

# INTERNSHIP REPORT

**B.Tech in Electronics and Communication Engineering**  
**Division of Electronics Engineering**  
**School of Engineering**  
**CUSAT**



**Roopesh O R**  
**20323085**

***Internship Details:***

Skyniche Technologies  
Plot No: 22, Kinfra Defence Park,  
Lakkidi, Ottapalam, Palakkad,  
Kerala, 679201

***Mentor Details:***

Sambhu Divakar  
Operation & Accounts executive  
sambhu@skyniche.com  
+91 89435 95561

**May 2024**

# Overview of Industry

Skyniche Technologies is a service provider specializing in custom software development, including web, desktop-, and mobile applications and SaaS solutions. The company also offers consulting services and IoT and cloud-based solutions.

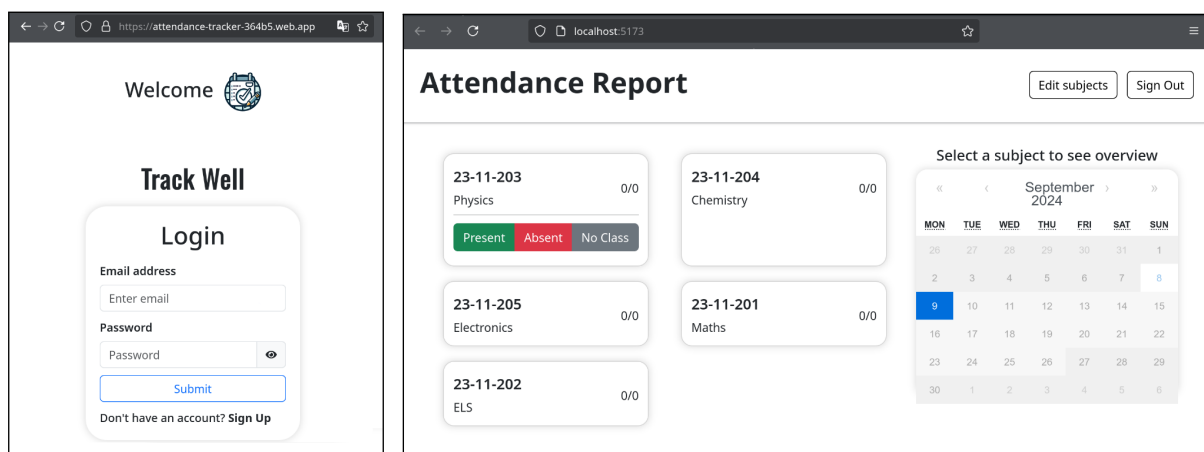
## Objective of the Study

The objective was to study MERN stack development by creating a functional web application. In practice, we focused primarily on the front side, where we utilized the **ReactJS** framework and **Bootstrap** library to develop a responsive user interface. For the backend, we used **Firebase** as the primary service.

## Details About the Study

Along with another intern, we were given an opportunity to choose an idea and develop an application around it. We proposed a *personal attendance tracker* aimed at helping college students monitor their attendance and stay informed about any potential shortages.

In the application, the user first has to create an account (to sync the data to other devices). They have to set a timetable by entering their subjects, time schedules, and the class starting and ending dates. The app checks the current time and checks if there is a class at that time according to the user's schedule. User has to check the app to mark their attendance periodically, and if no class is scheduled during that hour, an option is provided to log this accordingly. If the user has not marked attendance during the course schedule, the application will automatically mark it absent for that schedule. Once any update is marked, it is not possible to alter it.



We worked from May 7th to May 21st, following a Monday-to-Friday schedule. On the first day, we conducted research on the **React** framework and drafted a rough structure for both the application and the codebase. We then progressively added the necessary features.

