Volume 11, Issue 3, March – 2026 International Journal of Innovative Science and Research Technology
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

CampusConnect: Intelligent Campus Assistant

K. Durga Saranya[1]; P. Himanth Naga Charan[2]; G. Sri Deepti[3]; K. Mounika[4]; A. Gopal[5]; A. Ayyappa[6]; K. Chaitanya[7]

[1](M. Tech, Assistant Professor)

[1;2;3;4;5;6;7]Department of Artificial Intelligence and Machine Learning Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh India

Publication Date: 2026/03/20

Abstract: University students these days use a myriad of fragmented platforms to access information and services on campus such as emails, notice boards, social media groups and separate web portals. The consequences of this dispersed communication environment are often missed updates, slow responses, and an increased amount of administrative work. This paper describes CampusConnect as a centralized real-time web platform developed to integrate the key services on campus into one that is easily accessible to address the issues. The features offered by the platform can be listed as campus-wide announcements, lost and found reporting, sharing of academic notes, club and event updates, room information, and resource information, and simple student support requests. The CampusConnect is developed based on a modern full-stack architecture, React in the front end and Supabase in the role of a backend-as-a-service, which offers file storage, secure authentication, and real-time database updates. In addition, the system integrates a third-party language model API to provide document summarization and conversational assistance to enhance user interaction necessitating no specific model training. The platform makes campus information more accessible, opens up communication, and reduces reliance on manual administrative processes as was demonstrated through functional testing and user evaluation. Everything said and done, CampusConnect is a viable solution for making the campus more comfortable to students.

Keywords: *Campus Management, Real-Time Platform, Student Services, AI Integration, Responsive Design.*

How to Cite: K. Durga Saranya; P. Himanth Naga Charan; G. Sri Deepti; K. Mounika; A. Gopal; A. Ayyappa; K. Chaitanya (2026) CampusConnect: Intelligent Campus Assistant. *International Journal of Innovative Science and Research Technology*, 11(3), 1585-1595. https://doi.org/10.38124/ijisrt/26mar945

I. INTRODUCTION

A campus is a busy environment where administrators, instructors, and students interact in academic and extracurricular activities, information sharing, and through services. Though the digital tools have been adopted by campuses to improve communication over the past few years, much of the information flow is dependent upon geographically scattered platforms such as emails, messaging groups, page of social media, and physical notice boards. Such a haphazard style often results in valuable activities happening in campus being hard to follow, announcements overlooked and service replies taking too long.

The search of study materials, tracking of events, reporting lost things, and basic assistance services are all typical issues of students. At the same time, administrators must work hard in order to accommodate repetitive questions and distribute information via different sources. There is an increasing demand of a single system which is efficient and convenient in terms of communication and managing services provided to the community since communities of campuses are expanding and becoming more linked.

To meet this requirement, CampusConnect is a centralized web based interface that hosts prominent campus services on one platform. The system has built-in announcements and sharing of academic notes, loss and found reporting, club and event updates, room and resource, and student support requests. The platform ensures that the information is received by the relevant users at the right time and safely by offering hierarchical access control and dynamic up-to-date information. CampusConnect is a service that not only combines services but also includes a conversational assistant which uses an external language model API to assist users with easy to answer questions and summaries of content. The front end of the platform is developed with React, and the back end is developed with Supabase. This renders the platform responsive, safe and capable of processing data in real time. CampusConnect is primarily aimed at simplifying the process of communicating with others, simplifying the accessibility of services on campus and synchronising and harmonizing the campus.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

II. RELATED WORK

Over the past few years, numerous digital systems have developed to improve automation of institutional services and campus. communication have been developed. In most universities, academic management portals are utilized and handle student record, examination results among other functions. course enrolments. The systems are also likely to have a limited scope in administrative functions only unlike other systems which can be installed in institutions to serve academic functions. and do not involve sharing of resources among peers like campus announcements, lost and found reporting, etc.

It has also web based applications that are provided to support some student services such as event. advertisements, internet bulletin boards or documents sharing sites. Although these tools may enhance the accessibility of information, it is applied as an individual tool most of the time. This compels the students to work on a number of platforms to avail different services, which renders this less convenient, and introduces discontinuity to information.

The recent studies have examined the applications of real time database and cloud based back end services to assist in real time information updates in web applications [1]. The advantages of centralized communication platforms have been mentioned in other studies, which also emphasize the enhancement of engagement in academic institutions [2]. Also, it has suggested conversational interfaces and system that uses chatbots to help students with common questions and information searching [3]. Nevertheless, in the vast majority of the cases, these smart functions are not combined as a full-fledged campus service environment but rather as separate applications.

In contrast to the current solutions, CampusConnect is aimed at integrating various campus service modules into one real-time solution with a secure role-based access control and embedded conversation support. The proposed system would overcome the shortcomings of the disjointed campus applications and provide a more tightly integrated and effective solution to the modern educational facilities because of the integration of services and communication tools.

III. PROBLEM STATEMENT AND SYSTEM OBJECTIVES

➢ *Problem Statement*

Universities communicate with one another in a considerable number of ways such as notice boards, emails, messaging groups, and separate web portals to share information and conduct student services. This haphazard practice usually results in tardiness of announcement, lack of updates and difficulty accessing valuable campus materials. It is difficult to locate studying materials, to report about lost items or to request simple help services on a single and reliable platform. Meanwhile, administrators find it difficult to share the information promptly and respond to the same questions on other media to the students. The lack of the single campus service system in real- time makes the communication more difficult, and the number of tasks to be performed manually is increased, which makes the services of the institution inefficient in general.

➢ *Proposed System and Objectives*

CampusConnect is a web based centralised platform which seeks to centralize all the vital campus services on a single digital platform. The system has substituted the antiquated communication methods that used to be fragmented such as notice boards, emails and informal messaging groups with an interface that is easily accessible to students and the administration in chatting with each other. CampusConnect provides you with real-time access to campus news, campus academic resources, service requests, and support features. This facilitates and hastens the dissemination of information.

The platform allows students to view significant news, post and download study resources, report things they have lost or found, request information regarding rooms or resources, and request the basic help or medication. Administrators also have their access to make announcements, check requests of the students, update information in the campus and monitor the activities of the platform. The proposed system will also streamline the functions of the campus, minimize the number of people handling the manual workloads, as well as enhance access to institutional services by integrating several services into one platform.

- *The Primary Objectives of this Project are:*

✓ To develop a single interface to handle different services on campuses.
✓ To provide real time campus-wide announcements and updates to reduce manual workload.
✓ To make the study materials and academic notes easily share and accessible to students.
✓ To offer lost and found reporting services in the campus online.
✓ To utilize a structured system and direct and easy support and medication requests via the Internet to give room and resource information.
✓ To implement the application of the secure login and role- based access control between the students and the administrators.
✓ To improve communication between the students and the administrators by keeping them informed.
✓ To include a chat robot which will handle simple queries by the end-users.

IV. SYSTEM ARCHITECTURE AND DESIGN

The CampusConnect is designed on a service-oriented architecture that is modular, which enables students to interact with others in real time, provides them access to all their services under one roof, ensures their data is safe, and receive assistance of other users. The system consists of several functional components that collaborate to make one digital space between students and administrators.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

➢ *Student User*

This role is assigned to all students that make use of the platform. The students can read on-campus announcements, create and share notes, request information on lost and found, request information on rooms or resources, request assistance or medication, and chat with the chatbot assistant.

