

ClassLens

AI ATTENDANCE MANAGER DOCUMENTATION

Student1656558	OMOBOLANLE ADEGBOYE
Student1528971	MUDATHIR AJIBOLA
Student1511338	OGUNJIOFOR CHUDI
Student1560863	AJEBEMIKA SIMIOLUWA

APTECH LIMITED MARYLAND

TABLE OF CONTENTS

SECTION I: PROJECT SYNOPSIS

- 1.1 Project Title**
- 1.2 Project Overview**
- 1.3 Objectives**
- 1.4 Scope of the Project**

SECTION II: SYSTEM ANALYSIS

- 2.1 Problem Statement**
- 2.2 Proposed Solution**
- 2.3 System Requirements**
- 2.4 Functional Requirements**
- 2.5 Non-Functional Requirements**

SECTION III: SYSTEM DESIGN

- 3.1 System Architecture**
- 3.2 Dialog Flow**
- 3.3 Use Case Diagram**

SECTION IV: IMPLEMENTATION

- 4.1 Technologies Used**
- 4.2 System Features**
- 4.3 Backend Implementation**
- 4.4 Frontend Implementation**
- 4.5 Testing**

SECTION VIII: CONCLUSION AND RECOMMENDATIONS

- 8.1 Conclusion**

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to Aptech Computer Education for providing us with the platform to develop and apply our knowledge in software

development. We are especially grateful to our mentor for their insightful guidance and support

SECTION I: PROJECT SYNOPSIS

1.1 Project Title

ClassLens - AI Attendance Manager

1.2 Project Overview

ClassLens is an AI-powered attendance management system designed to make tracking students' attendance fast, accurate and easy to operate. Using a camera and AI model, the system recognizes students automatically, marks their attendance in real time and stores the data in a database. It generates reports that help monitor and save time for both staff and students.

1.3 Objectives

- Capture and process student images in real-time.
- Train and implement a facial recognition to identify each student
- Track the in-time and out-time for each student.
- Automate the attendance process and update records.
- Prevent proxy attendance using facial recognition.
- Deliver a user-friendly dashboard for admin and teachers to view records.

1.4 Scope of the Project

- Identify students based on pre-stored images to ensure authenticity and prevent proxy attendance.
- Record the exact duration each student is present in a classroom.
- Store and organize student images securely to build a reliable dataset for training the facial recognition system.
- Provide a user-friendly dashboard where teachers and admins can view attendance records and generate reports easily.

SECTION II: SYSTEM ANALYSIS

2.1 Problem Statement

Before now, Attendance management has proven to be a disadvantage to educational institutions, such as:

- Manual attendance is time-consuming and takes away from valuable instructional time.
- Traditional methods are prone to errors, including mistakes in marking and record-keeping.
- Proxy or fake attendance is common, reducing the reliability of attendance data.
- Managing large class sizes is difficult with manual methods, leading to inefficiency.
- Conventional systems cannot track in-time and out-time, making it hard to measure actual student presence duration

2.2 Proposed Solution

ClassLens is designed to provide an AI-powered attendance system that uses facial recognition to automate and secure the process. The system captures images of students as they enter by using a camera module in the classroom and matches them with a stored database. Once a face is verified, the system logs attendance in real time. The system will:

- Provides real-time attendance tracking.
 - Eliminates the risk of proxy attendance.
 - Gets rid of traditional attendance methods.
 - Enhances accuracy, reliability, and efficiency for both staff and students
-
- Automates attendance through facial recognition.
 - Captures both in-time and out-time of students.
 - Ensures authenticity by eliminating fake attendance.
 - Enhances overall class management and monitoring.

2.3 System Requirements

- **Frontend:** HTML5, CSS, JavaScript.
- **Backend:** Flask/Django.
- **Data Store:** jpg images.

- **Database:** Firebase/NoSQL.
- **Programming/IDE:** Python, VScode.
- **Libraries:** OpenCV, NumPy, Pandas, Tensorflow, PyTorch

2.4 Functional Requirements

User Authentication:

- **Users (teachers, students, administrators) must be able to create accounts, log in, and log out.**
- **Authentication should be secure and include password recovery options.**

Mark Attendance:

- Teachers can mark attendance for students manually or using AI-powered facial recognition.
- Automatic detection of student presence through live video feed or uploaded classroom images.
- Ability to correct or override AI-detected attendance.

View Attendance Records:

- Teachers and students can view attendance history.
- Attendance can be filtered by date, class, subject, or student.
- Ability to export attendance records as reports (PDF, CSV).

Notifications & Alerts:

- Notify students of their attendance status or if they are marked absent.
- Teachers receive alerts for irregular attendance patterns or anomalies.

User Profiles:

- Users can create and edit profiles including personal information and role (student/teacher/admin).
- Teachers can manage class lists and student information.

Analytics & Reports:

- Generate reports summarizing attendance trends over time.
- Visualizations such as charts or graphs showing class attendance statistics.

