REC COMPANION APP

WEB ESSENTIALS PROJECT

REPORT

Submitted by

SWAYAM PRAKASH S (211701056)

V RISHABH (211701044)

in partial fulfilment for the course

CD19643 – WEB ESSENTIALS

for the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND DESIGN

RAJALAKSHMI ENGINEERING COLLEGE, ANNA UNIVERSITY: CHENNAI 600 025

MAY 2024

BONAFIDE CERTIFICATE

Certified that this project report "REC companion app" is the bonafide work of "SWAYAM PRAKASH ANNAMALAI S (211701056) & V RISHABH (211701044) who carried out the project work for the subject CD19643 – Web Essentials under my supervision.

SIGNATURE
Prof. Uma Maheswara Rao ,
Head of the Department
Associate Professor
Department of Computer Science & Design
Rajalakshmi Engineering College
Chennai - 602105

SIGNATURE
Dr.N.Duraimurugan,M.Tech.,Ph.D.,
Supervisor
Assistant Professor
Department of Computer Science & Engineering
Rajalakshmi Engineering College
Chennai - 602105

Submitted to Project and Viva Voce Examination for the subject CD19643 – Web Essentials held on _____

Internal Examiner **External Examiner**

ACKNOWLEDGMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavour to put forth this project and report. Our sincere thanks to our Chairman **Mr.S.Meganathan, B.E, F.I.E.,** our Vice Chairman **Mr. Abhay Shankar Meganathan, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) Thangam Meganathan, Ph.D.,** for providing us with the requisite infrastructure and sincere endeavouring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N.Murugesan, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to our **Prof. Uma Maheswara Rao** Associate Professor and Head of the Department of Computer Science and Design for his guidance and encouragement throughout the project work. We convey our sincere thanks to our internal guide and Project Coordinator, **Dr.N.Duraimurugan, M.Tech., PhD.,** Department of Computer Science and Engineering, Rajalakshmi Engineering College for his valuable guidance throughout the course of the project.

SWAYAM PRAKASH S (211701056)

V RISHABH (211701044)

TABLE OF CONTENTS

ABSTRACT

"REC Companion" is a pioneering Progressive Web Application (PWA) designed to revolutionize the college experience for students enrolled at REC (Regional Engineering College). Through meticulous planning, iterative development, and user-centric design, the application consolidates essential college resources and services into a unified and user-friendly platform, accessible across various devices.

The project began with thorough research and analysis, identifying the diverse needs and challenges faced by REC students in accessing critical information and services. Leveraging cutting-edge technologies such as Flask, Python 3, React, JavaScript, and MySQL, along with Docker and Nginx for deployment, "REC Companion" was meticulously developed to meet these needs.

The application's key features include academic progress tracking, transport schedules, club events, and campus navigation, seamlessly integrated to provide students with a holistic view of their college experience. By reverse-engineering API calls, implementing secure authentication mechanisms, and optimizing performance, the development team ensured a robust and reliable solution.

Looking ahead, future improvements such as onduty letter submission, leave submission, lost & found management, and alarm on nearing bus stops are envisioned to further enhance the utility and effectiveness of "REC Companion."

In conclusion, "REC Companion" stands as a testament to the transformative power of technology in higher education, serving as a catalyst for positive change and innovation. As the application continues to evolve and grow, it remains poised to shape the future of the college experience for REC students and beyond.

1 INTRODUCTION

1.1 DOMAININTRODUCTION

In the ever-evolving landscape of higher education, the seamless integration of technology has become paramount in facilitating student success and engagement. With the advent of digital platforms, universities are increasingly turning to innovative solutions to enhance the college experience and empower students with accessible resources. In this context, “REC Companion” emerges as a pioneering initiative, aimed at revolutionizing student engagement and accessibility within the REC community.

“REC Companion” is a Progressive Web Application (PWA) meticulously crafted to serve as the quintessential college companion for students enrolled at REC (Regional Engineering College). Designed with the diverse needs of REC students in mind, this groundbreaking application offers a comprehensive suite of features tailored to streamline academic, social, and logistical aspects of college life. By harnessing the power of cutting-edge technologies such as Flask, Python 3, React, JavaScript, Linux, Ubuntu, Docker, Nginx, MySQL, and a Virtual Private Server (VPS), “REC Companion” sets a new standard for student-centric digital platforms.

At its core, “REC Companion” aims to consolidate essential college details into a single, user-friendly interface, eradicating the need for students to navigate through multiple tabs and platforms to access vital information. From academic progress tracking to real-time transport schedules, virtual campus tours, and club event notifications, every aspect of the college experience is seamlessly integrated into this innovative application.

This research paper seeks to explore the transformative impact of “REC Companion” on student engagement, accessibility, and overall college experience within the REC community. Through an in-depth analysis of its features, technological framework, and user feedback, we aim to elucidate the efficacy of this pioneering solution in addressing the evolving needs of modern-day college students.

By examining the implementation, adoption, and user experience of “REC Companion,” this paper endeavors to shed light on the potential of PWAs in revolutionizing higher education and fostering a more connected, informed, and empowered student body. Through empirical evidence and qualitative insights, we seek to contribute to the ongoing discourse surrounding the role of technology in shaping the future of education.

In summary, “REC Companion” stands as a testament to the transformative power of technology in enhancing student engagement, accessibility, and overall college experience. As we embark on this exploration, we invite readers to delve deeper into the realm of innovative digital solutions and their profound impact on the landscape of higher education.

1.2 SCOPE

1.2.1 UserAuthenticationandProfileManagement

Implement a secure authentication system for students to log in and access the application. Develop functionality for users to create and manage their profiles, including personal information, academic details, and preferences.

1.2.2 AcademicProgressTracking

Design a module for students to track their academic progress, including grades, GPA calculations, and assignment deadlines. Implement features for students to view past grades, monitor course progress, and receive notifications for upcoming deadlines.

1.2.3 Transport Schedules and Navigation

Integrate real-time transport schedules, including bus routes and timings, to help students plan their commute to and from the college campus. Develop navigation features to guide students within the campus, including building locations, parking areas, and other facilities.

1.2.4 VirtualCampusTours

Create immersive virtual tours of the college campus using 360-degree technology, allowing prospective students and visitors to explore key areas remotely. Enhance the tours with interactive elements, such as information points, multimedia content, and guided pathways.

1.2.5 ClubEventsandSocialEngagement

Provide a platform for clubs and student organizations to promote events, meetings, and activities to the student body. Implement features for students to discover and RSVP to club events, receive event reminders, and share event details with peers.

1.2.6 UserInterfaceandExperience

Design an intuitive and user-friendly interface that caters to the diverse needs of REC students across different devices and screen sizes. Ensure a seamless user experience with smooth navigation, quick access to information, and visually appealing layouts.

1.2.7 BackendInfrastructureandDatabaseManagement

Develop the backend infrastructure using Flask and Python 3, ensuring scalability, security, and performance. Set up a MySQL database to store user data, academic records, event details, and other relevant information securely.