➢ *Administrative User*

The administrative users are tasked with the responsibility of managing and controlling campuses in services and information. There are three sub-roles in this category which are defined by access rights:

- *HOD*

The HOD is responsible for managing department-level academic information and student-related services. This includes posting departmental announcements, sharing academic resources, overseeing student requests, and ensuring smooth communication within the department.

- *Club Administrator*

The club administrator is in charge of controlling the club activities, making announcements and tracking the number of students attending. The work is all about handling of information on student clubs and extracurricular activities.

- *Super Administrator*

All the operations of the platform are under the supervision of the Super Administrator, who posts announcements to the entire campus, approves or deletes content, controls user roles, and oversees the activity of the systems to ensure that the data is secure and correct.

➢ *System Architecture*

The CampusConnect system architecture is created to be a layered web architecture which associates users, application services, backend processing and data storage in a systematic way. The system is accessed by students, HODs, Club Administrators and Super Administrators using secure authentication provided by the Clerk service which performs sign up, log in and user sessions. After authentication, users navigate the platform by means of the frontend interface based on React that offers all the features of campus services.

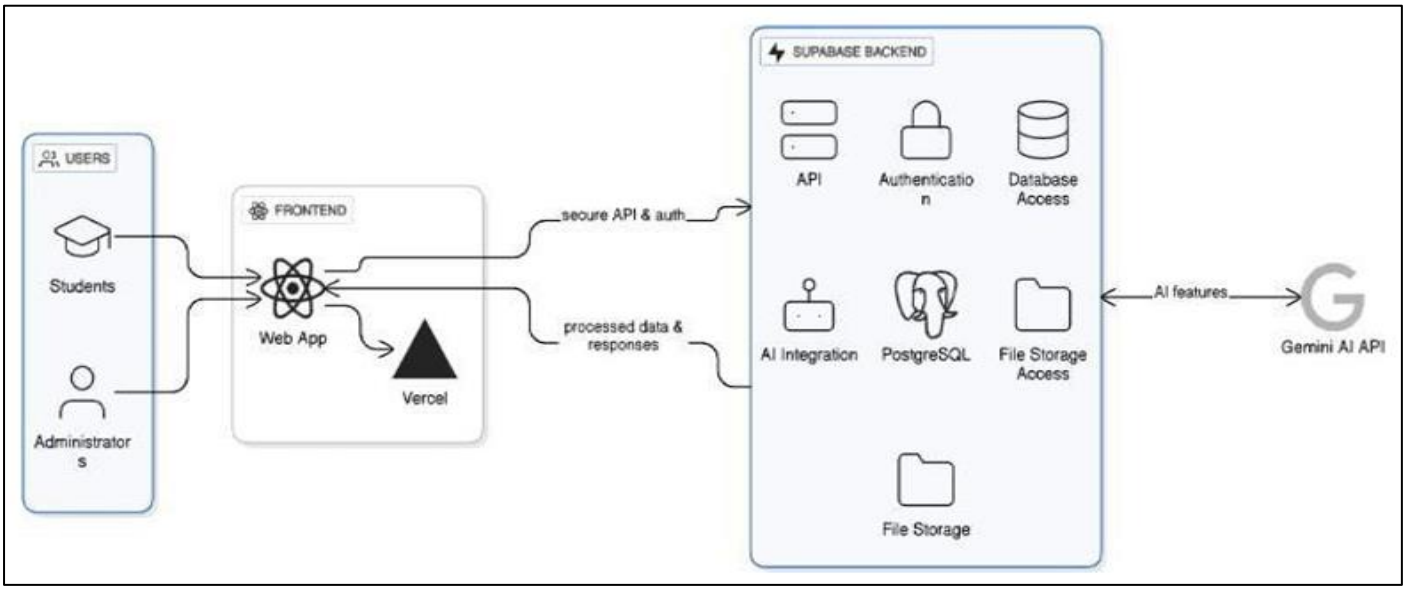

Fig 1 System Architecture

The frontend communicates with Supabase backend services using edge functions. Such services will take care of such issues as real-time data synchronization, authentication verification as well as executing operations on the server side. All structured information such as user information, announcements, service requests, and reports is stored in A PostgreSQL database with Row Level Security (RLS) to ensure that only authorized users can access it.

The file storage of Supabase deals with files uploaded like notes and pictures. Moreover, there are AI integrations like the student chatbot and the feedback summarizer that allow interacting with the users by offering rapid support and summarizing content.

➢ *UML Design*

CampusConnect is designed with the UML diagram to display the relationships among users and modules, how they interact in the system. The process of work of the student and administrator, system component organization and service communication are all explained. These diagrams provide a graphical depiction of the way the system should be constructed and the way the operations should go.

- *Class Diagram*

CampusConnect's static design is shown in the class diagram, which includes important classes like Student, Admin, Announcement, and Services. It explains each class's characteristics and functions and shows how different system components relate to one another, including dependencies and associations.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

- *Use Case Diagram*

Use case drawings for the following user roles are available in the CampusConnect system: Super Admin, Club Coordinator, Head of Department (HOD), and Student. Each diagram displays the features that a user can access once they have logged in. The student use case diagram shows actions like placing, receiving services, and reading announcements.

The operations of academic management, student communication, and approvals form the basis of the HOD use case model. Event management, announcement creation, and member coordination are all included in the Club use case diagram. The Super Admin's use case diagram shows system- level control as well as system-level tasks like role assignment and user management. The system responsibilities and access privileges are clearly described by these role-based use case diagrams. The admins use cases diagrams are provided below.

