### SYSTEM ARCHITECTURE

### 6.1 INTRODUCTION:

The Attendance Management System is an Android application that is developed to facilitate attendance in a user-friendly digital platform. This is a centralized platform that allows every stakeholder of our system to register and maintain attendance securely. We have adopted the repository architecture for our system contributing to improved encapsulation of data access logic. The User interface (UI), seamlessly interacts with the API classes to process the functionality and querying the database to retrieve and display data. User actions such as marking attendance, viewing attendance, taking absence forms, and viewing absence forms trigger calls to the relevant API methods, creating a safe and secure experience. Authentication verifies user identity enhancing system security.

**6.2 USER INTERFACES:**

The user interfaces of the attendance management system, developed in Flutter, encompass various pages such as the Home Page, Login/Registration Page, List classes Page, Take attendance Page. Each page is designed to offer a user-friendly experience, facilitating seamless navigation for faculty, admin, and students.

**6.3 COMMUNICATION WITH API:**

The communication between the frontend and the backend Flask API is crucial for the system's functionality. Each UI component interacts with the API. Flutter classes play a pivotal role in this process, handling the rendering of the user interface and managing user interactions.

**6.4 API**

The Flask API serves as the core backend component to handle distinct functionalities.

**COMMUNICATION WITH DATABASE:**

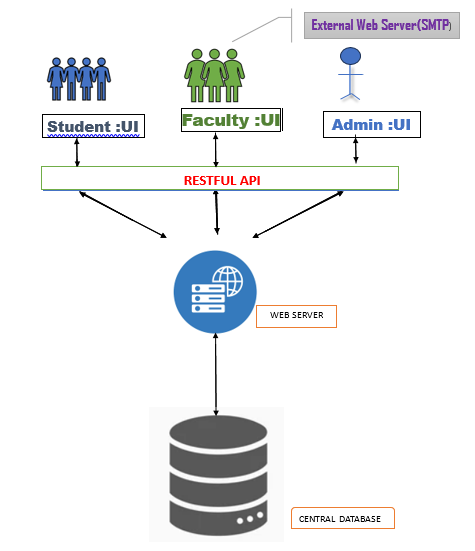
The Flask API interacts with the PostgreSQL database. This interaction involves executing queries to retrieve data from the database. The API converts incoming requests from the frontend into structured queries, ensuring seamless communication with the centralized database.

**6.5 EXTERNAL WEB SERVER(SMTP)**

**Integration with External Web server (SMTP):**

The Flask API integrates with an external Simple Mail Transfer Protocol (SMTP) server to facilitate absence intimation notifications. This external webserver is responsible for handling the communication between faculty and students. When a faculty member accepts or rejects an absence request, the Flask API triggers the SMTP server to send notifications, enhancing the communication flow within the system.

The attendance management system employs a structured architecture where each frontend UI component interacts with its relevant Flask API endpoint. The Flask backend, acting as the intermediary, communicates exclusively with the PostgreSQL database to retrieve and store data.



### Fig. 6.1 system architecture