**Vision Based Smart Attendance System**

**Introduction:**

The **Vision-Based Smart Attendance System** is an innovative and efficient solution for managing student attendance using computer vision and database integration. This project uses eye image detection and comparison techniques to register students and mark their attendance. It eliminates manual processes and enhances reliability and accuracy in attendance tracking.

**Features:**

* **Student Registration**: Captures and stores student details with an eye image in the database.
* **Attendance Marking**: Identifies students by comparing real-time and stored eye images using SSIM.
* **Database Integration**: Maintains records of students and their attendance securely in SQLite.
* **Real-Time Image Capture**: Uses a webcam for on-the-spot eye image detection.
* **Error Handling**: Robust exception handling for smooth operation and error recovery.
* **Console-Based Interface**: Easy-to-use text-based menu for user interactions.
* **Automated Logging**: Tracks and updates attendance with timestamps automatically.

**Applications:**

* **Educational Institutions**: For tracking student attendance in classrooms or during exams.
* **Corporate Environments**: For managing employee attendance and office entry/exit logs.
* **Events**: For checking in participants at conferences, seminars, and other events.

**Name Of Team Members:**

1] Pawar Nikita Rajendra

2] Telore Maheshwari Shivaji

3] Pawar Rohan Sunil

4] Deshmukh Rohan Ravindra

**Project Phases**

1] **Requirement Gathering & Analysis**

The first phase, **Requirement Gathering & Analysis**, focuses on understanding user needs and defining both functional and non-functional requirements. This helps set a clear direction for the project. In the **System Design** phase, the system architecture, user interface, database schema, and algorithm design are developed. This phase ensures that the entire system structure is planned before implementation begins.

**2] System Design**

In the System Design phase, the system architecture, user interface, database schema, and algorithm design are developed. This phase ensures that the entire system structure is planned before implementation begins.

**3] Hardware and Software Setup**

In the Hardware and Software Setup phase, the necessary equipment (such as cameras) is configured, and software libraries for image processing and recognition are installed. This sets up the foundation for the core system development.

**4] Eye Detection and Recognition Model Development**

In the Eye Detection and Recognition Model Development phase, the system’s core features are built, including eye detection using models like Haar Cascades or MTCNN and eye recognition through deep learning techniques, ensuring the system can reliably detect and identify individuals.

**5] Integration and Development of Attendance Management**

The Integration and Development of Attendance Management System phase follows, where the eye recognition system is integrated with the attendance logging mechanism, and the user interface is created for admin use.

**6] Testing**

Testing follows, where the system undergoes rigorous testing—unit testing, integration testing, load testing, and user acceptance testing—to ensure reliability, performance, and usability.

**Requirements**

**1] Hardware Requirments**

* **Camera:** High-resolution camera for real-time image capture.
* **System**: A computer or server with adequate processing power (e.g., Intel i5 or higher) to run image processing and recognition algorithms.
* **Storage**: Minimum 500 GB for storing attendance logs and image.

**2] Software Requirments**

**Software Requirements**

* Operating System: Windows, macOS, or Linux (Ubuntu preferred).
* Programming Language: Python 3.8 or higher.
* Libraries and Frameworks:
  + OpenCV: For image processing.
  + Dlib: For Eye detection and feature extraction.
  + NumPy: For numerical computations.
  + Scikit-image: For advanced image preprocessing and analysis
* Database:
  + SQLite3 for storing attendance logs and user data

**Steps to Run the Program**

**1. Clone the Repository**

* Download or clone the project repository:

git clone < [https://github.com/nikitapawar5121/Vision-Based-Smart- Attendance-System.git](https://github.com/nikitapawar5121/Vision-Based-Smart-%20%20%20%20%20Attendance-System.git)>

cd < [https://github.com/nikitapawar5121/Vision-Based-Smart- Attendance-System.git](https://github.com/nikitapawar5121/Vision-Based-Smart-%20Attendance-System.git)>

**2. Install Required Libraries**

* Ensure Python is installed on your system, then install the required libraries:

pip install -r requirements.txt

**3. Configure the Database**

* Set up the SQLite3 database:
  + Ensure the attendance.db file exists or create it.
  + Use the following SQL commands to initialize the tables (if needed):

CREATE TABLE users (id INTEGER PRIMARY KEY, name TEXT, eye\_encoding BLOB);

CREATE TABLE attendance (id INTEGER PRIMARY KEY, user\_id INTEGER, timestamp TEXT);

**4. Add EYE Data**

* Run the script to register users by capturing their eye data:

python add\_user.py

**5. Run the Program**

* Start the attendance system:

python main.py

**6. View Attendance Records**

* Use the admin panel (if available) or directly query the database:

sqlite3 attendance.db

SELECT \* FROM attendance

**7. Stop the Program**

* Press Ctrl + C in the terminal to stop the system.