CSAI 498 / CSAI 499

Final Report – Fall

Smart Attendance System

• **Team Members:**

Student Name	ID	Program
Hania Khaled	202202227	DSAI
Farah Samir	202201410	DSAI
Yassmin Raafat	202201706	SW
Hager Ali	202202011	DSAI

• **Supervisor:**

Dr. Mohamed Ghalwash

• **Semester / Year:**

Fall 2025

• **Team number:**

32

• **Submission Date:**

December 13, 2025

• **Department & School:**

Computer Science and Artificial Intelligence (DSAI/SW)

Table of contents:

Abstract:

Traditional attendance systems in educational institutions rely heavily on manual processes that are time-consuming, error-prone, and incapable of accurately reflecting students' actual presence duration. Such methods often disrupt lecture time and lead to inaccurate or unfair attendance records.

This project presents a **Smart Attendance System** that automates classroom attendance using computer vision and artificial intelligence techniques. Instead of processing continuous video streams, the system captures a single image at fixed time intervals, detects and recognizes student faces, and computes attendance based on the total duration a student appears across captured frames. The system integrates a two-stage face detection pipeline, a deep learning–based face recognition model using ResNet embeddings, and a centralized database for attendance storage. A web-based dashboard enables instructors to monitor attendance statistics in real time.

The proposed system aims to reduce administrative effort, improve accuracy, and provide a scalable and efficient alternative to traditional attendance methods while offering more meaningful insights into student participation.

1. Introduction

1.1 Problem Statement

Manual attendance recording remains widely used in classrooms despite its inefficiency and limitations. Traditional methods such as paper-based attendance sheets or basic digital logs consume valuable lecture time and are prone to human error, proxy attendance, and data inconsistency. In many cases, attendance is recorded as a binary value (present or absent), which fails to represent how long a student actually remains in class.

With increasing class sizes, instructors face significant challenges in maintaining accurate attendance records without interrupting the teaching process. These limitations highlight the need for an automated attendance system that is accurate, scalable, and capable of reflecting students' actual presence duration with minimal instructor intervention.

1.2 Motivation

Accurate attendance tracking is essential for academic evaluation, student engagement analysis, and institutional reporting. Instructors benefit from automation by reducing administrative workload and preserving teaching time. Students benefit from fairer attendance evaluation based on actual classroom presence rather than a single roll-call moment. Institutions benefit from reliable attendance data that supports academic monitoring and decision-making.

Advances in computer vision and deep learning make it possible to design an intelligent, non-intrusive attendance system that operates efficiently in real classroom environments.

1.3 Proposed Solution Overview

The proposed solution is a Smart Attendance System that automatically captures classroom images at fixed intervals, detects and recognizes student faces, and calculates attendance duration accordingly. Attendance records are stored in a centralized database and visualized through a web-based dashboard accessible to instructors. The system is modular, scalable, and designed to integrate artificial intelligence with a user-friendly software interface.

2. Literature Review and Market Survey

2.1 Related Academic / Technical Work

Several academic studies have explored the use of face recognition systems for automated attendance tracking due to their non-intrusive nature and high accuracy. One study proposed a convolutional neural network–based attendance system that utilized facial embeddings to identify students in classroom environments, achieving high recognition accuracy under controlled lighting conditions. Another research work employed deep residual networks (ResNet) for face recognition, demonstrating robustness against variations in facial pose, expression, and illumination.

Additional studies investigated hybrid face detection approaches combining traditional methods, such as Haar Cascades, with deep learning–based detectors to improve detection reliability and reduce missed detections in real-time applications. These works highlight the effectiveness of deep learning models in biometric identification while also emphasizing challenges related to computational cost, scalability, and real-world deployment.

2.2 Existing Systems and Market Solutions

Several attendance systems are currently used in academic institutions. **Mobile-based face attendance applications** allow students to check in by capturing a selfie using a mobile application; however, these systems depend heavily on student cooperation and supervision. **QR code–based attendance systems** require students to scan a code displayed in the classroom, introducing dependency on mobile devices and internet connectivity.

Additionally, **AI camera–based attendance systems** utilize continuous video streams to detect and recognize faces in real time. While these systems offer automation, they often involve high hardware and computational costs, raise privacy concerns due to continuous monitoring, and provide limited flexibility for customization.