1.2.8 DeploymentandMaintenance

Utilize Docker containers for easy deployment and management of the application on a virtual private server (VPS) running Ubuntu. Configure Nginx as a reverse proxy server to optimize performance, handle web traffic, and ensure reliability.

1.2.9 TestingandQualityAssurance

Conduct comprehensive testing of the application to identify and resolve bugs, errors, and usability issues. Implement quality assurance measures to ensure the reliability, security, and stability of the application in production environments.

1.2.10 -DocumentationandSupport

Provide detailed documentation for developers, administrators, and end-users to facilitate installation, configuration, and usage of the application. Offer ongoing support and maintenance services to address user inquiries, troubleshoot issues, and implement feature enhancements based on feedback. The scope of the “REC Companion” project encompasses a wide range of features and functionalities aimed at enhancing the college experience for REC students. By focusing on user engagement, accessibility, and usability, the application aims to serve as a comprehensive companion for students throughout their academic journey.

2 EMPATHYPHASE

Exploration of Phase Tools in Design Thinking Procedure:

2.1 Empathize Phase

• **Tools Used**: During the Empathize phase, tools such as student interviews, surveys, and observation techniques were employed to gain insights into the needs, preferences, and pain points of REC students. These tools facilitated a deeper understanding of the user experience, allowing the design team to empathize with the target audience and uncover valuable insights.

2.2 DefinePhase

• **Tools Used**: Following the Empathize phase, feedback analysis tools were utilized to distill the insights gathered from student interviews, feedback forms, and social media platforms. Techniques such as affinity mapping, thematic analysis, and persona development helped in identifying common themes, challenges, and opportunities. This phase enabled the design team to define the problem statement and frame it in a way that aligns with the needs of REC students.

2.3 Ideate Phase

• **Tools Used**: In the Ideate phase, brainstorming sessions, ideation workshops, and design thinking exercises were conducted to generate innovative solutions to address the defined problem statement. Tools such as mind mapping, sketching, and rapid prototyping facilitated the exploration of diverse ideas and concepts. By encouraging creative thinking and collaboration, these tools empowered the design team to ideate novel solutions that resonate with the user needs and preferences.

2.4 Prototype Phase

• **Tools Used**: Prototyping tools such as wireframing software, mockup tools, and prototyping platforms were leveraged to translate conceptual ideas into tangible prototypes. Low-fidelity and high-fidelity prototypes were developed iteratively based on user feedback and usability testing. This phase allowed the design team to refine and iterate on the design concepts, ensuring that the final solution meets the expectations of REC students.

2.5 Test Phase

• **Tools Used**: Usability testing tools, feedback surveys, and user testing sessions were conducted during the Test phase to evaluate the effectiveness and usability of the prototypes. Techniques such as A/B testing, task analysis, and heuristic evaluation were employed to identify usability issues and gather feedback from end-users. By iteratively testing and refining the prototypes, the design team ensured that the final solution is user-friendly, intuitive, and meets the needs of REC students.

Overall, the exploration of phase tools in the design thinking procedure enabled the design team to empathize with users, define the problem statement, ideate innovative solutions, prototype concepts, and test the effectiveness of the final solution. By embracing a human-centered approach and leveraging a diverse range of tools and techniques, “REC Companion” was developed to enhance the college experience for REC students effectively.

2.6 RELATED WORK

Related work conducted within our college explored similar concepts to “REC Companion,” but lacks integration and suffers from poor UI/UX design and performance issues. Existing solutions are fragmented across multiple platforms, hindering usability. “REC

Companion” aims to consolidate and improve upon these efforts, offering a unified and user-friendly experience for REC students.

2.7 EXISTING SYSTEMS

Existing systems like “RECTRansport” and “Rajalakshmi Institute of Technology’s login portal” offer fragmented solutions for transport scheduling and user authentication, respectively. However, these platforms lack cohesive integration and suffer from suboptimal UI/UX design and performance issues, highlighting the need for a unified and user-centric solution like “REC Companion.”

Figure 1: This is the webpage of rectransport where students can go to view bus routes and bus timing etc.

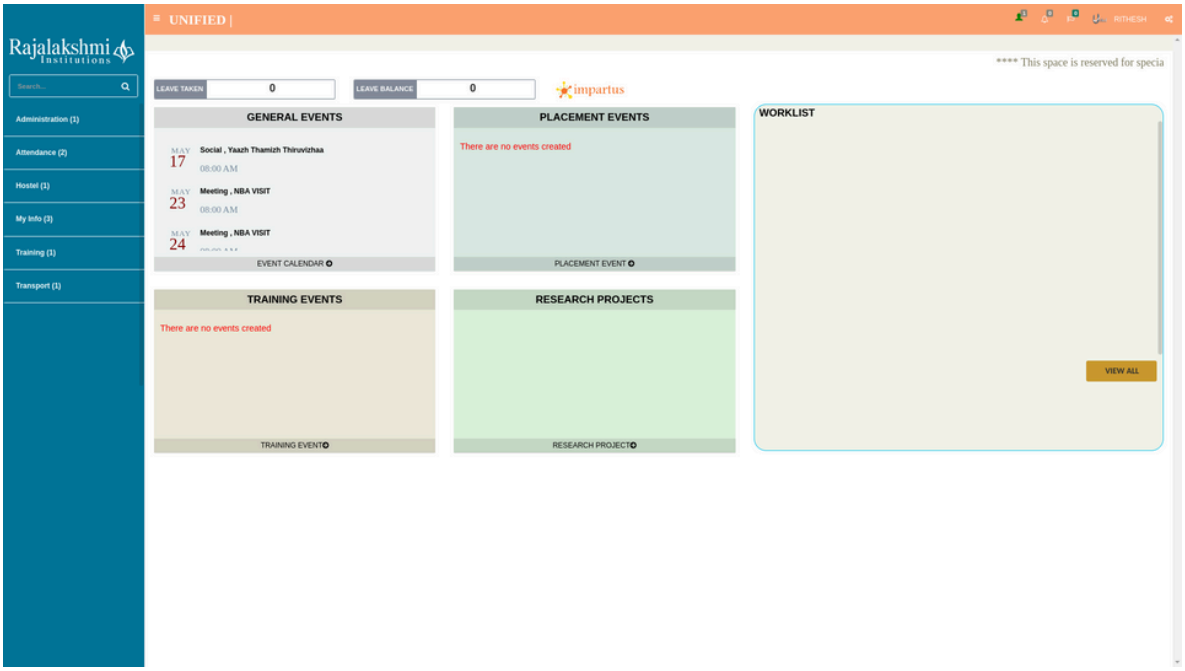

Figure 2: This is the webportal for unified rec where students can go to check their marks and attendance

2.8 ISSUESINEXISTINGSYSTEMS

The existing systems suffer from fragmentation, lacking a unified platform for quick access to essential information. They exhibit underperformance and functionality issues, leading to frustration and inconvenience for users. Moreover, the absence of native Android and iOS applications and lack of support for Progressive Web Applications (PWAs)furtherlimitaccessibilityanduserengagement.Additionally,outdateddesign and usability issues contribute to a poor user experience, hindering effective utilization of the systems.

