Name	Shorya Dixit
Email ID	2021uee0155@iitjammu.ac.in
Phone Number	7838076433
GitHub ID	[Shorya-Dixit (github.com)](github.com)
Discord ID	1098949519015673856
Current occupation	Student
Education Details	IIT Jammu B.Tech Electrical Engineering
Technical skills with level	Python, C++, HTML, Javascript, CSS, Tensorflow, Keras, Flask, PyTorch, OpenCV, Scikit-learn, NumPy, Pandas, Google Firebase: - **Intermediate** NLTK, ReactJS, Docker, MatPlotlib, MySQL, PL/SQL, VHDL:- **Novice**

Title:- Smart Facial Attendance System

Summary:-

- **Research on Open-Source Models:** I will leverage my experience with dlib for facial recognition and explore other open-source libraries like DeepFace and InsightFace, that work effectively in both online and offline settings.

- **Benchmarking and Dataset Preparation:** To ensure an objective evaluation, I will prepare a benchmark dataset of images under various conditions (lighting, poses) to ensure model robustness. This dataset will be used to assess the performance of shortlisted models based on predefined criteria, including recognition accuracy, processing speed, and memory requirements.

- **User Registration Pipeline:** A user registration pipeline will be developed to capture and store facial data (images) and facial landmarks (key points on the face) of teachers. This pipeline will ensure secure and efficient data collection. Facial images and extracted landmarks will be securely stored in a database with access controls. Encryption can be employed for additional security.

- **Face Lookup Algorithm:** An efficient face lookup algorithm will be implemented to compare live captured faces with the stored data. This algorithm will output confidence scores indicating the likelihood of a match.

- **Demo Application:** A user-friendly demo application will be built to showcase the system's functionalities. This application will facilitate both user registration and user lookup, with geo-tagging.

Project Details:

1. Project Overview:

Understanding of the project:

This project aims to develop a facial recognition attendance system for teachers in government schools. It will offer the following functionalities:

- **User Registration:** Teachers will be able to register themselves by capturing their facial image for recognition.
- **Facial Recognition Attendance Marking:** The system will utilize Face recognition models from libraries like DeepFace or dlib to recognize teachers' faces and mark their attendance.
- **Geo-Tagging:** As a potential security measure, the system will capture the user's location data during attendance marking to verify their physical presence at the school (requires additional user consent and privacy considerations).
- **Offline Functionality:** The system must be able to operate in both online and offline environments, increasing flexibility for various school settings.

Possible Problems:

1. Facial Recognition Accuracy:

- Pre-trained models might not achieve perfect accuracy, especially under varying lighting conditions, poses, or facial occlusions (e.g., glasses).
- Teachers with similar facial features could lead to misidentification.

2. Data Security and Privacy:

- Storing sensitive facial data requires robust security measures to prevent unauthorized access or breaches.
- User consent and data privacy regulations need to be addressed regarding data collection, storage, and usage.

3. System Performance:

- Facial recognition algorithms can be computationally expensive, potentially impacting performance on resource-constrained devices.
- Network connectivity issues might hinder system functionality in offline environments (if implemented).

4. User Acceptance:

- Teachers might be hesitant to use a facial recognition system due to privacy concerns or lack of technical familiarity.

Solutions:

1. Model Selection and Training:

- Evaluate different pre-trained models to find the one offering the best balance between accuracy and resource efficiency.
- Consider fine-tuning a pre-trained model on a dataset of teachers' faces from your specific school environment to improve recognition accuracy.
- Implement techniques like facial landmark detection to enhance recognition robustness.

2. Data Security and Privacy:

- Utilize client-side encryption to encrypt facial data before storing it on the server.
- Implement access controls to restrict unauthorized access to data.
- Obtain informed consent from teachers regarding data collection and usage.
- Adhere to relevant data privacy regulations.

3. System Optimization:

- Choose libraries and frameworks optimized for performance on the target devices.
- Consider implementing lightweight versions of facial recognition models if necessary.
- Explore offline processing techniques for attendance marking if internet connectivity is unreliable.

4. User Training and Communication:

- Clearly communicate the benefits and security measures of the system to teachers.
- Offer training sessions to familiarize teachers with the registration and attendance marking process.

2. Implementation Details with timelines:

Milestone 1: Investigation (Required - 2 weeks)

- Research on open-source facial recognition models.
- Define evaluation criteria (accuracy, speed, memory footprint).

Deliverables:

- Project report outlining model research and evaluation plan.

Milestone 2: Benchmarking and Dataset Preparation (Required - 3 weeks)

- Prepare a benchmark dataset (if chosen) or explore publicly available options.
- Implement data preprocessing techniques (cleaning, alignment).

Deliverables:

- Prepared benchmark dataset or justification for utilizing a public dataset.
- Code for data preprocessing.

Milestone 3: User Registration Pipeline (Required - 2 weeks)

- Design and implement the user registration process.
- Establish secure storage for facial data and landmarks.

Deliverables:

- Functional user registration pipeline with secure data storage.
- Code for facial landmark detection and data storage.

Milestone 4: Face Lookup Algorithm (Required - 2 weeks)

- Choose the optimal facial recognition model based on benchmarking results.
- Implement the face lookup algorithm with confidence score output.

Deliverables:

- Integrated face lookup algorithm using the chosen model.
- Code for face recognition and confidence score generation.

Milestone 5: Demo Application (Required - 2 weeks)

- Develop a user-friendly application for registration and user lookup functionalities.
- Implement optional geo-tagging feature (requires additional planning).

Deliverables:

- Functional demo application showcasing registration and attendance marking.
- Code for the demo application (including optional geo-tagging if implemented).

<u>Milestone</u> 6: Documentation (Required - 1 week)

- Prepare comprehensive project documentation.
- Include user manuals, code comments, and deployment instructions.

Deliverables:

- Project documentation for future reference and potential deployment.

Availability:-

Number of hours available to dedicate to this project per week	25 hours
Do you have any other engagements during this period ? (projects/internships)	Yes

I am starting a remote internship at DRDO from 21st May,2024. (For 6 weeks)

Personal Information:-

About Me: 3rd year student at IIT Jammu, currently enthralled by the potential of Machine Learning. I'm a perpetual learner, always seeking to expand my knowledge and tackle novel problems.

What is your motivation to apply for this project?

Developing a facial recognition attendance system previously ignited my passion for applying Machine Learning's power to real-world challenges, particularly in facial recognition. This project presents a captivating opportunity to delve deeper into the intricacies of these algorithms and tailor them for optimal performance in a practical setting.

Previous experience/open source projects:-

Project Name	Project Description	Links (if any)
Face Attendance System	Tracks student attendance using facial recognition. Integrates Firebase for real-time updates and data storage. Offers instant verification and detailed student profiles.	Shorya-Dixit/Face_Attendance_System (github.com)
Sentiment Analysis	Developed sentiment analysis model. Deployed model as web app for real-time sentiment analysis.	Shorya-Dixit/Sentiment-Analysis (github.com)

Contribution in C4GT's open community:-

Have you contributed to tickets in C4GT's open community? **(Mandatory to answer)**	No
Have you successfully completed C4GT's GitHub Classroom Assignment? **(Mandatory to answer)**	No
Enter your DPG points **(Mandatory to answer, Enter 0 if not applicable)**	0
Screenshot of leaderboard with your GitHub ID **(Mandatory to answer, enter 0 if not applicable)**	0