2.3 Comparative Analysis

Existing attendance systems suffer from various limitations, including high hardware costs, privacy concerns due to continuous monitoring, high computational requirements, and limited flexibility. Manual systems lack automation and accuracy, while many AI-based commercial solutions operate as closed systems with limited customization.

2.4 Identified Gap

Despite the availability of camera-based and AI-driven attendance systems, several important gaps remain. Most existing solutions rely on continuous real-time video processing or proprietary attendance cameras that focus only on marking students as present or absent at a single point in time. This results in high computational cost, increased privacy concerns, and limited insight into actual student presence during the lecture.

In addition, many commercial systems operate as closed, black-box solutions with minimal flexibility, making it difficult to adapt attendance logic, integrate with academic dashboards, or customize the system for specific university requirements. Cloud-based solutions further introduce dependency on internet connectivity, recurring costs, and potential data privacy issues.

The proposed Smart Attendance System addresses these gaps by using periodic image capture instead of continuous video streaming, significantly reducing computational overhead and improving privacy. Unlike existing systems, the proposed solution measures attendance based on presence duration across multiple captured frames, providing a more accurate representation of student participation. Furthermore, its modular and customizable architecture enables seamless integration with academic databases and instructor dashboards, making it more suitable for real classroom environments.

3. Requirements Analysis

3.1 Functional Requirements

FR-1: The system shall automatically capture classroom images at fixed time intervals.
FR-2: The system shall detect all student faces within each captured image.
FR-3: The system shall recognize detected faces and identify students by name and ID.
FR-4: The system shall calculate attendance based on the duration of student presence across multiple images.
FR-5: The system shall store attendance records securely in a centralized database.
FR-6: The system shall provide a dashboard for instructors and students to view attendance statistics.
FR-7: The system shall authenticate users and restrict access to authorized personnel.

3.2 Non-Functional Requirements

Performance: Face detection and recognition should operate with minimal processing delay.
Security: Attendance data must be protected through authentication and access control mechanisms.

Scalability: The system should support increasing numbers of students and classes.

Usability: The dashboard should be intuitive and easy to navigate for instructors.

Reliability: The system should function consistently under varying classroom conditions.

4. System Design

4.1 Overall System Architecture

The Smart Attendance System follows a modular architecture consisting of the following components:

1. Image Capture Module
2. Face Detection Module
3. Preprocessing and Cropping Module
4. Face Recognition Module
5. Attendance Processing and Database Module
6. Web/Application Dashboard

This architecture ensures clear separation of concerns and allows independent development and testing of each module.

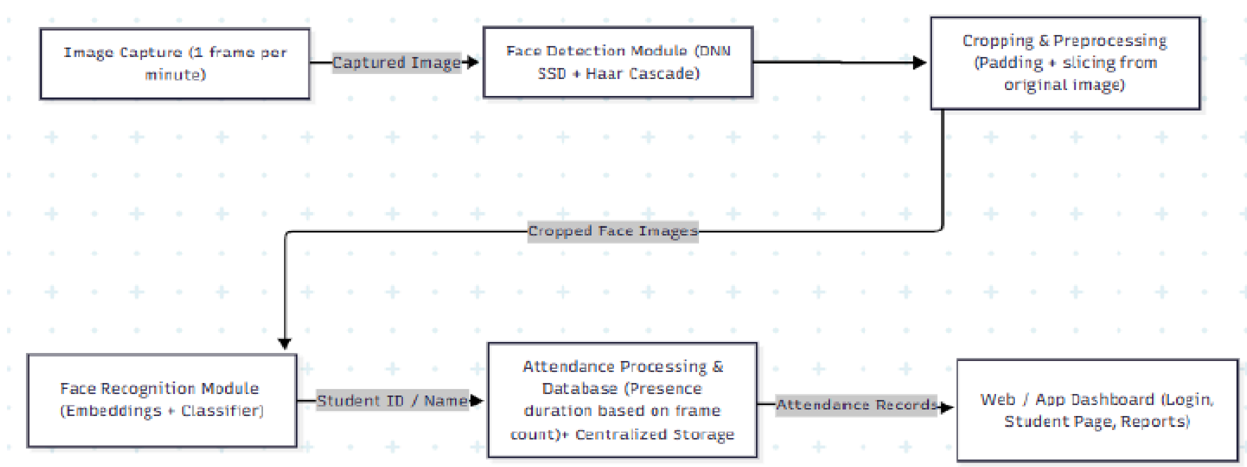

Figure 1: *System Architecture Diagram*

4.2 Component Breakdown

The Smart Attendance System is composed of multiple modular components, each responsible for a specific function within the system pipeline.