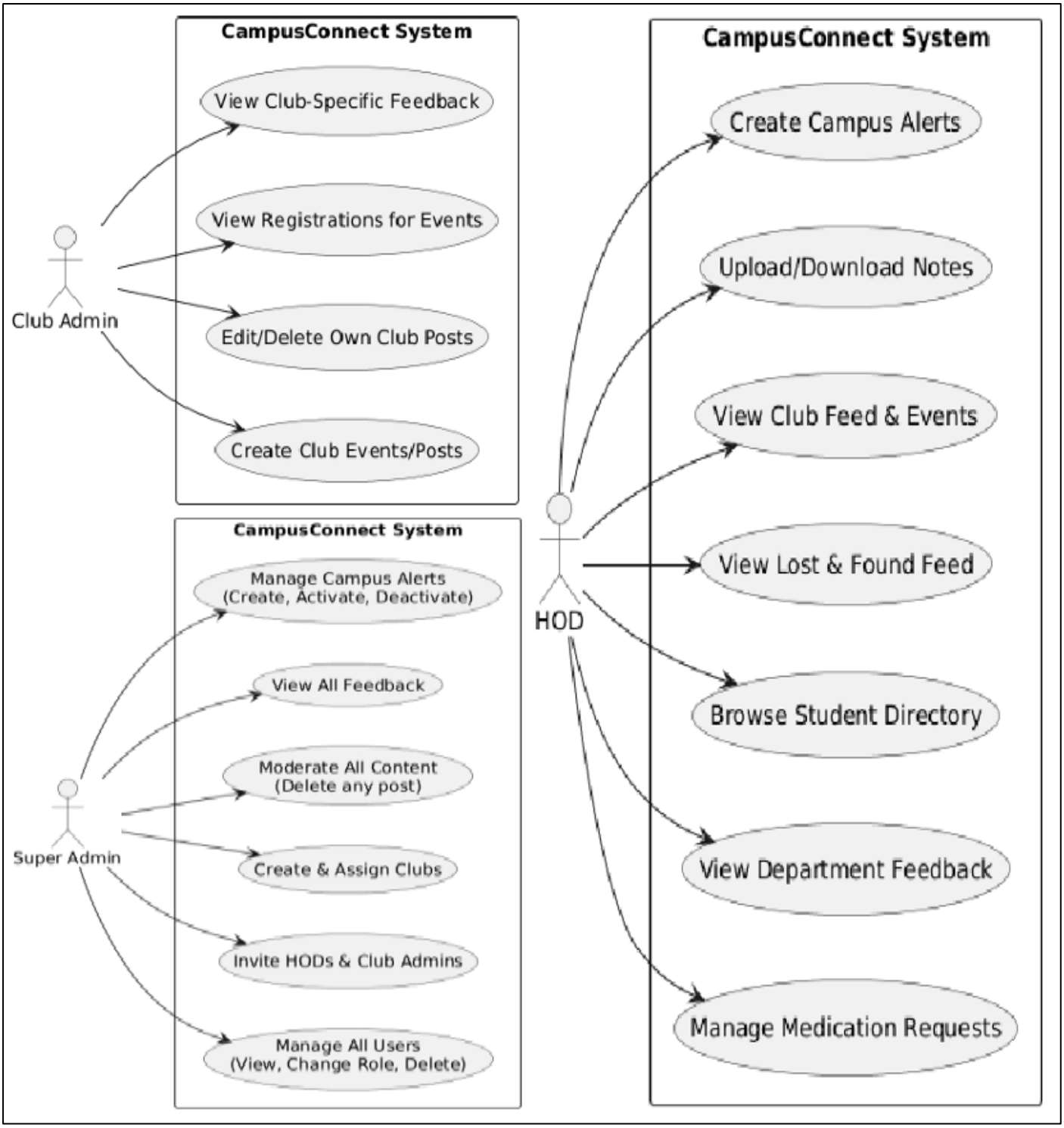

Fig 2 Use Case Diagrams

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

- *Sequence Diagram*

The sequence diagram of CampusConnect shows the sequential interaction of the students, administrators, and system modules. It indicates how requests (e.g. watching an announcement or posting or asking questions) get processed sequentially by different aspects.

The diagram will help in identifying important operationalities, message flow and the way operations will occur sequentially.

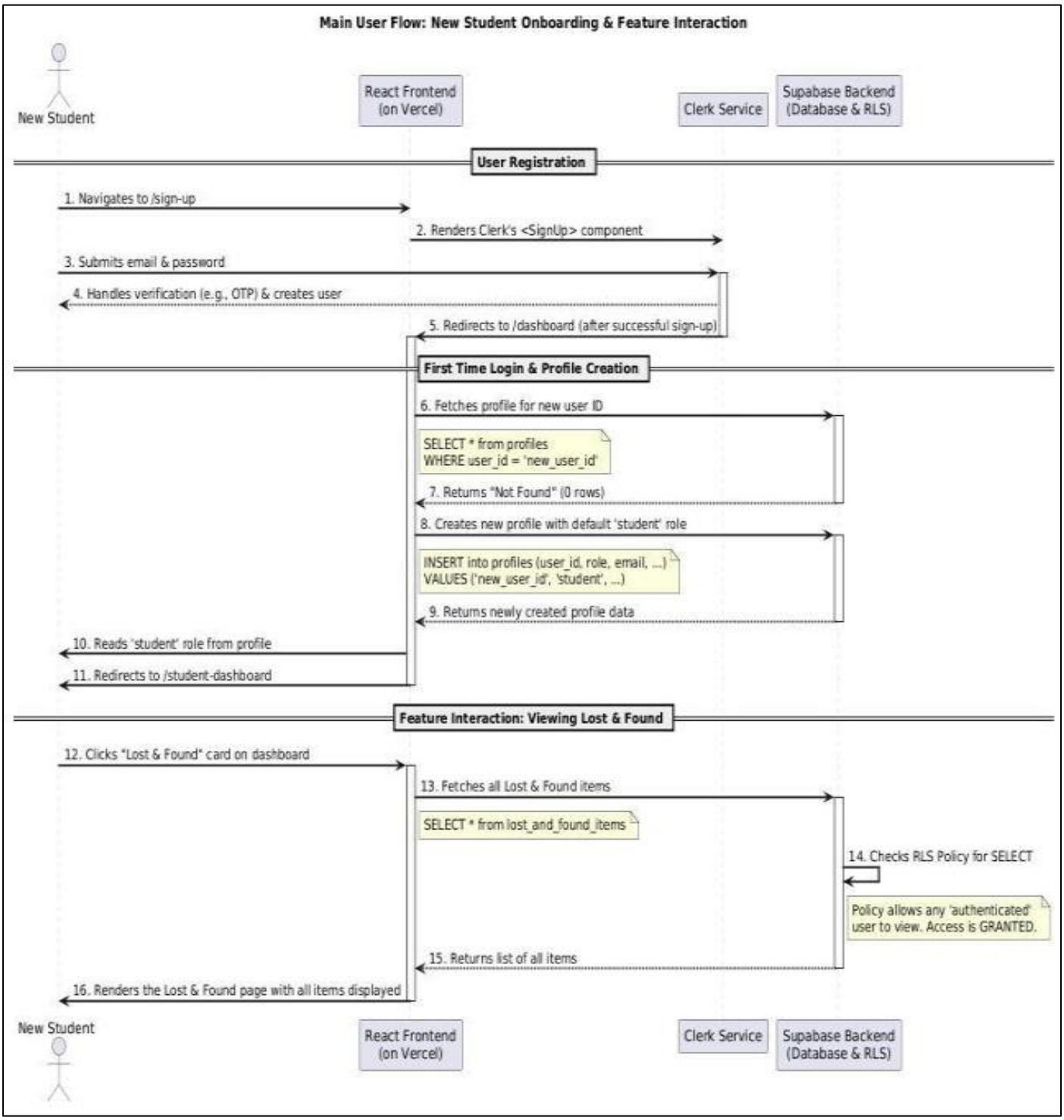

Fig 3 Sequence Diagrams

V. METHODOLOGY AND IMPLEMENTATION

CampusConnect system is built on the concept of a modular, stratification and role based model to enable various functions of the users such as Student, Club, HOD, Coordinator and Super Admin. The system shall be designed to have a layered architecture which will consist of presentation layer, application layer, and data layer which will offer isolation of concern and easy maintenance. Authentication and authorization of users will be the

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

beginning of the workflow, and the access privileges will be determined with regard to the positions of the users.

The strategy is to architecturally design in a modular fashion with the key functionalities of announcements, academic management, clubs, and notifications and administration control being designed as individual modules. This starts with user validations and role checking, modules validation, request processing and sending a reply. This enhances its scale and security and the system to communicate effectively between the system components.

With CampusConnect, a number of technologies are employed in the front-end, back-end and database layers to deploy a campus communication system, which is responsive and secure. The implementation of the system will be based on the client-server model where the frontend will be communicating with the backend services with the help of RESTful APIs. The contributions of each component in technology stack are distinct to the functionality, performance and scalability of the system.

➢ *Technology Stack*

CampusConnect is a new web-based technology stack, which can be implemented to realize the deployment of the system and obtain the scalability, security, and interactive interaction.