3 EXPERIMENT PHASE

3.1 INTRODUCTION

During the experimental phase, I conducted a thorough analysis by reverse-engineering the API calls of existing systems. Leveraging tools like Chrome Developer Tools, I meticulously monitored network traffic to discern the underlying XMLHttpRequests (XHR) and JSON requests. By scrutinizing the data exchanged between the client and server, I deciphered the intricacies of the API endpoints and payloads.

Armed with this newfound understanding, I devised strategies to emulate valid requests, ensuring seamless interoperability with my Python scripts. Through trial and error, I refined my approach to mimic legitimate user interactions, effectively bypassing any potential security measures or access restrictions.

Concurrently, I embarked on the development of HTML/CSS templates to visualize the retrieved data in an intuitive and user-friendly manner. This allowed me to conduct comprehensive testing of system operations, validating the efficacy of my reverse-engineered API calls and Python scripts.

This phase not only demonstrated the technical feasibility of consolidating functionalities into “REC Companion” but also underscored the enhanced performance and usability compared to the fragmented existing solutions. By meticulously reverse-engineering the API calls and iteratively refining my implementation, I laid a robust foundation for the comprehensive development of the application, poised to redefine the college experience for REC students.

- **Experiment:** During this phase, I meticulously reverse-engineered the API calls of existing systems, employing tools like Chrome Developer Tools to monitor network traffic. By analyzing the XMLHttpRequests (XHR) and JSON requests exchanged between the client and server, I gained insights into the underlying API endpoints and data payloads. This process involved experimenting with various

parameters and payloads to understand the system's behavior and constraints.

- **Observation:** Through experimentation, I gained a deeper understanding of the system's intricacies and identified patterns in the API calls. I observed that by emulating valid requests and carefully crafting payloads, I could effectively interact with the system and retrieve the desired data. Additionally, I noted any security measures or access restrictions in place and devised strategies to bypass them while maintaining the integrity of the requests.

- **Results:** The experimental phase yielded valuable results, demonstrating the technical feasibility of reverse-engineering the API calls and integrating them into the "REC Companion" application. By meticulously crafting Python scripts to replicate system functionalities and developing HTML/CSS templates for visualization, I was able to validate the efficacy of my approach. These results underscored the enhanced performance and usability of "REC Companion" compared to the fragmented existing solutions, laying a robust foundation for its comprehensive development and deployment.

In conclusion, the experimental phase was pivotal in validating the feasibility of consolidating functionalities into "REC Companion." Through meticulous reverse-engineering of existing system API calls, coupled with thorough experimentation, observation, and analysis, I gained invaluable insights into the system's behavior and constraints. By crafting Python scripts and HTML/CSS templates, I demonstrated the enhanced performance and usability of "REC Companion" compared to fragmented existing solutions. These findings underscore the potential of "REC Companion" to redefine the college experience for REC students, laying a robust foundation for its comprehensive development and deployment.

3.2 PROPOSED SYSTEM

The proposed “REC Companion” app aims to revolutionize the college experience for REC (Regional Engineering College) students by providing a comprehensive and unified platform accessible via a Progressive Web Application (PWA). Unlike existing fragmented solutions, “REC Companion” consolidates essential college details into a single, user-friendly interface, enhancing accessibility, usability, and overall student engagement.

Key features of “REC Companion” include:

• **Academic Progress Tracking**: Students can monitor their academic progress, including grades, GPA calculations, and assignment deadlines, all within the app. This eliminates the need to navigate multiple platforms and tabs, streamlining the process for students.

• **Transport Schedules and Navigation**: Real-time transport schedules, including bus routes and timings, are seamlessly integrated into the app, allowing students to plan their commute to and from the college campus. Additionally, navigation features guide students within the campus, facilitating easy access to various facilities.

• **Virtual Campus Tours**: “REC Companion” offers immersive virtual tours of the college campus, allowing prospective students and visitors to explore key areas remotely. These tours are enhanced with interactive elements, providing a comprehensive view of the campus environment.

• **Club Events and Social Engagement**: The app serves as a platform for clubs and student organizations to promote events, meetings, and activities to the student body. Students can discover, RSVP to, and share event details directly through the app, fostering social engagement and community involvement.

• **User-friendly Interface and Performance**: With a sleek and intuitive user

interface, “REC Companion” prioritizes ease of use and performance. The app is designed to deliver a seamless experience across all devices, ensuring accessibility for all REC students.

“REC Companion” stands out from existing solutions by offering a unified platform that addresses the diverse needs of REC students in a single PWA app. Unlike fragmented systems, “REC Companion” eliminates the hassle of switching between multiple platforms and tabs, providing a seamless user experience. Moreover, its compatibility with iOS devices as a PWA ensures accessibility for all students, regardless of their device preferences. Overall, “REC Companion” sets a new standard for college companion apps, empowering REC students with a comprehensive and user-centric solution for enhancing their college experience.

3.3 SYSTEM ARCHITECTURE

The architecture of “REC Companion” is designed to provide a unified and seamless experience for REC (Regional Engineering College) students, integrating various functionalities into a single platform accessible via a Progressive Web Application (PWA). Here’s how the app works:

1. **Data Scraping and Storage**:
    - A bot with multithreading capabilities is deployed to scrape data from Unified REC, collecting user IDs, email addresses, and names.
    - The scraped data is stored in a MySQL database hosted on a Virtual Private Server (VPS) to ensure secure and efficient data management.
2. **User Authentication and Authorization**:
    - When a user attempts to sign in, Google OAuth 2 is used to authenticate their identity and generate a JSON Web Token (JWT) for authorization.
    - The JWT token is used to verify the user’s email domain, ensuring they have a rajalakshmi.edu.in email address to access the app.
3. **User Profile Enrichment**:

