Project Report

FUMB - Face Capture Attendance System

Department	**Computer System Engineering**
Class	**24 CS Section 1**
Submitted By	**Umer Qureshi (24CS05)**
	Burhan Shaikh (24CS13)
	Musab Khan (24CS19)
	Fahees Shaikh (24CS107)
Submitted To	**Sir Fawad Mangi**
Subject	**Computer Programming (Practical)**
Course Code	**CS-151**

Contents

1. Introduction

1.1 Attendance System

An **attendance system** is a method used to track and record the presence or absence of individuals, such as students in a classroom or employees in a workplace. Traditional attendance tracking often involves manual methods, such as using a register or a sign-in sheet, where participants mark their presence. While manual systems are simple and require no technological setup, they are prone to errors, such as false entries, time wastage, and difficulty in managing large groups. In contrast, automated systems, like **biometric** attendance or **face recognition** technology, offer more **accuracy**, **efficiency**, and **security**. Modern solutions also enable integration with digital platforms, allowing for real-time data access and analysis. As organizations grow, automated systems are increasingly preferred over manual processes to ensure reliability and scalability.

1.2 Problem Statement

Attendance management remains a critical and time-consuming task in academic institutions and organizations. Traditional attendance systems such as manual registers, roll calls, or paper-based logs suffer from several significant drawbacks:

- **Proxy Attendance**: Individuals may mark attendance on behalf of others, leading to inaccurate records.
- **Time Inefficiency**: Roll calls and manual data entry consume valuable time, affecting overall productivity.
- **Security Concerns**: Manual attendance systems are susceptible to unauthorized changes and tampering.

These challenges necessitate the development of an automated, secure, and efficient solution for attendance management.

1.3 Proposed Solution

To address these challenges, we developed a **Face Capture-Based Attendance System** that utilizes **facial detection** technology to automate the process of attendance marking. This system ensures:

- **Accurate Attendance Logging**: Face capture guarantees that only the right individual can mark their attendance.
- **Time Efficiency**: Attendance is automatically marked, eliminating the need for roll calls.
- **Secure Logs**: All attendance records are stored securely, with password protection for sensitive data.
- **Scalability**: The system can easily scale to accommodate both students and teachers

- **Cost Efficiency**: This system decreases the cost of Attendance systems as compared to other traditional manual methods.

---

2. Research Objective

- To design and develop an efficient and user-friendly **face capture attendance system** integrated with a web portal.
- To automate the attendance process by leveraging **face capture technology** for accurate identification and eliminating the need for traditional manual methods.
- To provide a **secure, scalable, and easily accessible solution** for attendance tracking.
- To use **C++ as the core programming language** for development and **CSV files** for data storage, ensuring simplicity in implementation and management.
- To enable **online functionality**, allowing seamless integration with web-based platforms for **real-time attendance monitoring and reporting**.

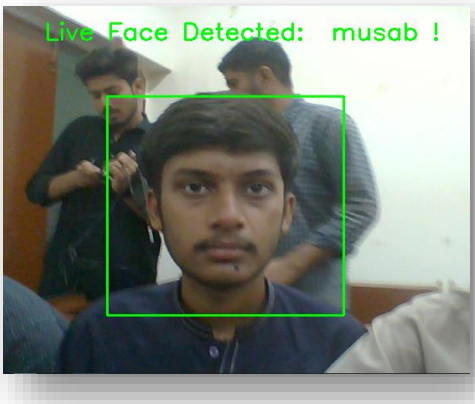
---

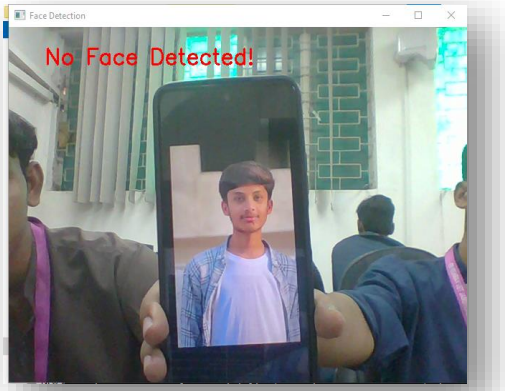
3. System Overview

3.1 Face Detection Module

This module leverages **OpenCV** to perform face detection and capture. The system captures the individual's face through the webcam, saves it in directory named as **student_photos**, and marks the attendance if a match is found in database file.