➢ *React with Vite*

The CampusConnect has been developed using React.js as the interface. It allows developing reusable pieces of login, dashboards, announcements and role-specific pieces. The component nature of React has enhanced the consistency of the UI besides offering the capability of dynamical representation of the data that has ensured that it has rendered the experience responsive and user-friendly among user roles.

➢ *Tailwind CSS*

The application is styled using tailwind CSS. It offers utility- first design, this guarantees a similar design, responsiveness, and contemporary user interface across the devices. The Tailwind can also be used to reduce CSS by removing unused styles during the time of assembling the production.

➢ *Supabase (Backend-as-a-Service)*

CampusConnect uses Supabase as the foundation of the entire backend infrastructure. It offers a PostgreSQL database, which is managed to receive all the application information including user profile, role, events, and other messages, alerts, and uploaded resources. The implementation of system security as far as the level of access control of databases is concerned is also centered around Supabase. The Row-Level Security (RLS) policies also allow the users to access the data that they are allowed to look at depending on their roles. In addition to that, Supabase storage and edge functions have an opportunity to control files and conduct operations safely on server as Supabase has a reliable and scaled back-end functionality.

- PostgreSQL Database: Leveraged and centralized information storage with imposed integrity constraints.
- Row-Level Security (RLS): DB level access control to grant role-based permissions.
- Realtime Subscriptions: This would allow real time updates on features like chat and system alerts.
- Secure Storage: Stores files that have been uploaded by the user with access-controlled storage policies.
- Scalability: It relies on PostgreSQL and cloud computing in order to support the growth of the system.

➢ *Clerk Authentication*

The identity management and authentication system that has been used in the CampusConnect system is Clerk. It manages the complete user authentication process, and these are user registration, user login, user session control and secure access control. The authentication interfaces are ready-made and support different ways of signing in; these ensure that all users go through an easy and safe sign in process. Clerk generates role information and session tokens which are used to enforce role-based access in the application. To authenticate with Supabase, Clerk also does the JWT-based authentication model, and every authenticated request includes a temporary token, which Supabase must validate. Such an integration offers a safe communication between frontend and the backend, and it avoids friendly access to the secured resources as well.

- User Authentication: Responsible of managing safe sign-up, sign-in and sessions.
- JWT-Based Access: It is a dynamic one-time generated JSON Web Token which is employed to verify the backend request.
- Role-Based Access Control: The policy enables identification of roles to enable authorised users gain access.
- Secure Integration with Supabase: Supabase uses JWT to verify database and API access.

➢ *Gemini API*

It offers the Gemini API to offer AI-based solutions including chatbot support and content summary. The Gemini API allows the use of Supabase Edge Functions to perform secure request processing in order to avoid sensitive data exposure. AI scales keys and provides interactions of AI.

VI. RESULTS AND DISCUSSION

CampusConnect was implemented and deployed as a fully operational web application. The four user roles, including Student, HOD, Club Admin and Super Admin, worked well with the appropriate role-based access control and authentication in all the modules.

Real-time features such as chats and notifications were working well, and the chatbot assistance was provided by the AI, and content summarization generated the correct and context-dependent answers. The effectiveness of the proposed architecture and implementation was proved by the system stability testing.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