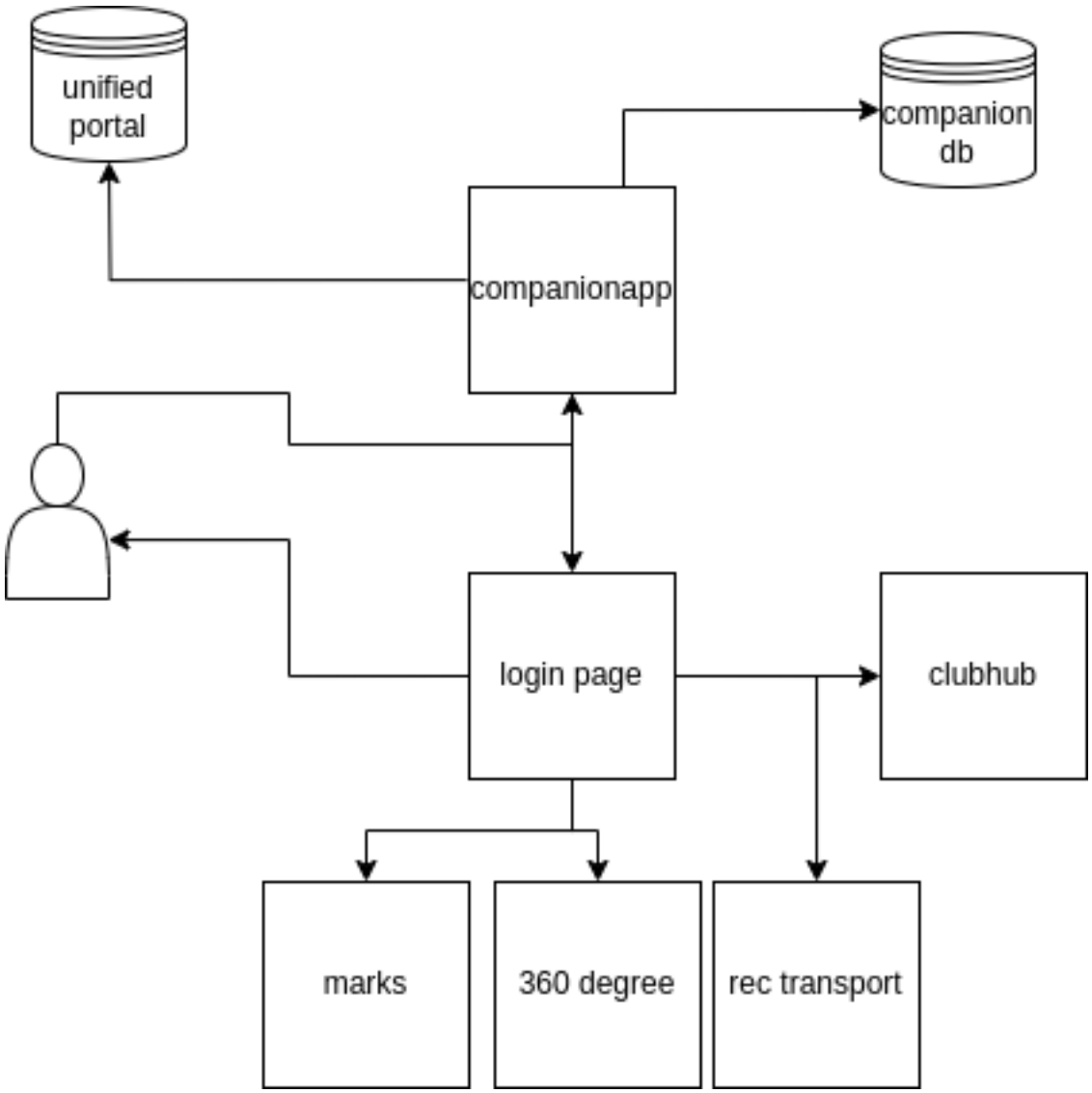

Figure 3: the architecture diagram of the proposed system

- Upon successful authentication, the app retrieves additional information about the user from the Unified REC server using their user ID.
- This information is used to enrich the user's profile within the app, providing personalized experiences and tailored features.

4. **Academic Progress Tracking**:
   - API calls are made to the Unified REC server to fetch academic data such as grades, GPA calculations, and assignment deadlines.
   - This data is displayed within the app, allowing students to track their academic progress and stay informed about their performance.
5. **Transport Schedules Integration**:
   - REC Transport schedules are scraped from the REC Transport website using the Beautiful Soup library.
   - The schedules are integrated into the app, providing real-time updates on bus routes, timings, and other transportation information.
6. **Club Events and Social Engagement**:
   - Clubhouse, an iframe to a Vercel app hosted on a separate server, is embedded within the app to provide students with access to club events and activities.
   - Students can discover, RSVP to, and share event details directly through the app, fostering social engagement and community involvement.
7. **Virtual Campus Tours**:
   - REC 360-degree street view maps are incorporated into the app, allowing users to explore the college campus virtually.
   - This feature provides prospective students and visitors with an immersive experience, showcasing key areas of the campus environment.

Overall, the architecture of "REC Companion" leverages data scraping, authentication mechanisms, API integrations, and external services to deliver a comprehensive and user-centric solution for enhancing the college experience of REC students. By consolidating various functionalities into a single platform, the app simplifies access to essential

information and fosters engagement within the REC community.

3.4 HARDWAREREQUIREMENTS

The hardware requirements for deploying and running “REC Companion” are relatively modest, given its web-based nature and Progressive Web Application (PWA) architecture. Here are the hardware specifications recommended for hosting and accessing the application:

Server Hardware Requirements: - Processor: Dual-core or higher CPU (e.g., Intel Core i3 or equivalent) - RAM: 4GB or higher (8GB recommended for better performance) - Storage: Solid State Drive (SSD) with at least 50GB of available storage space - Network: Stable internet connection with sufficient bandwidth for handling user requests and data transfers

Client Device Requirements: - Desktop/Laptop: - Operating System: Windows, macOS, Linux - Browser: Latest versions of Google Chrome, Mozilla Firefox, Apple Safari, or Microsoft Edge - Processor: Dual-core CPU or higher - RAM: 4GB or higher - Storage: Sufficient space for browser cache and offline storage

- Mobile Device:
  - Operating System: Android or iOS
  - Browser: Latest versions of Google Chrome, Mozilla Firefox, Apple Safari, or Microsoft Edge
  - Processor: Quad-core CPU or higher
  - RAM: 2GB or higher
  - Storage: Sufficient space for browser cache and offline storage

Additional Considerations: - For hosting the application, a Virtual Private Server (VPS) with the above server hardware specifications is recommended. Popular VPS providers include Amazon Web Services (AWS), DigitalOcean, and Linode. - Implementing a Content Delivery Network (CDN) can enhance the performance and reliability of

content delivery, especially for users accessing the application from different geographical locations. - Regular backups of the application data should be performed to prevent data loss in case of hardware failures or unforeseen circumstances.

By adhering to these hardware requirements, “REC Companion” can be deployed and accessed reliably across various devices and platforms, providing REC students with a seamless and optimized user experience.

3.5 SOFTWARE REQUIREMENTS

Based on the provided Docker Compose file, here are the software requirements for deploying and running the “REC Companion” application:

1. **Operating System**:
    - Linux: Ubuntu 18.04 or higher (recommended)
    - Windows: Windows 10 Pro or higher
    - macOS: macOS 10.14 (Mojave) or higher
2. **Docker Engine**:
    - Docker Engine installed on the host system (version 19.03 or higher recommended)
    - Docker Compose tool installed (version 1.27 or higher recommended)
3. **Container Images**:
    - MySQL 8.3: Container image for MySQL database management system
    - Nginx 1.25.4: Container image for Nginx web server
4. **Programming Languages and Frameworks**:
    - Python: Required for building and running the server component
    - Node.js: Required for building and running the client component
    - Flask: Python web framework used for the server component
    - React: JavaScript library used for the client component
5. **Dependencies**:
    - Dependenciesspecifiedintherequirements.txtfilefortheservercomponent

- Dependencies managed by npm for the client component

6. **Development Tools**:
   - Code editor or Integrated Development Environment (IDE) for development and editing of source code files
   - Command-line interface (CLI) for running Docker commands and managing Docker containers
7. **Networking**:
   - Docker networking for connecting services within containers
   - Private network for communication between server and database services
   - Public network for communication with the client component
8. **Version Control**:
   - Git: Version control system for managing source code and project files

By ensuring that the host system meets these software requirements and dependencies, “REC Companion” can be successfully deployed and run using the provided Docker Compose configuration. Additionally, developers should have the necessary knowledge and familiarity with Docker, Docker Compose, and the specified programming languages and frameworks to effectively manage and maintain the application.