- **Image Capture Module:**
  This component is responsible for periodically capturing classroom images at fixed time intervals (once per minute). It serves as the system's input source while avoiding continuous video recording to reduce computational cost and privacy risks.
- **Face Detection Module:**
  The face detection module identifies student faces within each captured image. It uses a deep learning–based detector as the primary method, with a traditional detector as a fallback to ensure robustness under different lighting and positioning conditions.
- **Preprocessing and Cropping Module:**
  This component processes detected faces by applying padding, cropping, resizing, and normalization. The goal is to generate clean and standardized face images suitable for accurate recognition.
- **Face Recognition Module:**
  The face recognition module extracts facial embeddings and matches them against stored student identities to determine each student's name and ID.
- **Attendance Processing Module:**
  This component aggregates recognition results across multiple images and calculates attendance based on presence duration rather than a single detection event.
- **Centralized Database:**
  The database stores student profiles, attendance records, and system metadata in a structured and secure manner.
- **Web/Application Dashboard:**
  The dashboard provides instructors with secure access to attendance summaries,and individual student records
- **Authentication and Authorization Module:**
  This component ensures secure access to the system by validating user credentials and enforcing role-based permissions.

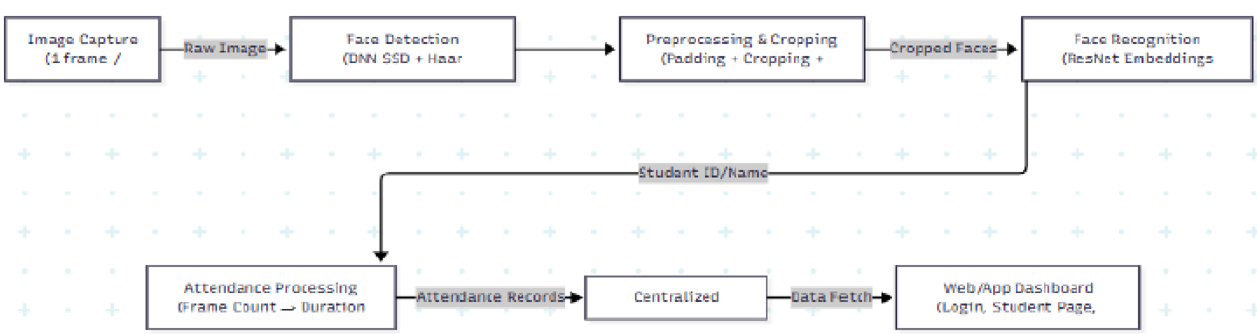

Figure 3: *System Data Flow Diagram*

4.3 Design Decisions and Rationale

Several key design decisions were made to ensure system efficiency, scalability, and suitability for academic environments.

- **Periodic Image Capture vs. Continuous Video:**
  Periodic image capture was chosen to significantly reduce computational overhead and storage requirements while addressing privacy concerns associated with continuous monitoring.
- **Modular Architecture:**
  A modular design allows independent development, testing, and future replacement of system components without affecting the entire system.
- **Two-Stage Face Detection:**
  Combining a deep learning–based detector with a fallback traditional method improves robustness and reduces face detection failure cases.
- **Presence Duration–Based Attendance:**
  Attendance is calculated based on how long a student appears across multiple frames, providing a more accurate and fair representation of participation.
- **Web-Based Dashboard:**
  A web-based interface was selected for accessibility, ease of deployment, and platform independence.

The final system architecture and component design were refined based on feedback received from the project supervisor. The feedback emphasized improving modularity, ensuring clear separation between artificial intelligence components and software modules, and enhancing scalability to support future system extensions. These recommendations were incorporated into the final design to improve system clarity, maintainability, and academic alignment.