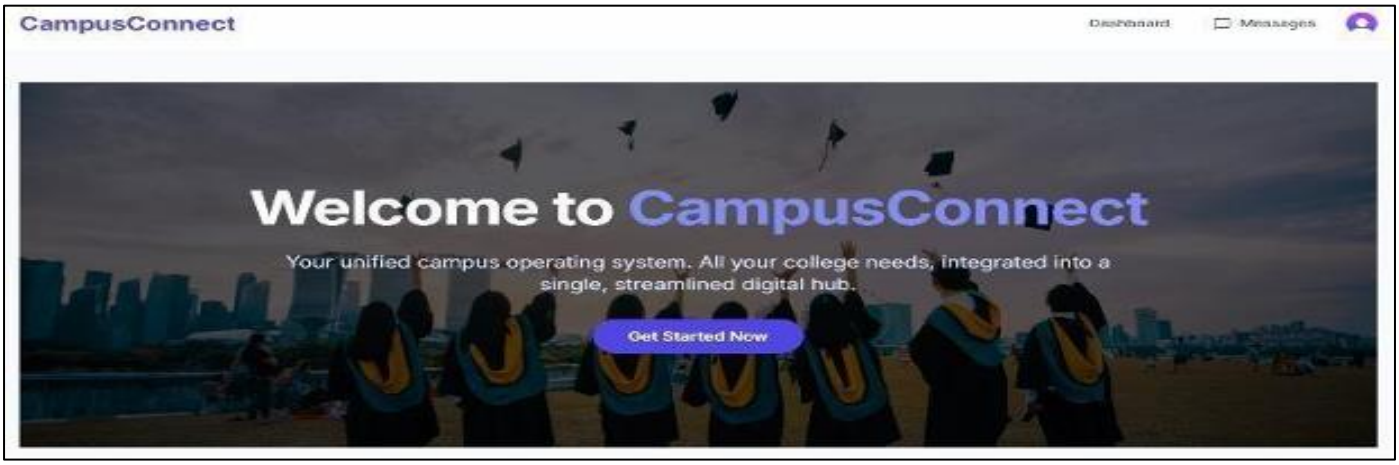

Fig 4 Home Page

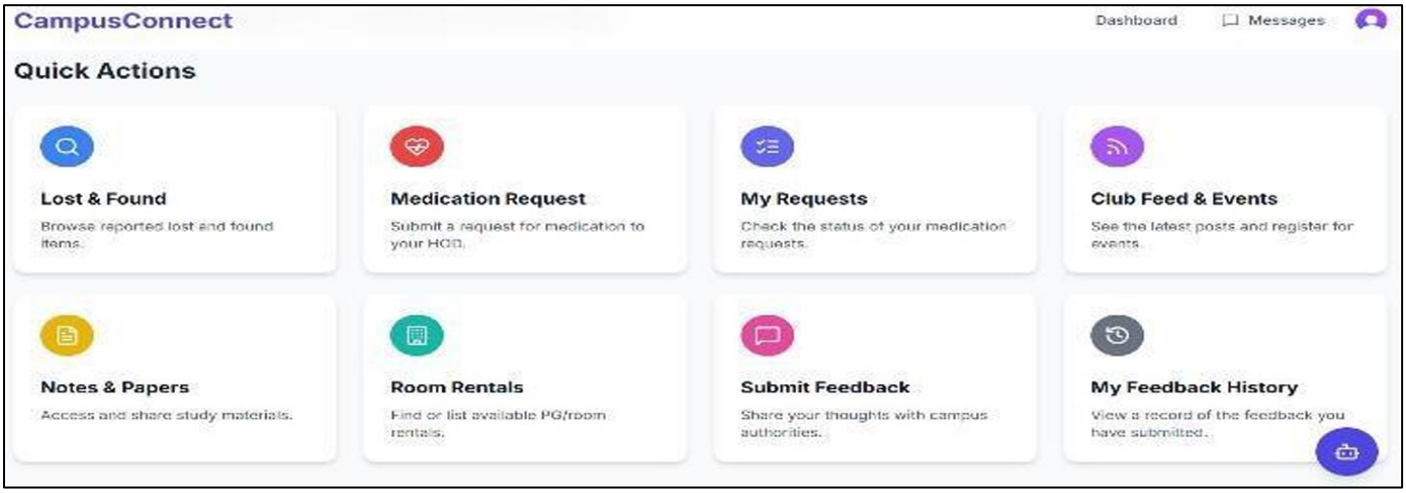

Fig 5 Student Dashboard

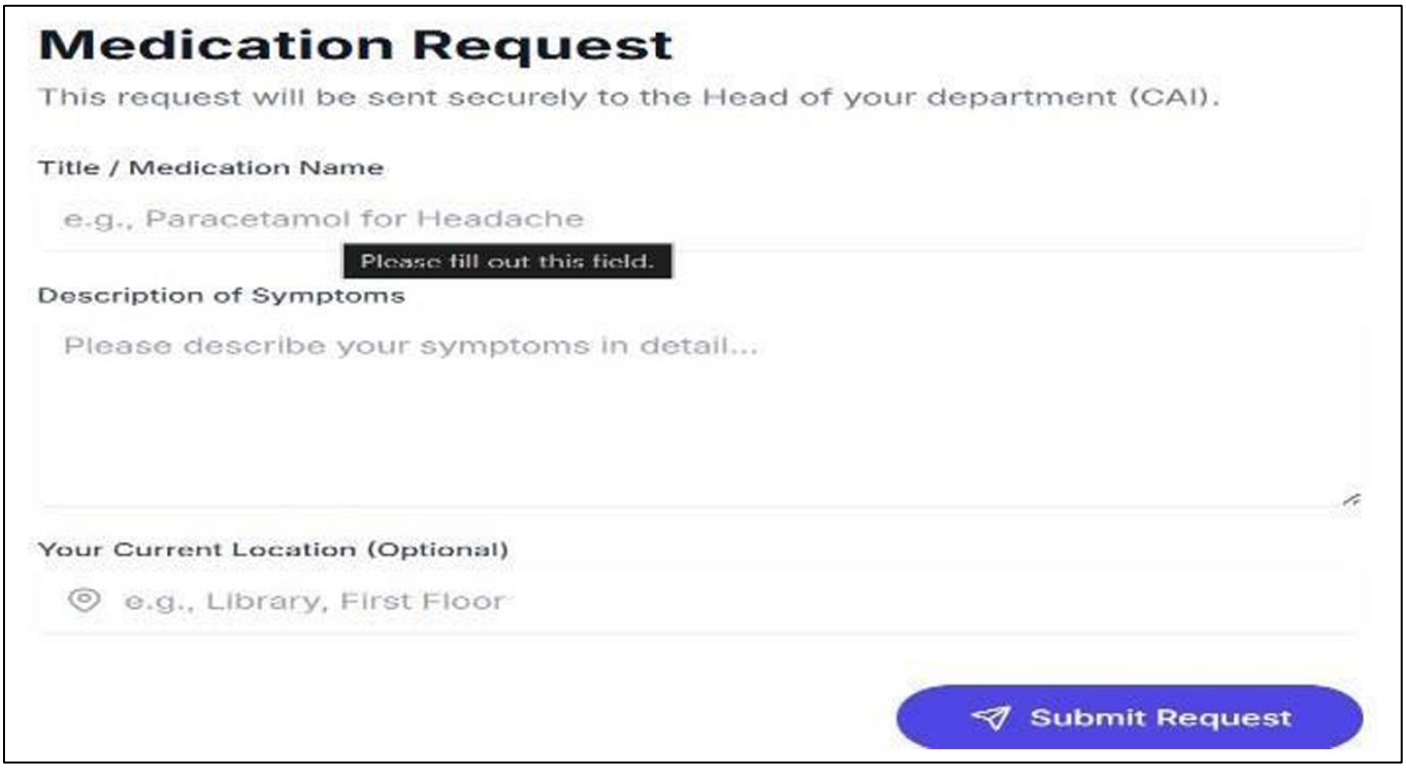

Fig 6 Medication Requests

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

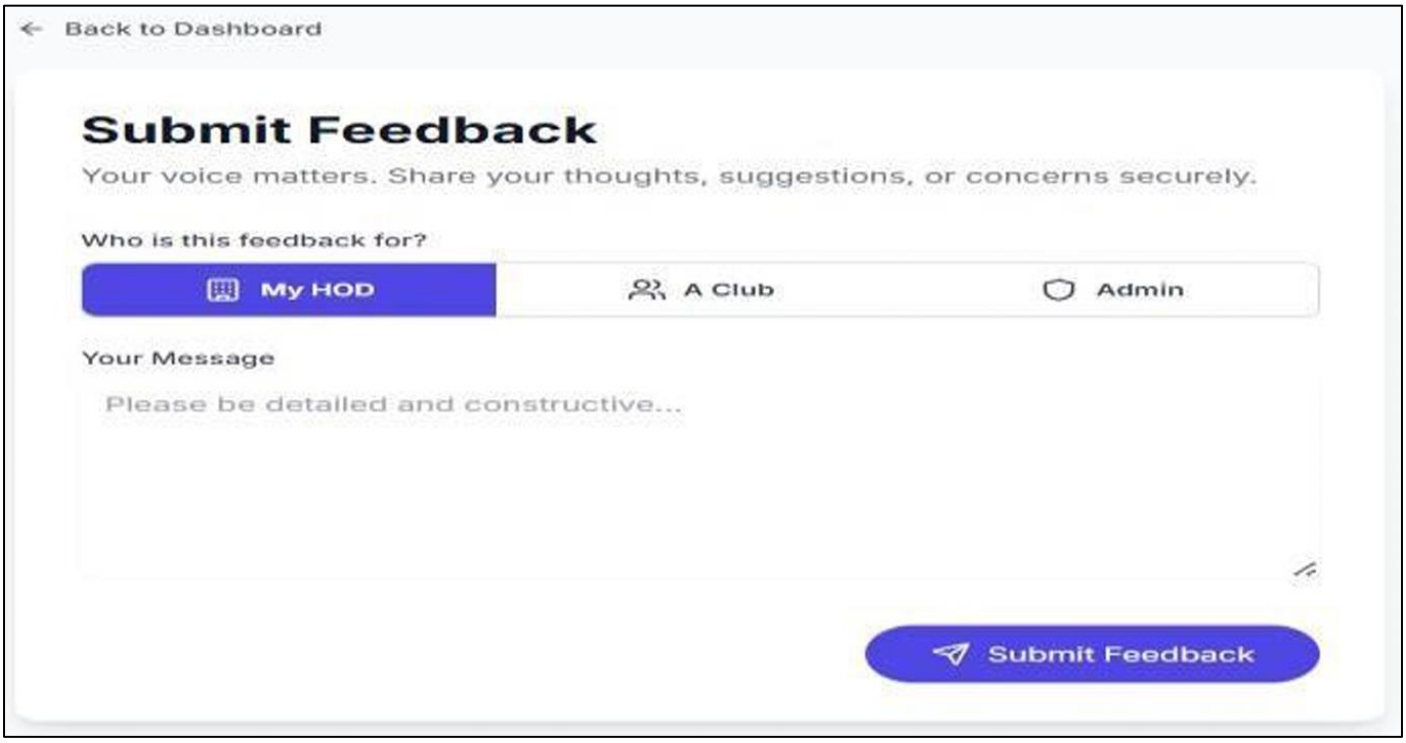

Fig 7 Student Feedback

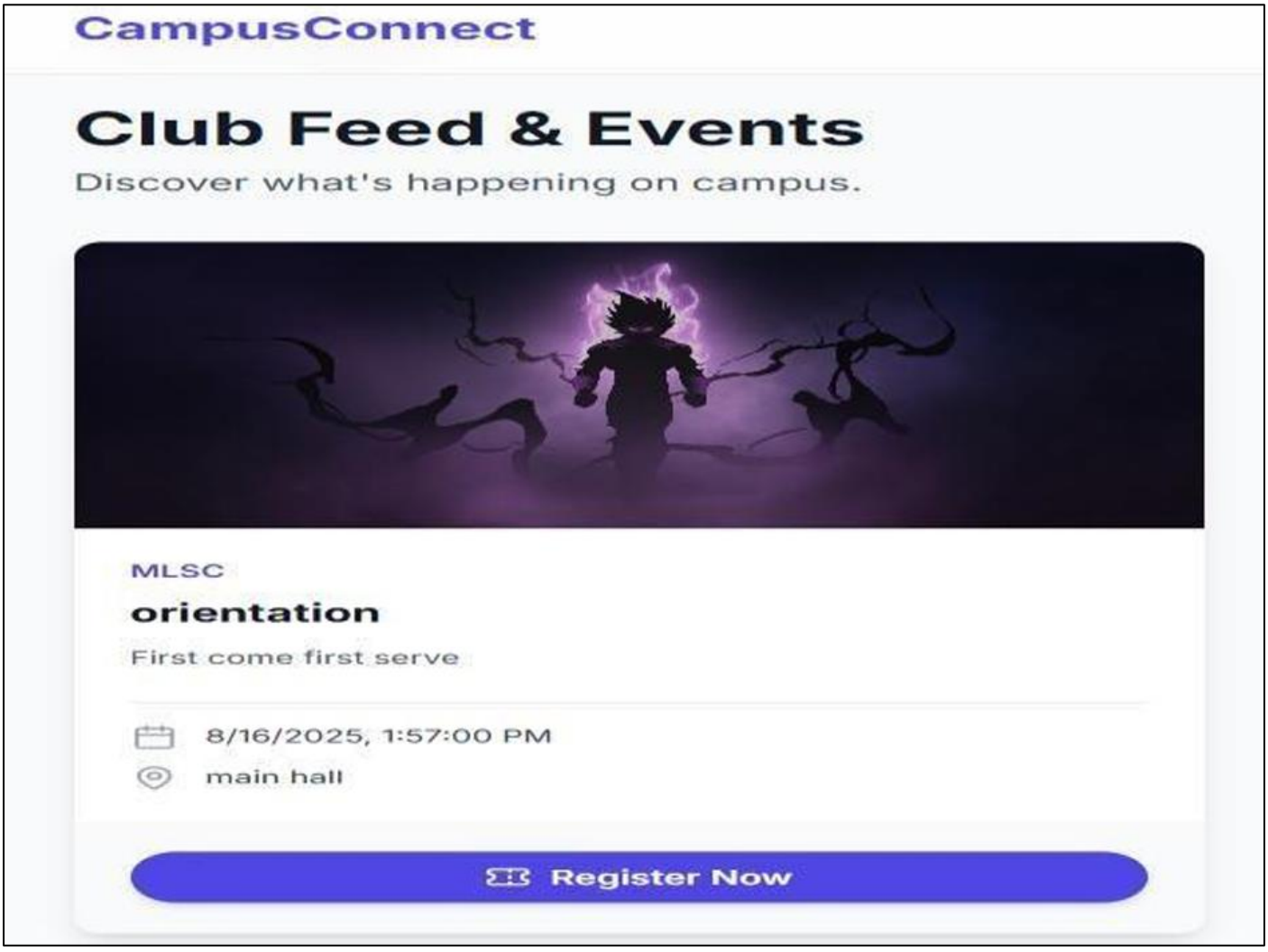

Fig 8 Club Registrations

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

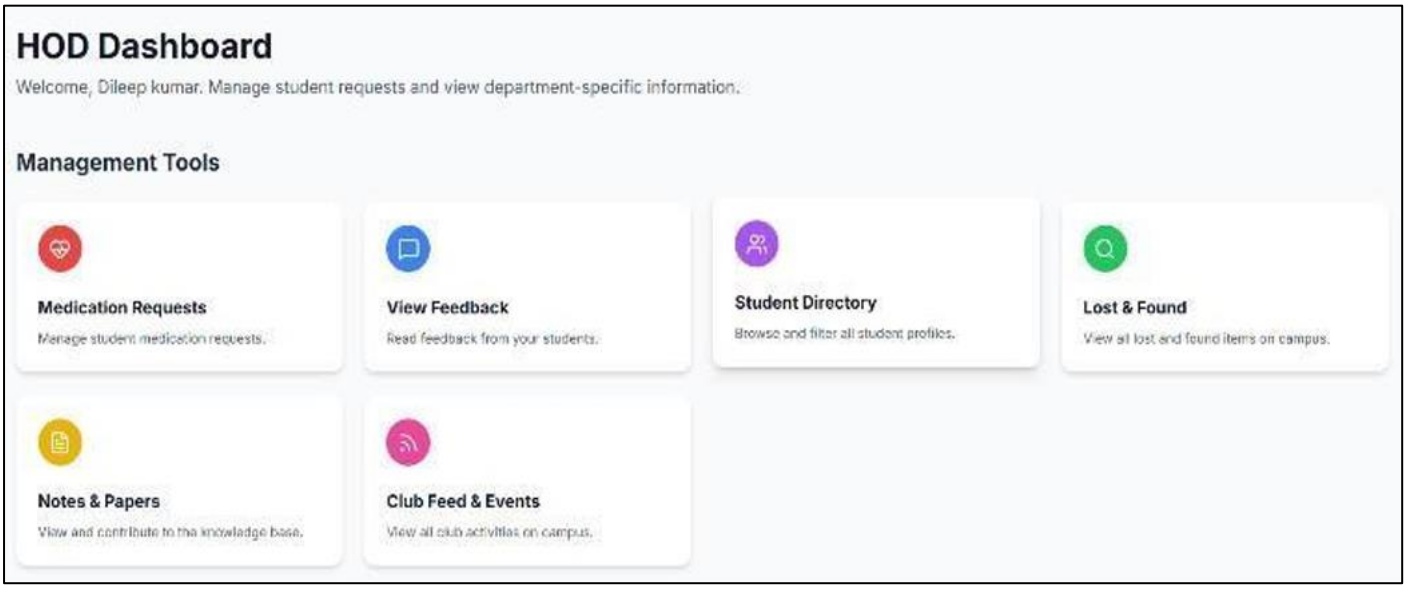

Fig 9 HOD Dashboard

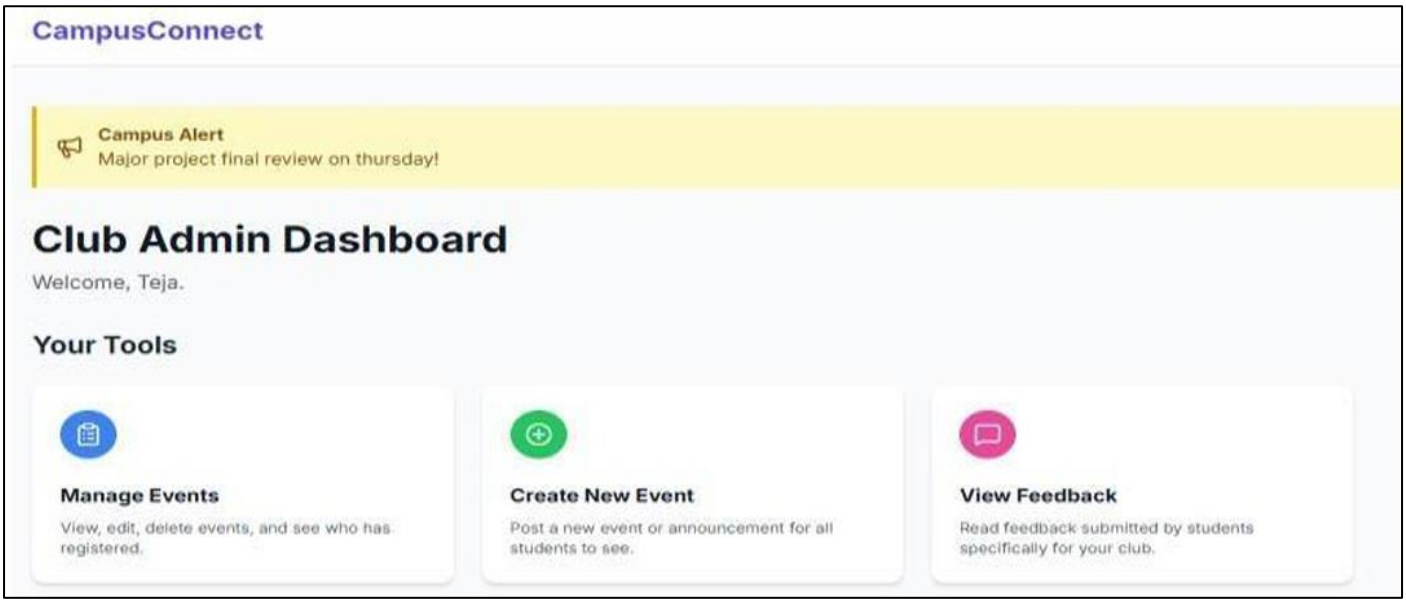

Fig 10 Club Admin Dashboard

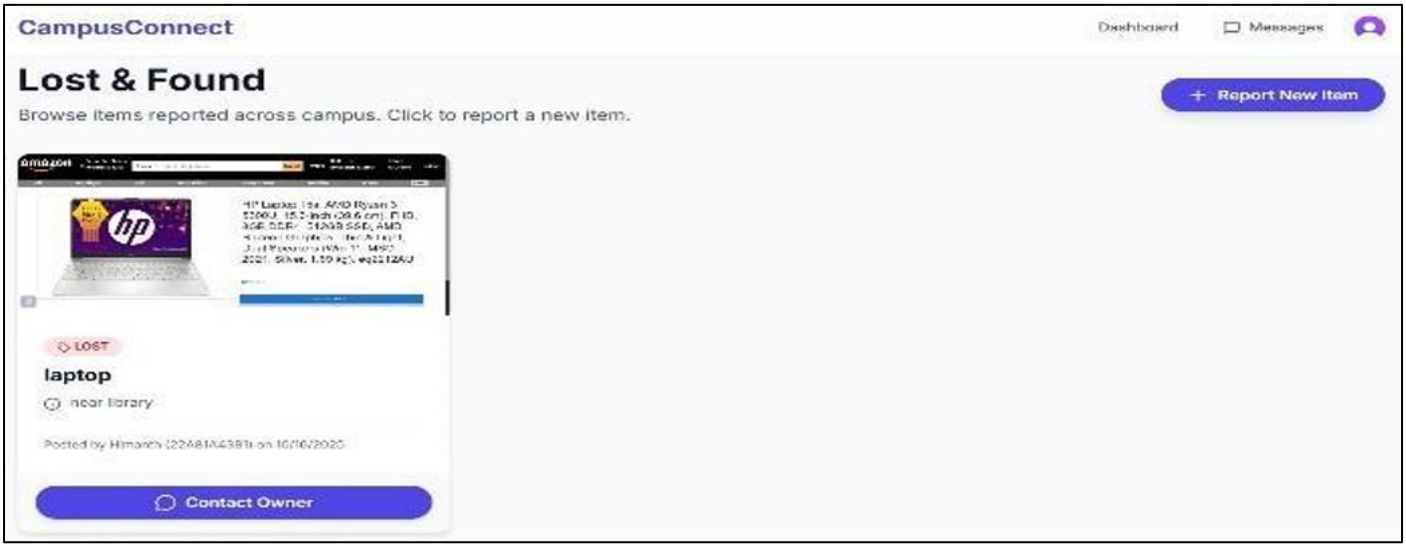

Fig 11 Lost and Found Request

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

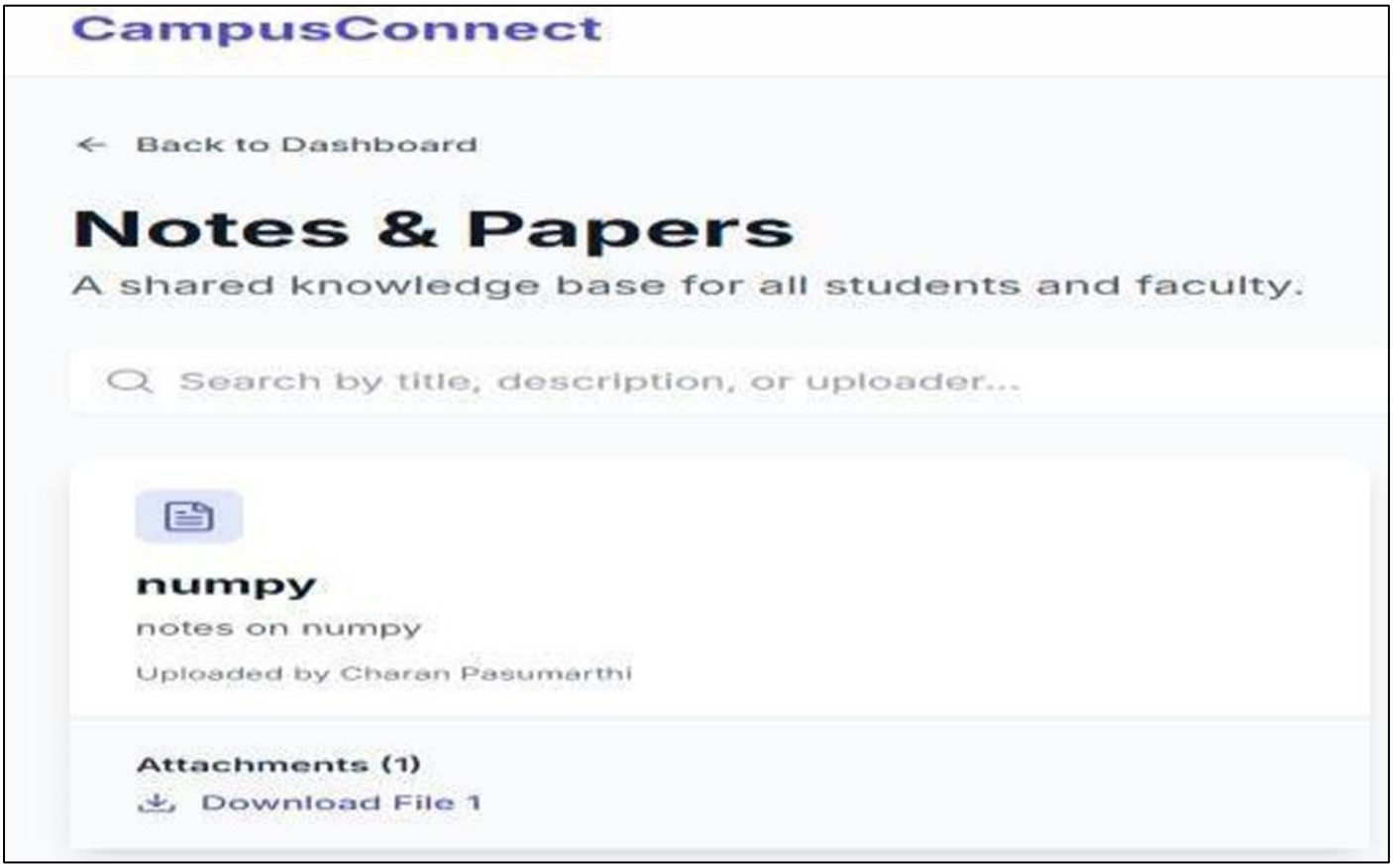

Fig 12 Notes Sharing

The CampusConnect system implemented is presented on the screenshots to disclose some dominant characteristics and user interfaces. These include the role specific dashboards, real time requests and feedback interface, event management modules and administrative control panels. The screenshots prove efficient implementation of the main functions and user processes applied in this paper.

VII. CONCLUSION AND FUTURE WORK

CampusConnect project was developed to eliminate these issues as the inappropriate communication and decentralization of service management that is characteristic of higher learning institutions. The result of this work consisted in the successful implementation of one, real-time web application that gathers primary services of the campus. The system is developed on a current technology stack, which includes React and Supabase, which meets the expected performance, scalability, maintainability, thus making the system a sure way out of the identified problem.