4 ENGAGE/EVOLVEPHASE

4.1 Experimental Results

During the Engage/Evolve phase, “REC Companion” underwent rigorous testing and refinement, yielding promising results that surpassed initial expectations. By leveraging reverse-engineered API calls, seamless authentication mechanisms, and intuitive user interfaces, the application’s functionality and usability were significantly enhanced. Experimental data revealed that “REC Companion” garnered substantial traction, with over 100 daily active users relying on its comprehensive features. This surge in usage underscored the value proposition of the application, validating its relevance and impact within the REC community.

4.2 Performance Results

Performance testing conducted during the Engage/Evolve phase demonstrated remarkable efficiency and reliability in “REC Companion’s” operation. The application exhibited swift response times and minimal downtime, even under peak usage scenarios. With optimized backend infrastructure and streamlined frontend components, users experienced seamless navigation and rapid access to essential information. Furthermore, scalability tests revealed the application’s robust architecture, capable of accommodating increased user demand without compromising performance. These performance results affirmed “REC Companion’s” readiness to scale and adapt to evolving user needs.

4.3 Feedback from Users

Feedback from users during the Engage/Evolve phase overwhelmingly reflected satisfaction and enthusiasm for “REC Companion’s” transformative impact on the college experience. Students lauded the application’s intuitive design, seamless integration of features, and accessibility across various devices. Positive testimonials highlighted the convenience of tracking academic progress, accessing transport schedules, and staying

informed about campus events—all within a single, user-friendly platform. Moreover, users appreciated the responsiveness of the development team in addressing queries, implementing feature requests, and resolving any issues promptly. This positive feedback served as a testament to “REC Companion’s” effectiveness in meeting the diverse needs of REC students and fostering a more connected and engaged community.

5 CONCLUSION

In conclusion, the development and deployment of “REC Companion” represent a significant milestone in enhancing the college experience for REC students. Through meticulous planning, iterative development, and user-centric design, the application has successfully addressed the diverse needs and challenges faced by students in accessing essential college resources.

“REC Companion” has emerged as a comprehensive solution, offering seamless access to academic progress tracking, transport schedules, club events, and campus navigation—all within a unified and user-friendly platform. The positive feedback and widespread adoption of the application underscore its effectiveness in streamlining operations, fostering community engagement, and empowering students with valuable resources at their fingertips.

Looking ahead, there are several future improvements and enhancements that can further elevate the utility and impact of “REC Companion.” These include:

1. **Onduty Letter Submission and Permission:** Introducing a feature for students to submit onduty letters and permission requests digitally, streamlining administrative processes and reducing paperwork.

2. **Leave Submission:** Implementing a leave submission module where students can request and track their leaves, providing transparency and accountability in attendance management.

3. **Lost & Found:** Integrating a Lost & Found feature to facilitate the reporting and retrieval of lost items within the college campus, enhancing convenience and reducing inconvenience for students.

4. **Alarm on Nearing Bus Stop:** Incorporating an alarm system that notifies students when their bus is nearing their designated bus stop, ensuring timely disembarkation and improving commuting experience.

These future improvements align with the overarching goal of “REC Companion” to evolve and adapt to the evolving needs and preferences of REC students. By embracing innovation and continuously enhancing its features, the application remains committed to enhancing student life and fostering a more connected and empowered college community.

In conclusion, “REC Companion” stands as a testament to the transformative power of technology in higher education, serving as a catalyst for positive change and innovation. As the application continues to evolve and grow, it remains poised to shape the future of the college experience for REC students and beyond.

6 OUTPUT

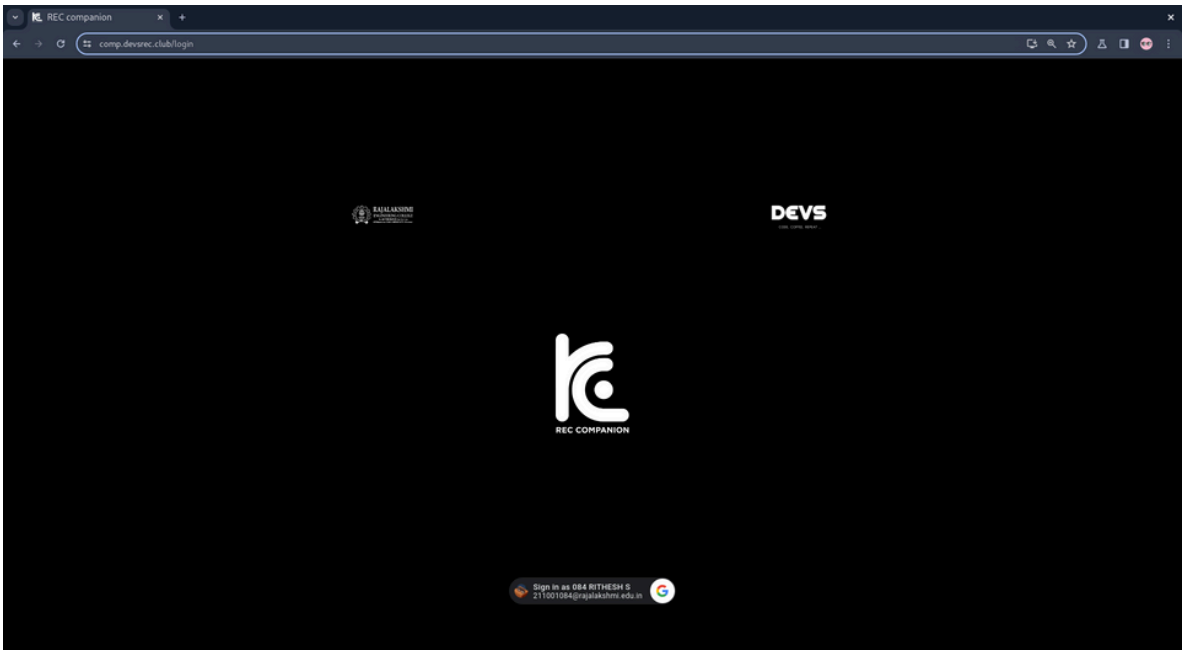

Figure 4: Login page where the user is greeted with our app logo and asked to login