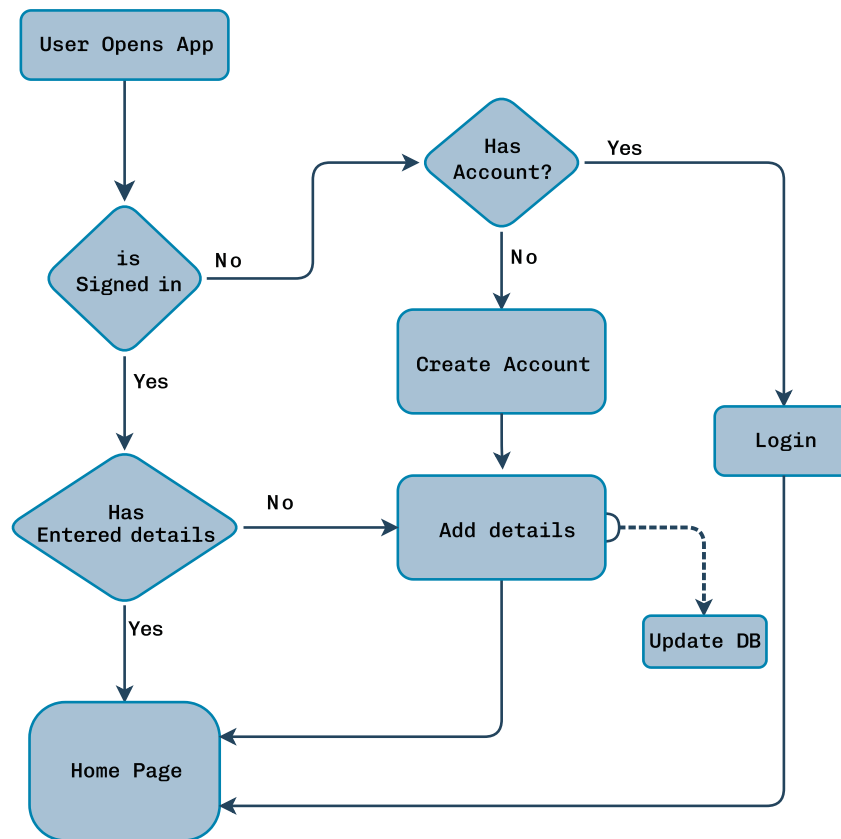


Figure 1: User flow diagram: when a user opens the app.

During the development process, we familiarized ourselves with **React**'s design principles and incorporated them into our codebase, ensuring a clean codebase and well-organized file structure throughout development. We split the UI elements into different components following **React**'s functional component style and used **React**'s state management functions to manage data passage between components.

We utilized **VSCode** with various extensions to streamline development and other developer tools to view the web page in action. For version control, we used **Git**. Since the application was not that complex, we developed the whole application in a single branch (which unexpectedly led to some merge conflicts). The codebase was hosted on **GitHub** to ensure seamless collaboration and synchronization. We deployed the web application using **Firebase Web Hosting** and shared it with our mentor to test the site.

We created an intuitive user interface with the help of **Bootstrap** and tried to add theming support that automatically detects the browser's colour scheme. Although it was a little challenging to integrate **Bootstrap** into the codebase, with some help from AI tools, we managed

to do it. We didn't follow the conventional mobile-first or desktop-first approach. Instead, the UI was designed with both mobile and desktop sizes in mind. This approach helped us to come up with a code structure that can adapt to different screen sizes and refactored (which could have become difficult when following conventional methods). We utilized **Firebase Authentication** to implement a simple email-password-based authentication system and **Firestore** to manage the database.

At the end of each day, our mentor evaluated our progress and provided feedback to guide our development. Feedback includes how the layout should be made and how the code should be organized, etc.

**Source Code:** <https://github.com/Roopesh2/attendance-tracker-cu/>  
**Application link:** <https://attendance-tracker-364b5.web.app/>

## Conclusion

Overall, we managed to develop a functional application within two weeks. The development experienced minor delays as we were familiarizing ourselves with an unfamiliar framework during the process. Through this internship, I gained experience in code refactoring, separating code into modules based on functionality. Working with **React** improved my understanding of creating scalable solutions and developing individual components of the solution in a way that allows for future extension and maintainability. It also encouraged me to think about problems at higher levels.

## Future Scope of Work

The current version of the application is relatively minimal, but it has significant potential for further enhancement. Improvements include handling edge cases such as class swaps or additional classes during weekends and evenings. Implementing a notification system to remind users of upcoming classes is another helpful feature. However, implementing reliable notifications on the web is challenging, and due to time constraints, we decided not to pursue it.

While the primary objective of the internship was to gain proficiency in MERN stack development—which was largely achieved—the most valuable takeaway was the development of engineering skills necessary to build real-world systems.