Artificial Intelligence implementation through the Gemini API is one of the greatest contributions of this project. Smart automation, i. e. AI chatbot and note summarizer, may help facilitate the tasks of the administration and improve the performance of students. These characteristics demonstrate that AI plays a crucial role in making helpful changes to everyday interactions on campus.

Generally, CampusConnect attains the intended objectives since it has simplified the administrative processes and made life as a student more enjoyable. The project represents a pilot project that is helpful in demonstrating the feasibility of intelligent and integrated systems in the educational environment and preconditions the building of more interlaced and responsive campuses.

➢ *Future Work*

Even though the current version of CampusConnect is a strong application, the application has numerous opportunities to grow and develop further. The recommendations that could be enumerated as valuable follow-ups are as follows:

- Native Mobile Application: Native Mobile Application Developing this is the development of even a greater mobile experience by writing iOS and Android applications specifically, and can make better use of native features like push notification.
- Campus Calendar Integration: The option to add calendar support with university-official calendar (i.e. Google Calendar) so that the academic deadlines, holidays and university events could be added automatically.
- Interactive Campus Map: This is a module that will be added as a campus map that will include building locations, navigation, and other places of interest.
- Smart Notifications & Campus Alerts: Implement a rolebased notification system to send instant alerts for

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar945

important updates such as medication request status, lostand-found updates, event announcements, and administrative notices to students, HODs, and admins.