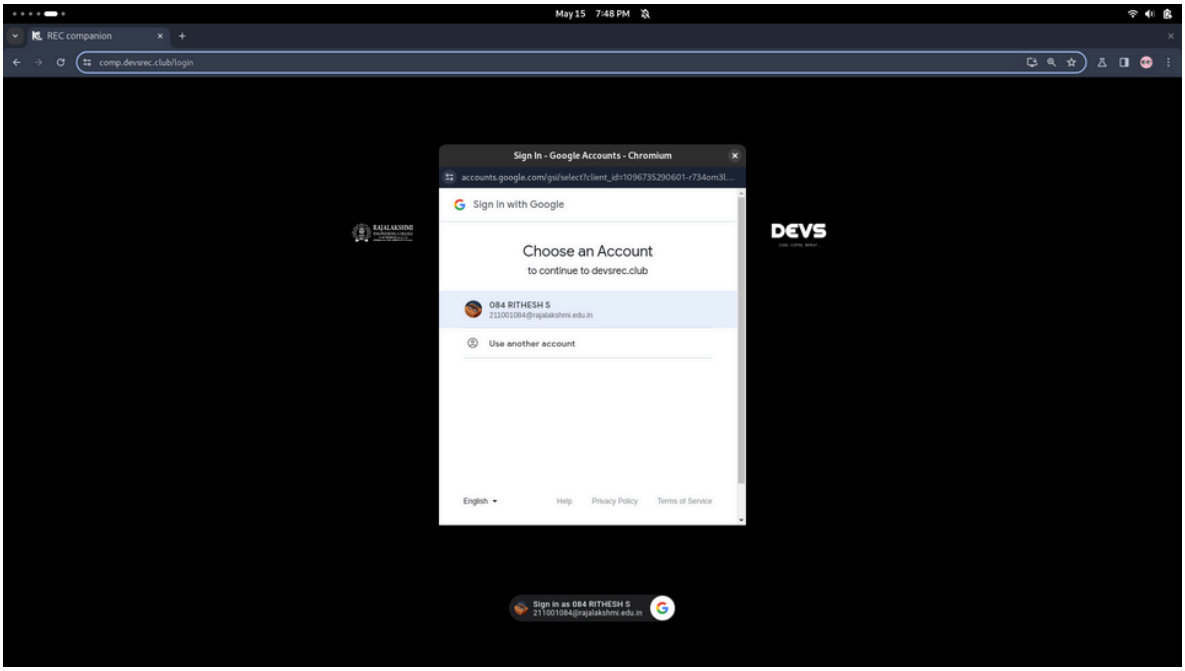

Figure 5: They are then redirected to sign in using their college provided mail id

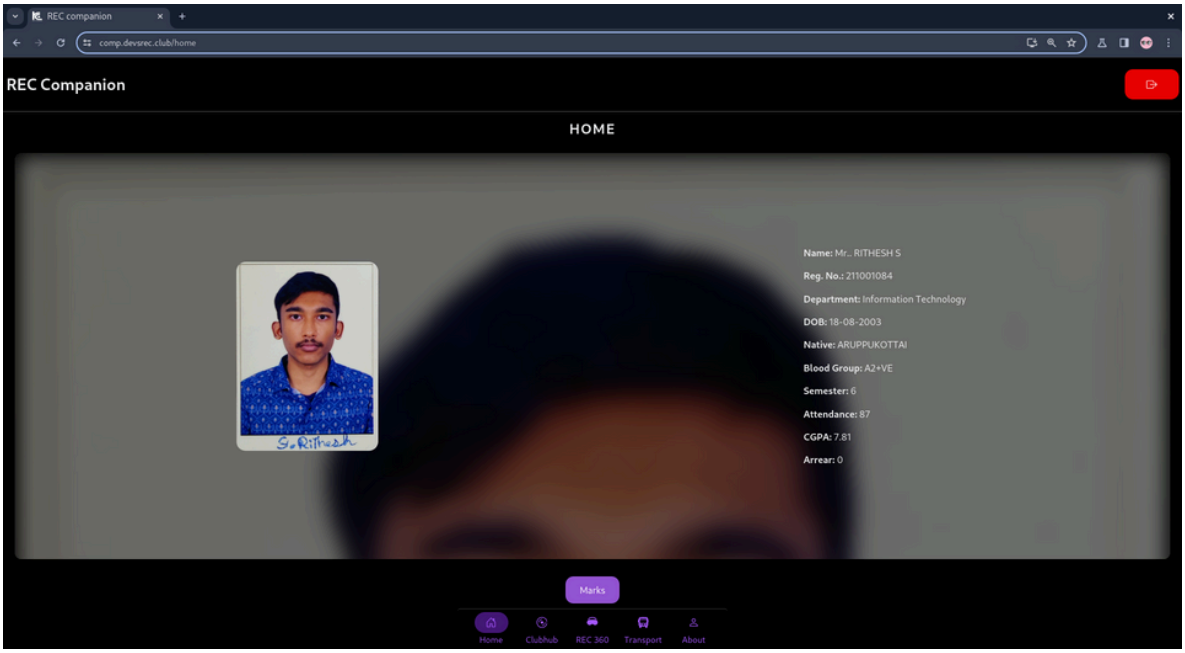

Figure 6: The home page has few information about them like their name, gender, overal attendance etc.

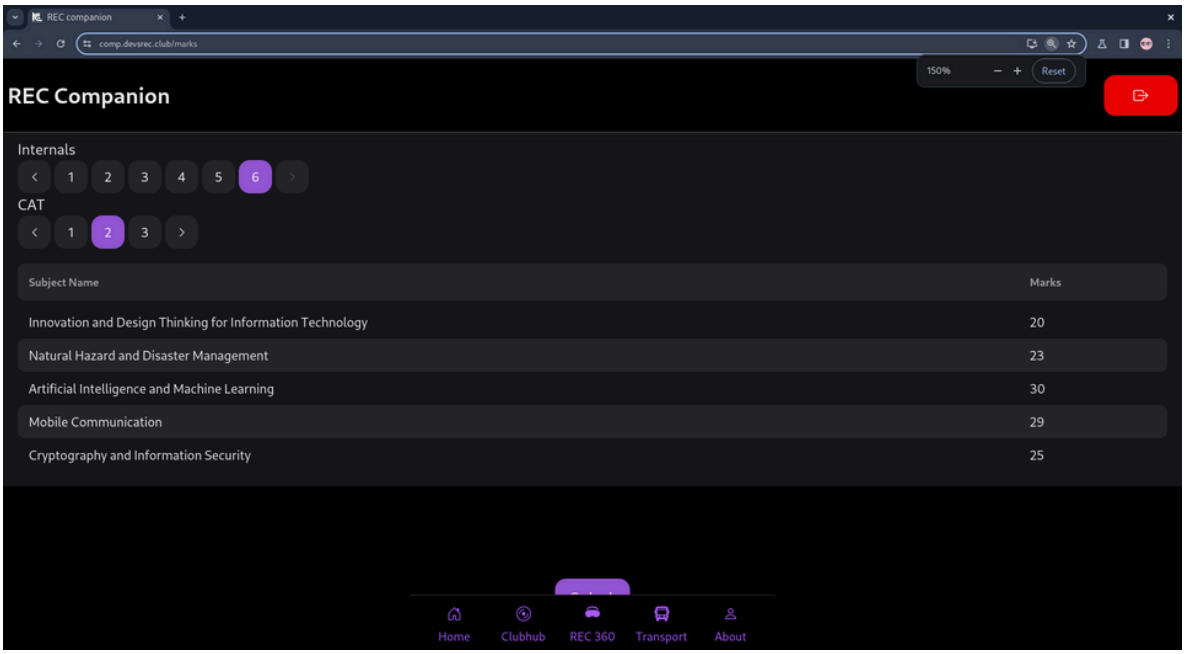

Figure 7: The user can see their internal marks seperated by each semester and each cat exam