4.4 Program-Specific SO(6) Focus

4.4.A SWD – Software Development

- **User Interface (UI) / Dashboard Module:**
  Provides a web-based interface that allows instructors to log in, view attendance summaries, access individual student records, and analyze class-level attendance statistics.
- **Backend Application Module:**
  Handles business logic, attendance calculations, API endpoints, and communication between the AI modules, database, and UI.
- **AI Integration Interface:**
  Receives recognition results (student ID and timestamps) from the face recognition module and forwards them to the attendance processing logic.
- **Attendance Processing Module:**
  Aggregates recognition results across multiple captured images and computes attendance based on presence duration rather than a single detection.

- **Database Module:**
 Stores structured data including student profiles, class sessions, attendance records, and user authentication information.
- **Authentication and Authorization Module:**
 Manages secure user login, session handling, and role-based access control for instructors and administrators.

4.4.B DSAI – Data Science & AI

- **Data Lifecycle:**
Data collection begins with capturing student images. Preprocessing includes face detection, cropping, resizing, and normalization. Modeling is performed using a deep learning–based face recognition model. Evaluation is conducted using recognition accuracy and validation testing on unseen images.
- **ML / AI Models and Justification:**
A CNN-based model (ResNet architecture) was selected due to its proven performance in face recognition tasks and ability to generate discriminative facial embeddings.
- **Feature Engineering and Evaluation Metrics:**
Facial embeddings serve as the primary features. Model performance is evaluated using recognition accuracy and identity matching correctness.
- **Model Integration into the System:**
The trained model is integrated into the backend pipeline, where it processes cropped face images and outputs identity predictions used by the attendance processing module.

5. Project Timeline (Next Semester)

• Phase 1: System Implementation (Weeks 1–6, Spring Semester)

Objective:
 Complete the core functional modules of the Smart Attendance System and finalize model training.

Key Activities:

- Finalize face recognition model training using the prepared dataset.

- Optimize model accuracy through hyperparameter tuning and data augmentation.
- Implement the Attendance Processing Module to compute presence duration.
- Set up the centralized database schema for students and attendance records.
- Complete backend logic linking face recognition output to attendance records.

Milestones:

- Trained and validated face recognition model.
- Fully implemented attendance computation logic.
- Database schema finalized and tested.

Deliverables:

- Trained ResNet-based face recognition model.
- Attendance Processing Module.
- Database tables for students and attendance logs.

Phase 2: System Integration (Weeks 3–8, Spring Semester)

Objective:
Integrate all system components into a single end-to-end pipeline.

Key Activities:

- Integrate face detection, preprocessing, recognition, and attendance logic.
- Connect backend modules with the centralized database.
- Link backend services to the web dashboard.
- Implement user authentication and access control.

Milestones:

- End-to-end pipeline operational.
- Backend successfully communicating with frontend dashboard.

Deliverables:

- Integrated system pipeline.
- Secure login and role-based access functionality.
- Fully connected dashboard with live attendance data.

Phase 3: Testing and Validation (Weeks 6–12, Spring Semester)

Objective:
Ensure system reliability, accuracy, and robustness under real classroom conditions.

Key Activities:

- Unit testing for individual modules (detection, recognition, attendance logic).

- End-to-end testing using real classroom images.
- Accuracy evaluation of face recognition results.
- Performance testing under different lighting and classroom scenarios.

- Bug fixing and system optimization.

Milestones:

- Verified system accuracy and stability.
- All critical bugs resolved.

Deliverables:

- Test reports and evaluation metrics.
- Optimized and stable system version.

Phase 4: Deployment and Environment Setup (Weeks 9–13, Spring Semester)

Objective:
Prepare the system for real-world usage and demonstration.

Key Activities:

- Configure deployment environment (local or web-based).
- Deploy backend services and database.
- Deploy dashboard interface.
- Validate system accessibility and performance post-deployment.

Milestones:

- System successfully deployed.
- Stable access for instructors and administrators.

Deliverables:

- Deployed Smart Attendance System.
- Deployment documentation.

Phase 5: Documentation and Final Reporting (Weeks 4–13, Spring Semester)

Objective:
Produce comprehensive documentation and final project deliverables.

Key Activities:

- Write final graduation project report.

- Document system architecture, design decisions, and implementation.
- Prepare user guide for instructors.
- Update GitHub repository with final code and README.

- Prepare final presentation and demo materials.

Milestones:

- Final report completed.
- Presentation ready.

Deliverables:

- Graduation Project Final Report.
- Updated GitHub repository.
- Presentation slides and demo.