- Administrative Analytics Dashboard: Develop a centralized dashboard for administrators to track student requests, medication approvals, lost-and-found activity, and feedback trends, helping management make data-driven decisions.

REFERENCES

[1]. Reddy, P., Sharma, K., & Verma, A. (2023). *Mafqodat – A dedicated lost and found mobile application*. In Proceedings of the 2023 International Conference on Smart Computing and Informatics (pp. 112–118). IEEE.

[2]. Kumar, S., Patel, R., & Nair, V. (2022). *A system for college resources and event management*. International Journal of Advanced Computer Science and Applications, 13(6), 421– 427.

[3]. Joshi, A., Mehta, D., & Kulkarni, P. (2023). *"Notely" – A cloud-based note sharing system*. In 2023 IEEE International Conference on Cloud Computing and Emerging Technologies (pp. 89–95). IEEE.

[4]. Singh, R., Malhotra, N., & Choudhary, A. (2024). *An AI- powered conversational agent for student services*. In 2024 IEEE Frontiers in Education Conference (FIE) (pp. 1–8). IEEE.

[5]. Thomas, J., Lee, M., & Park, S. (2022). *A real-time system for campus-wide alerts and student feedback*. Journal of Educational Technology Systems, 51(2), 173–189.

[6]. J. Raja Shanmugam, P. Thirunavukarasu, and T. Ragunathan, "Event Management System on Web Platform," in *Proc. 2nd Int. Conf. on Latest Innovations in Applied Science, Technology & Engineering*, March 2018, pp. 1–6.

[7]. R. L. Aspa, M. C. Ditche, J. A. Atos, M. D. Bagaporo, and J. R. Sarmiento, "Campus Activities Management System with SMS Reminders," *Global Scientific Journal*, vol. 12, no. 6, June 2024.

[8]. Aditi Chaturvedi, K. Sharma, A. Dua, and A. Gupta, "CU-Events: A Comprehensive Event Management System for University," *Int. J. Research Appl. Science & Eng. Technology*, Nov. 2024.

[9]. *Design and Implementation of Web-Based Digital Campus Navigation and Orientation System for New Student, Int. Res. J. Innov. Eng. Technol.*, vol. 9, no. 2, 2025.