Integration with Class Schedules:

- Sync attendance data with class schedules or timetables.
- Ability to handle multiple classes and sessions simultaneously.

Security & Privacy:

- Ensure that attendance data is securely stored and only accessible to authorized users.
- Protect student images and personal information according to data privacy standards.

Non-Functional Requirement Specification

Performance:

- The system should process attendance marking in near real-time.
- AI detection should handle classrooms of up to 50–100 students without noticeable delays.

Reliability:

- The system must be available during school hours with minimal downtime.
- Backups should be maintained to prevent data loss.

Scalability:

- Should support multiple classes, teachers, and students as the institution grows.

- Can integrate with additional AI modules or features in the future.

Usability:

- User interface should be intuitive for both teachers and students.
- Minimal training should be required to use the system effectively.

Security:

- Use encryption for sensitive data, including passwords and student images.
- Ensure secure communication between the application and server.

Compatibility:

- Should work on web and mobile platforms (cross-platform support).
- Compatible with standard browsers and Android/iOS devices.

Maintainability:

- Code and system should be modular and well-documented for easy updates.
- Support versioning and patch updates without affecting ongoing attendance operations.

Data Privacy:

- Comply with local regulations on student data privacy (e.g., GDPR, FERPA).
- Ensure consent is obtained before capturing student images.

SECTION III: SYSTEM DESIGN

3.1 System Architecture

The system consist of:

3.2 Use Case Diagram:



SECTION IV: IMPLEMENTATION

4.1 Technologies Used:

- **Frontend:** HTML5, CSS, JavaScript.
- **Backend:** Flask/Django.
- **Data Store:** jpg images
- **Database:** Firebase, NoSQL.
- **Libraries:** OpenCV, TensorFlow, NumPy, Pandas.

4.2 System Features:

- Automated Attendance Marking (Camera & AI Recognition).
- Proxy Attendance Prevention (Verification System)
- Real-Time Tracking & Reporting (Dashboard Analytics)
- Time Logging (In-Time & Out-Time Records)
- Data Security & Privacy (Encrypted Storage)
- Scalability & Integration (Flexible Deployment)

/ClassLens

|

|— app.py

|— attendance_ai.py

| # - Processes images/video feeds


```
| # - Detects and recognizes student faces
|
| # - Marks attendance and returns results to app.py
|
|
|— templates/
|   |— base.html
|   |— login.html
|   |— dashboard.html
|   |— student_view.html
|   |— attendance_report.html
|
|
|— static/
|   |— css/
|   |— js/
|   |— images/
|
|
|— notebooks/
|   |— eda.ipynb
|   |— model_training.ipynb
```

GitHub URL

<https://github.com/Jibex-Banks/Ai-Attendance-Manager>

4.3 Frontend Implementation:

- The application was developed with Flutter for cross-platform support.
- It includes key screens such as Login, Home, Conversion, Alerts, History, Preferences, Support, and Feedback.
- Firebase is integrated to handle authentication and notifications.

4.4 Backend Implementation:

- The backend was developed using Flask/Django to manage server-side logic and API communication.
- It is responsible for handling requests from the frontend, processing image data, and interacting with the database.
- Firebase/NoSQL is integrated for data storage and retrieval, ensuring scalability and real-time updates.
- The backend ensures secure communication, accurate attendance logging, and real-time synchronization with the frontend.

4.5 Testing

- **Unit Testing:** Each individual component (such as face detection, data storage, and attendance logging) was tested separately to confirm correct functionality.
- **Integration Testing:** Modules were combined and tested together to verify smooth communication between the frontend, backend, and database.
- **User Acceptance Testing (UAT):** Real users (teachers and administrators) interacted with the system to validate that it met the intended requirements and was user-friendly.

5.1 Project Outcome:

- Attendance marking time was reduced from several minutes to a matter of seconds, allowing teachers to dedicate more time to teaching.
- Facial recognition ensured that only physically present students were marked, eliminating proxy or fake attendance.
- Teachers and administrators could monitor the in-time and out-time instantly through the dashboard, giving them real-time visibility.
- Attendance records were automatically stored in a database, enabling the generation of reports for performance evaluation and administrative decision-making.

CONCLUSION

The development of ClassLens demonstrates how Artificial Intelligence can transform classroom management by automating attendance tracking. Unlike traditional roll calls and manual registers, this system provides real-time, accurate, and secure attendance records with minimal effort from teachers.

By integrating face recognition, database storage, and a user-friendly dashboard, ClassLens reduces errors, prevents proxy attendance, and saves valuable teaching time.

The system not only benefits teachers and administrators but also creates a more efficient and transparent learning environment for students.

This project highlights the potential of AI-driven solutions in education and sets the foundation for future enhancements such as cloud storage, mobile app integration, and multi-classroom support.

Blog Link

<https://classlensvision.blogspot.com/2025/09/ai-and-machine-learning-projects-arent.html>