5.2 Deliverables per Phase

Phase	Deliverables
Implementation	Trained model, attendance module, database schema
Integration	Fully integrated pipeline, dashboard connectivity
Testing	Evaluation results, optimized system
Deployment	Deployed system, environment setup
Documentation	Final report, GitHub repo, presentation

Responsibilities are distributed to ensure balanced workload and efficient progress:

- **Hania Khaled:** Face detection pipeline, face recognition model training, system integration.
- **Farah Samir:** Image preprocessing, model training and evaluation, attendance processing logic.
- **Hager Ali:** Dataset labeling, model optimization, testing and validation.
- **Yassmin Raafat:** UI development, backend–frontend integration, database setup, deployment.

All members will contribute to testing, documentation, and final reporting.

Figure 3 illustrates the Gantt chart for the Smart Attendance System, showing task distribution, timelines, and project milestones planned for the spring semester.

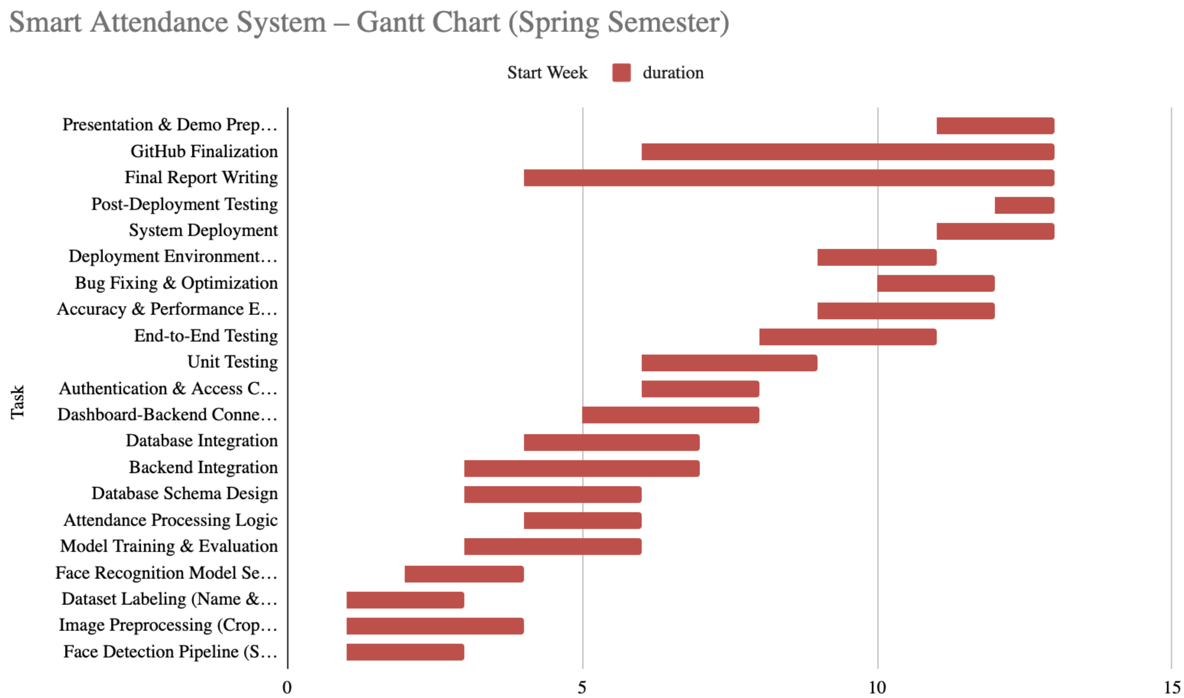

Figure 3: *Smart Attendance System – Gantt Chart (Spring Semester)*

6. Challenges and Solutions

6.1 Technical Challenges

The primary technical challenge encountered in the current phase of the project is data gathering for face recognition training. Collecting a sufficient number of high-quality student images under varying lighting conditions, angles, and facial expressions is essential to ensure reliable recognition performance. Inconsistent image quality, occlusions, and limited samples per student can negatively affect model accuracy and system reliability.

Another challenge related to data collection is data organization and labeling. Ensuring that all collected images are correctly labeled with the appropriate student name and ID is critical to avoid recognition errors during system operation.

6.2 Organizational / Team Challenges

A major organizational challenge was coordinating data collection across multiple students and classes. Not all participants were available at the same time, which caused delays in completing the dataset. Additionally, aligning schedules between team members from different majors required careful coordination.

