**AI Attendance System - Team Implementation Guide** Guys split up the work !!

**Team Responsibilities & Module Breakdown**

**Member 1: Facial Recognition & Attendance Marking**

**Tasks:**

1. **Dataset Collection & Preprocessing**
   * Collect images of students.
   * Use OpenCV to capture and process facial images.
   * Apply data augmentation to enhance training data.
2. **Face Detection & Recognition**
   * Utilize OpenCV's haarcascade\_frontalface\_default.xml for face detection.
   * Train a facial recognition model using FaceNet or OpenCV’s LBPH.
   * Store facial embeddings in a database.
3. **Attendance Marking Logic**
   * Compare detected faces with stored embeddings.
   * If matched, mark attendance in the database with a timestamp.
   * Generate daily reports in CSV or Firebase.
4. **Integration with Backend**
   * Develop an API endpoint for attendance tracking.
   * Ensure secure data transfer using JWT authentication.

**Tools & Technologies:**

* Python, OpenCV, FaceNet, Firebase Realtime Database/MySQL
* Flask/Django for backend API

**Member 2: Database & Backend API Development**

**Tasks:**

1. **Setup Firebase/SQL Database**
   * Create tables for users (admin, students) and attendance logs.
   * Store facial embeddings and timestamps.
2. **Develop REST API for Attendance System**
   * Implement authentication (JWT-based login for admin).
   * API endpoints:
     + /login: Admin authentication
     + /register\_face: Save new facial embeddings
     + /mark\_attendance: Store attendance records
     + /get\_attendance: Fetch attendance logs
3. **Backend Security & Optimization**
   * Encrypt sensitive data (AES for transaction security).
   * Implement rate limiting & HTTPS for secure API calls.

**Tools & Technologies:**

* Firebase/SQL, Flask/Django, JWT, AES Encryption

**Member 3: Chatbot Integration & Admin Dashboard Features**

**Tasks:**

1. **Chatbot Integration**
   * Use Dialogflow to create a chatbot for attendance inquiries.
   * Implement API calls to fetch attendance data when queried.
   * Ensure chatbot is visible and interactive on the homepage.
2. **Admin Dashboard Features**
   * View attendance logs, student details.
   * Generate reports and export them as Excel/CSV.
   * Implement search filters to query specific students or dates.
3. **Notification System**
   * Send alerts when students miss a certain percentage of classes.
   * Email/SMS integration for attendance updates.

**Tools & Technologies:**

* Dialogflow, Firebase Functions, React, Twilio (for notifications)

**Final Integration Steps**

1. Connect frontend UI with backend API.
2. Test end-to-end functionality for attendance marking.
3. Debug and optimize system performance.
4. Conduct final deployment (Firebase Hosting/Vercel).