[10]. T. Bagade, A. Wagh, V. Kshirsagar, and S. Ubarhande, "AI-Powered College Review Chatbot for Student Decision-Making," *IJSRET*, vol. 11, no. 3, May–Jun. 2025.

[11]. *Student Feedback Systems: Developing a Web-Based Solution for Efficient Complaint Processing, Int. J. Research Publication & Reviews*, Vol. 4, 2025.

[12]. D. Walanjkar, "Boosting Campus Life with Real-Time Event Updates and Incentives," *Int. J. Future Modern Res.*, Jun. 2024.

[13]. *A Review of University Chatbots for Student Support: FAQs and Beyond*, *Discover Education*, vol. 4, 2025.

[14]. "A Chatbot Student Support System in Open and Distance Learning Institutions," *Computers*, vol. 14, no. 3, 2025.

[15]. J. Tang, S. Chen, and Y. Shang, "TigerGPT: A New AI Chatbot for Adaptive Campus Climate Surveys," *arXiv preprint arXiv:2504.13925*, 2025.

[16]. H. Tanwar, K. Shrivastva, R. Singh, and D. Kumar, "OpineBot: Class Feedback Reimagined Using a Conversational LLM," *arXiv preprint arXiv:2401.15589*, 2024.

[17]. S. Agarwal, R. Bansal, and P. Malviya, "Smart campus management system using cloud computing," *International Journal of Computer Applications*, vol. 182, no. 25, pp. 1– 6, 2024.

[18]. K. Verma, A. Singh, and S. Rao, "Design and development of a role-based web portal for university administration," *International Journal of Engineering Research & Technology (IJERT)*, vol. 13, no. 4, pp. 512–518, 2024.

[19]. M. Al-Shehri and A. Khan, "A secure web-based student information and service management system," *Journal of Information Systems Education*, vol. 35, no. 1, pp. 44–55, 2024.

[20]. P. Choudhary, N. Gupta, and S. Jain, "Implementation of real-time chat applications using WebSocket and cloud backends," *International Journal of Advanced Research in Computer Science*, vol. 15, no. 2, pp. 87–93, 20