6.3 Mitigation Strategy

To address the data gathering challenge, a structured data collection process was defined. Students were instructed to provide images under different lighting conditions and poses to increase dataset diversity. Clear naming conventions and folder structures were enforced to ensure consistent labeling and data organization.

To mitigate organizational challenges, tasks were distributed among team members, and progress was tracked regularly to ensure data collection milestones were met. Additional data will be collected in future phases if needed to further improve model robustness.

7. Work Summary (Fall Semester)

7.1 Research completed

During the fall semester, the team conducted extensive research on existing automated attendance systems and relevant technologies. This included a literature review of face recognition models ResNet, face detection methods, and attendance calculation strategies. Research also covered competitive market solutions, such as AI-based attendance cameras and cloud services, to identify system gaps and justify our design choices.

7.2 Design artifacts produced

The following design artifacts were produced during the fall semester:

- Final **system architecture diagram** defining the major system modules and data flow.
- **Component breakdown** specifying responsibilities of each system component.
- Initial **UI and dashboard layout designs**.
- **Project timeline and Gantt chart** for planning the upcoming semester.
- A **trained face recognition model prototype** based on a pre-trained ResNet architecture.

The model was fine-tuned using a custom dataset created by collecting full student images, applying face cropping using Haar Cascade detection, and using the cropped face images for training. Training accuracy was evaluated to validate model performance.

7.3 Implemented or prototyped features

- **Dataset Collection and Preparation:**
  Full student images were collected and organized. Face regions were extracted from these images using Haar Cascade detection to generate cropped face samples suitable for training.

- **Face Detection and Preprocessing Pipeline:**
  A preprocessing pipeline was implemented to detect faces, apply cropping and normalization, and prepare clean inputs for the recognition model.
- **Face Recognition Model Prototype:**
  A face recognition prototype was developed using a pre-trained ResNet architecture. The model was fine-tuned on the prepared dataset, and training accuracy was evaluated to validate model performance.
- **End-to-End AI Pipeline Testing:**
  Initial testing was conducted by passing captured images through detection, preprocessing, and recognition stages to verify correct data flow and identity prediction.
- **UI and Dashboard Prototype:**
  A preliminary web-based dashboard was designed to demonstrate login screens and basic attendance visualization concepts.
- **Project Repository and Version Control:**
  The project repository was set up, and version control was used to track development progress and maintain collaboration

7.4 Individual team member contributions

- **Yassmin Raafat (Software Development):**
  Responsible for the software development aspects of the project. Contributed to the overall system architecture design, development of the web-based user interface and dashboard prototypes, and definition of backend–frontend integration concepts. Worked on designing the database structure and outlining API interactions between system modules. Also participated in project planning, documentation, and coordination between software and AI components.

- **Hania Khaled (Data Science & AI):**
  Focused on data collection and preparation for the face recognition system. Led the implementation of the face detection pipeline, including applying detection techniques to extract face regions from raw images. Contributed to building and organizing the dataset used for training the face recognition model and participated in system integration activities.

- **Farah Samir (Data Science & AI):**
  Worked on training and fine-tuning the face recognition model using a pre-trained ResNet

architecture. Contributed to preprocessing pipeline validation, model evaluation, and accuracy assessment. Participated in developing the attendance processing logic and validating model outputs during end-to-end pipeline testing.

- **Hager Ali (Data Science & AI):**
  Assisted in dataset labeling, data organization, and preprocessing verification. Contributed to documentation, testing, and validation of the face recognition pipeline. Supported model evaluation activities and helped ensure consistency and correctness of training data and experimental results.

All team members collaborated in testing, reviewing system outputs, and preparing documentation and reports to ensure successful completion of the fall semester deliverables.

8.References:

1] Khaled, K. (n.d.). *Computer vision playlist* [Video playlist]. YouTube. https://youtube.com/playlist?list=PL5JZLxl_tFCeB5bOLYGXNdjnmgRIU6Ov0

2] OpenCV Team. (n.d.). *OpenCV library*. https://opencv.org/

9.Appendix :

GitHub Repository:

The complete source code, documentation, and project updates for the Smart Attendance System are available on GitHub at the following link:

https://github.com/Yassmin-raafat/Smart-Attendance-App