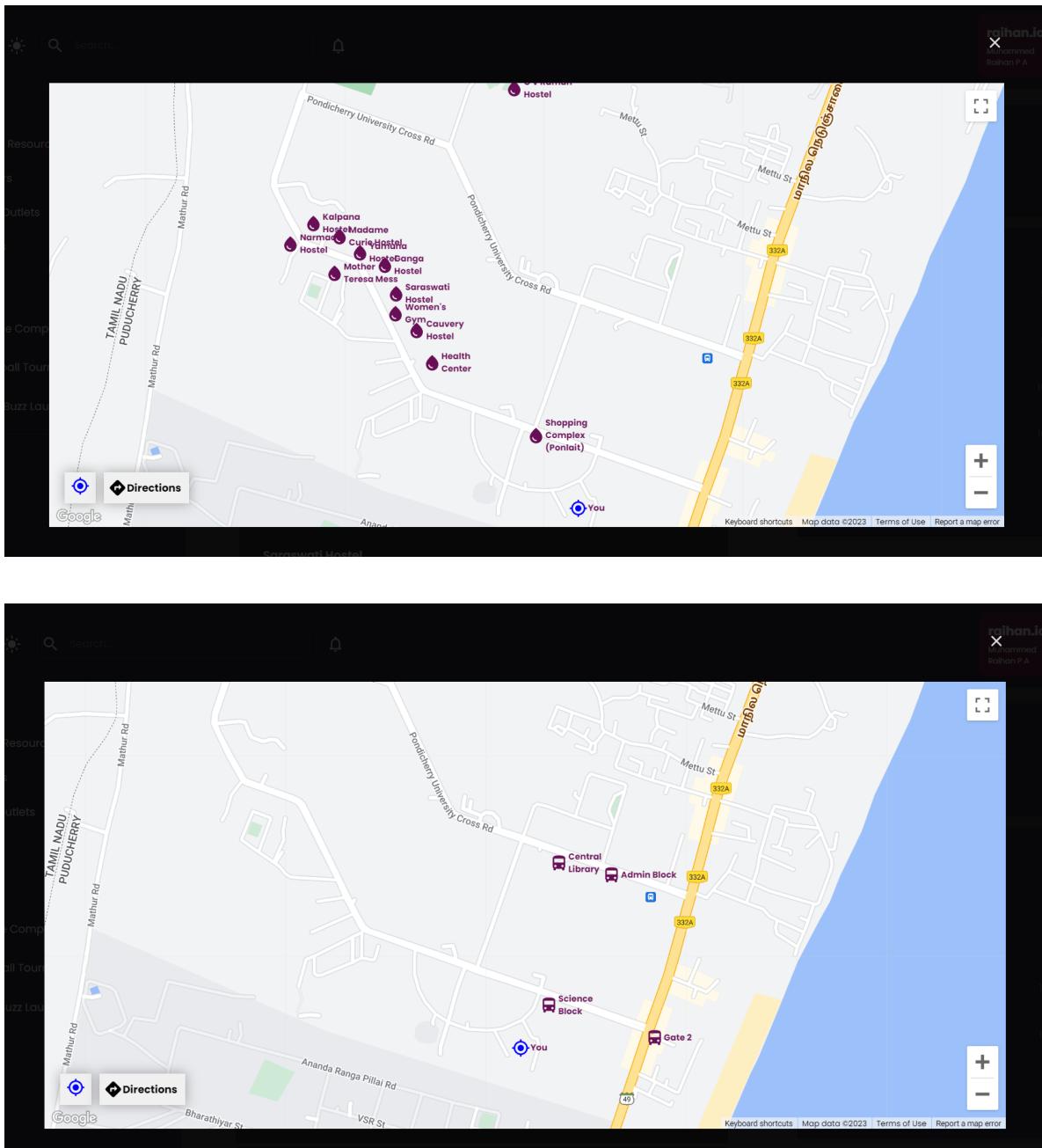


Figure 8.4 Resource Map- Water, Bus Stops(Desktop)