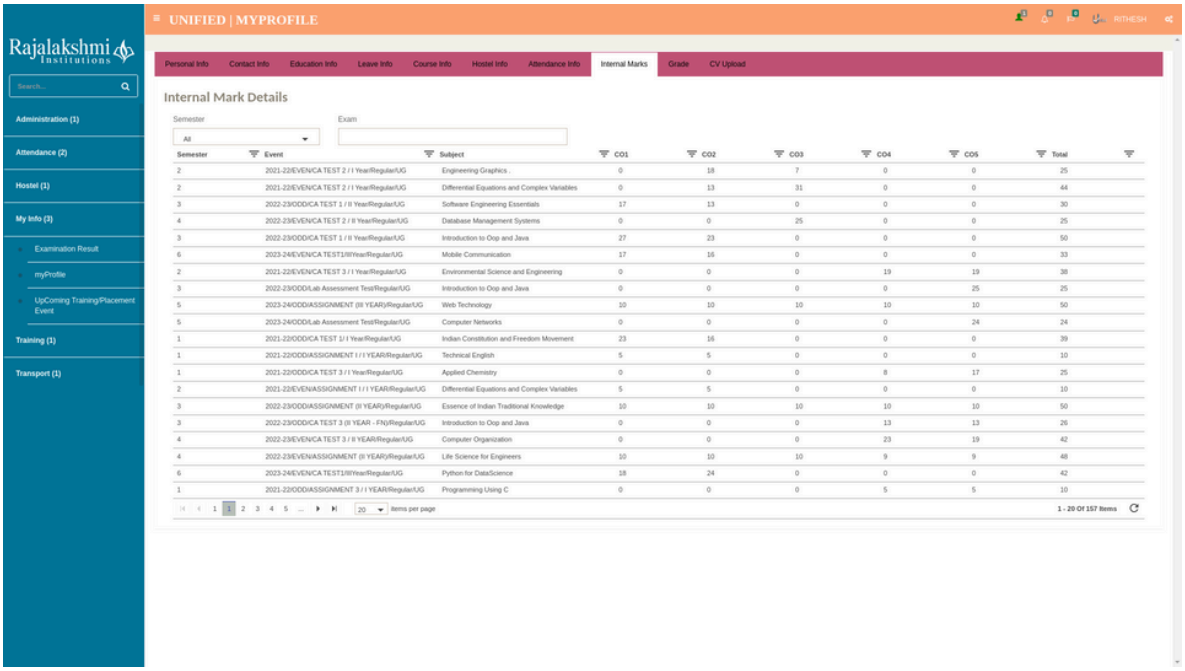

Figure 8: The previous figure can be used to compare with this image of internals marks that can be found in the unified portal