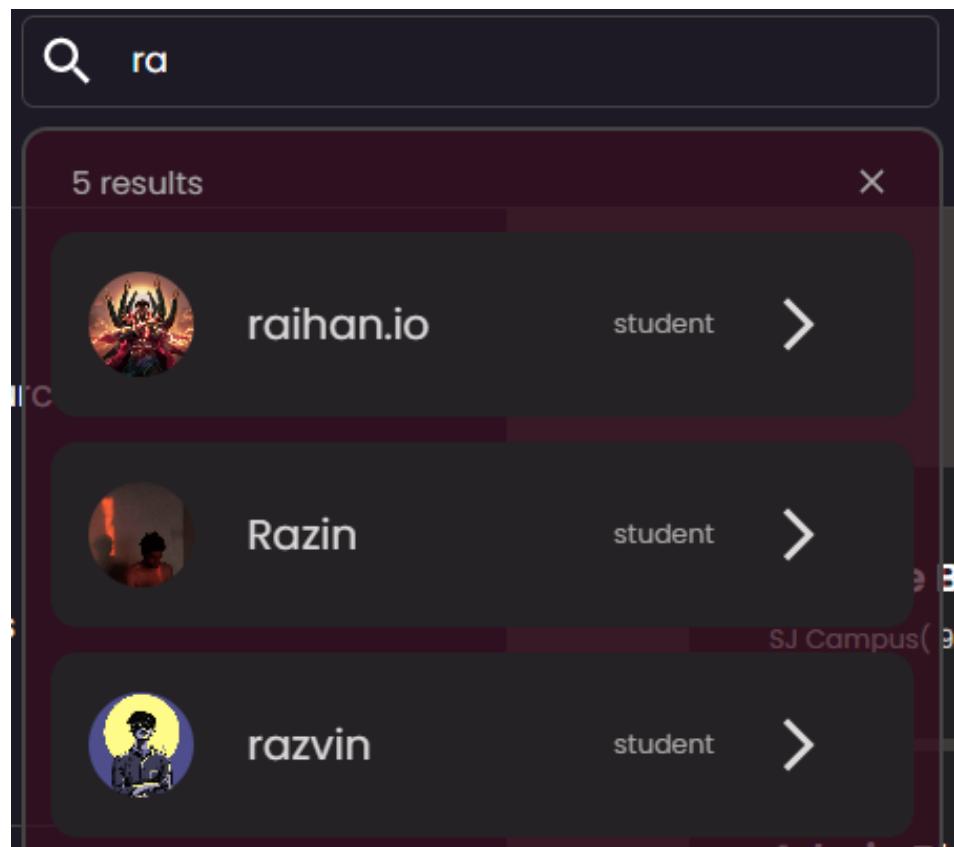


Figure 8.5 Search (Desktop)

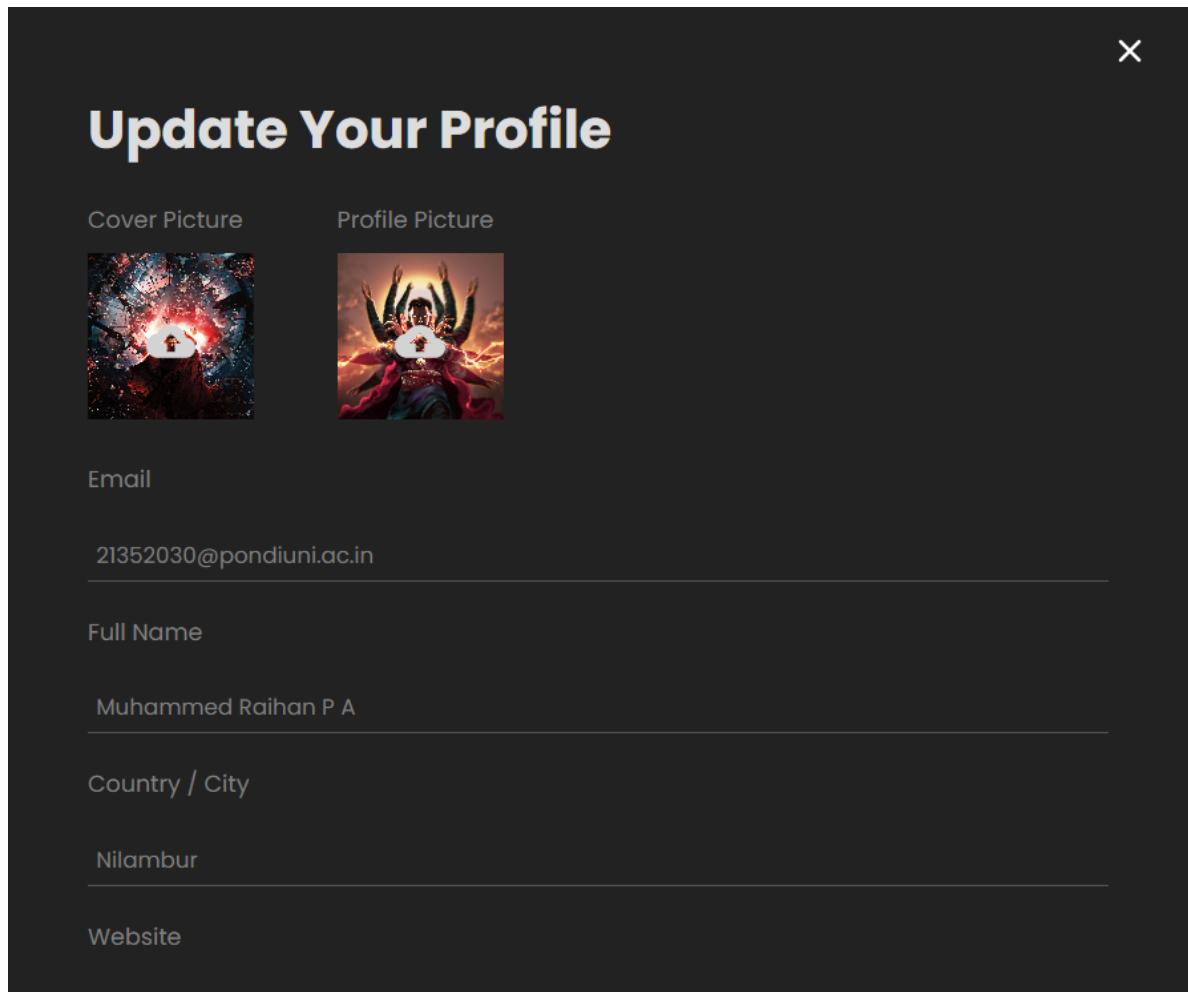


Figure 8.6 Update Profile(Desktop)