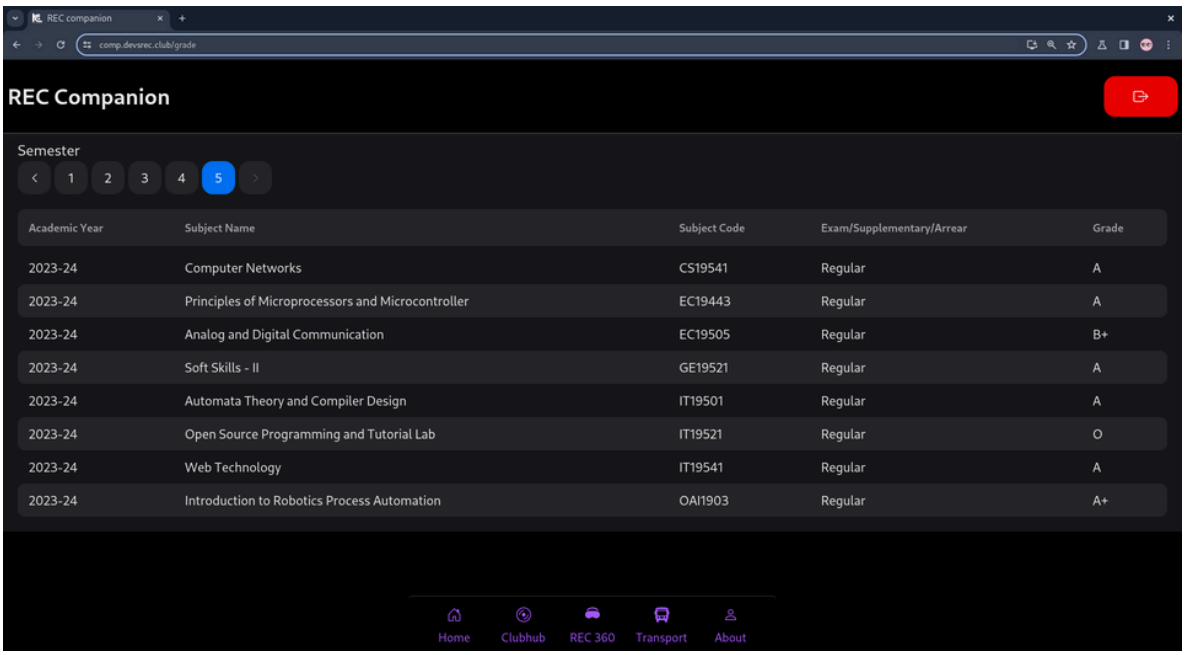

Figure 9: the user can view their semester grade

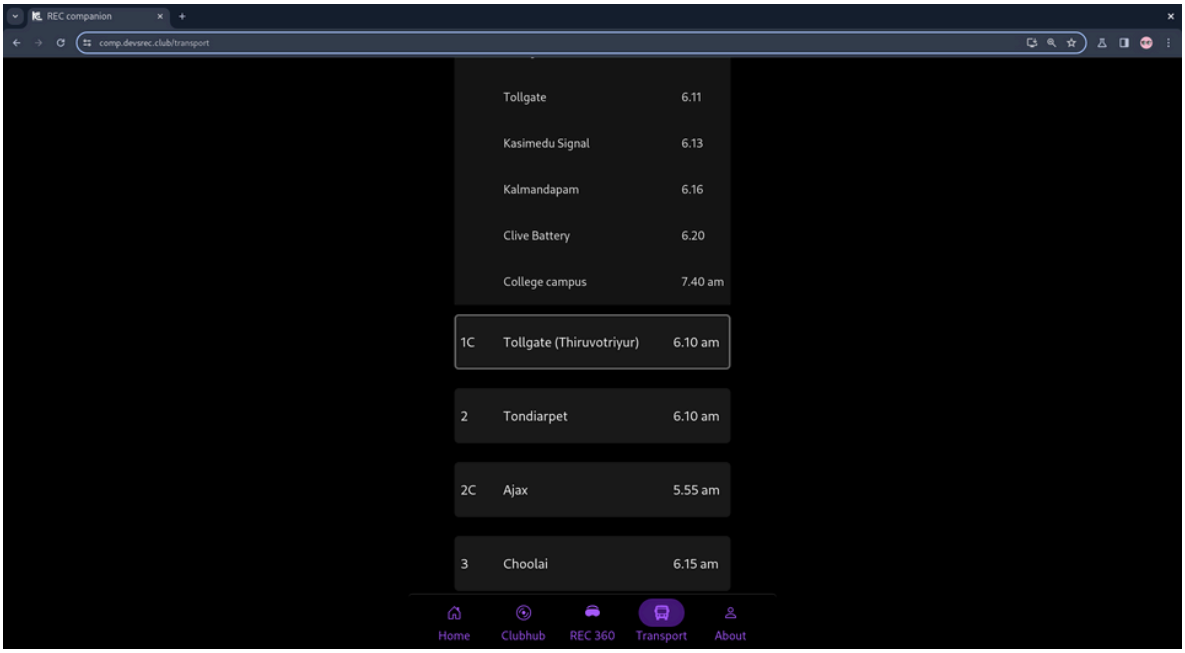

Figure 10: The user can see the bus routes and timings

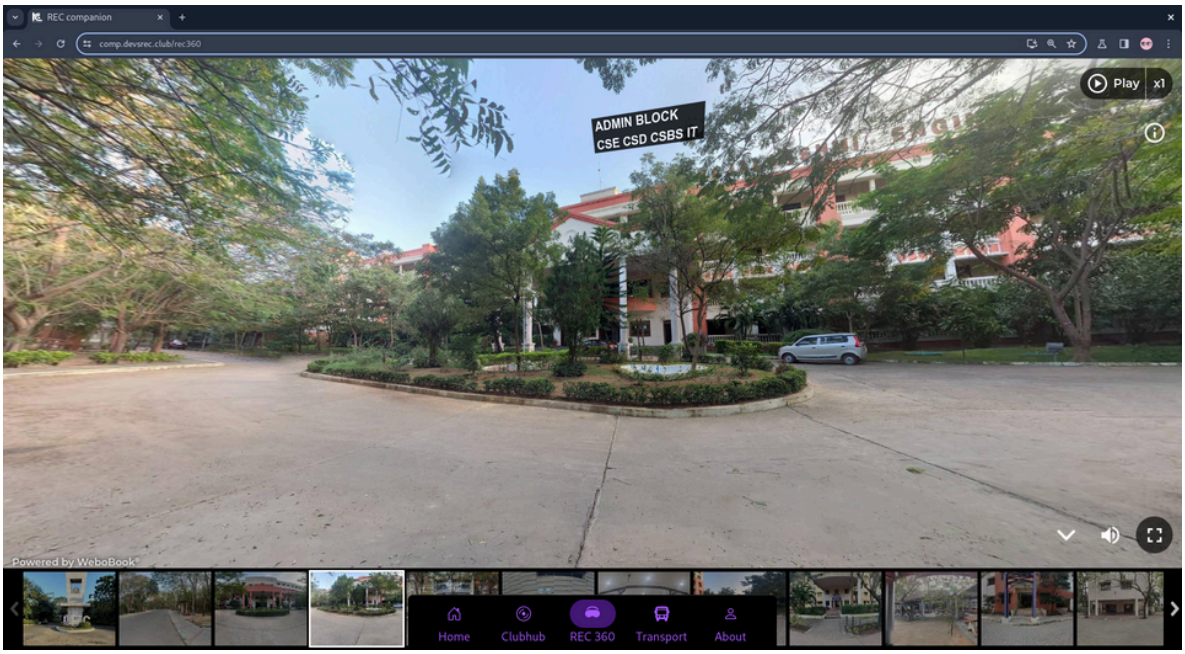

Figure 11: The user can view the 360 degree view of Rajalakshmi Engineering college

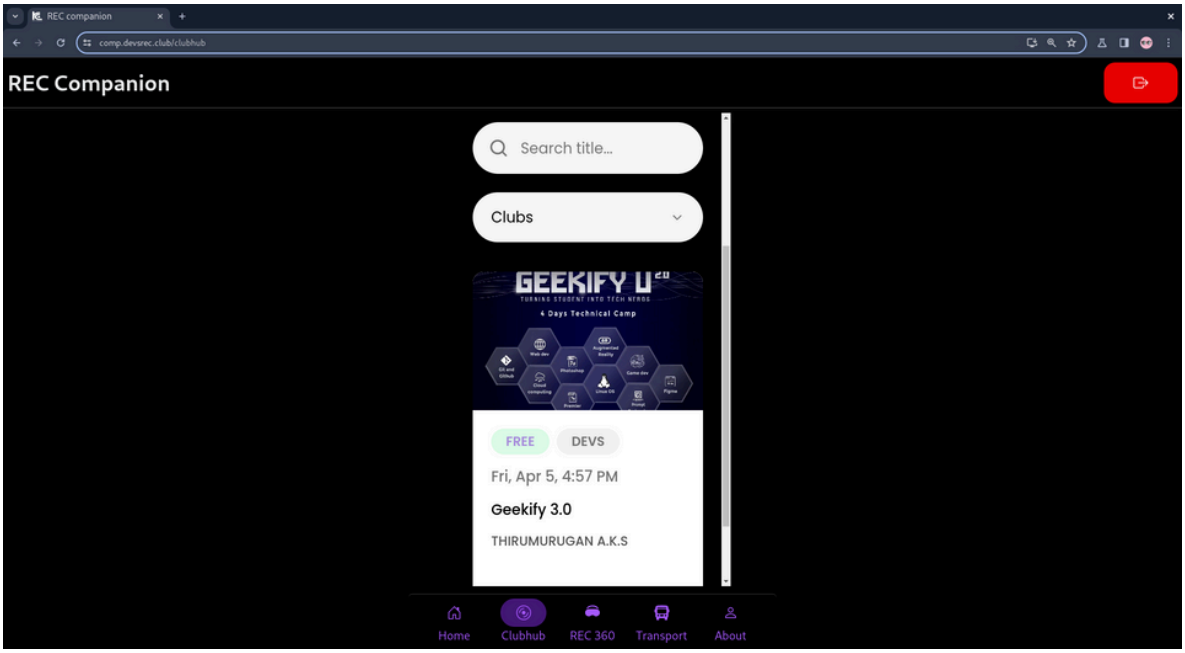

Figure 12: The user can view all the events that are happening in the various clubs of the college