- **Face Detection**: The system uses **Haar Cascades** for detecting faces in real-time.
- **Face Capture**: The system stores images of students and teachers in separate directories named as **student_photos** & **teacher_photos**.





3.2 Attendance Database

Attendance records are stored in **CSV** files to ensure easy access and management. There are separate CSV files for **students** (students_attendance.csv) and **teachers** (teachers_attendance.csv). Each file logs holds the following information:

- **Name**
- **Roll/Registration Number**
- **Timestamp**

Date	Name	ID	Photo Path
########	umer	24cs05	Manual Entry
########	musab	24cs19	Manual Entry
11/1/2024	musab	24cs19	student_photos/musab_24cs19.jpg
11/1/2024	fahees	24cs107	student_photos/fahees_24cs107.jpg
11/1/2024	burhan	24cs13	student_photos/burhan_24cs13.jpg
11/1/2024	oun	24cs37	student_photos/oun_24cs37.jpg

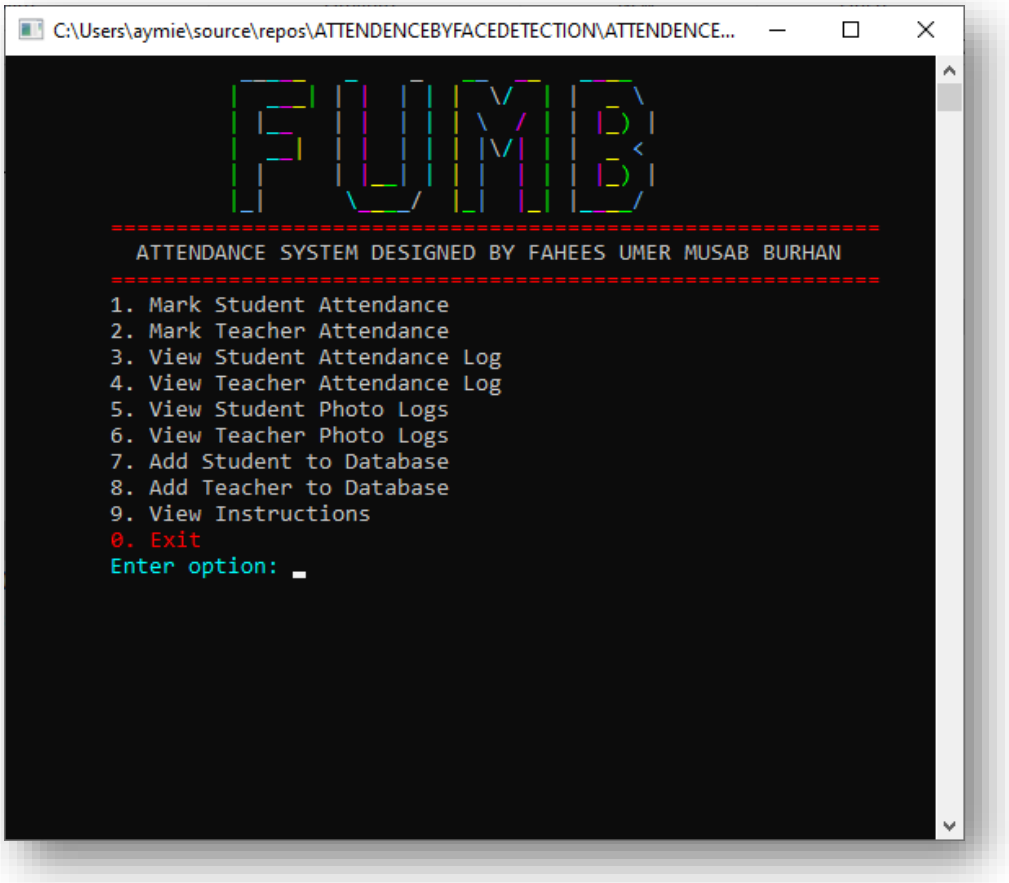
3.3 Photo Logs

The system also maintains a photo log of individuals for reference purposes. Captured images are saved in the student_photos directory with the person's **roll number** or **registration number** as the filename.

3.4 Interactive Menu

An interactive colorful console menu guides the user through the available operations:

- Marking attendance for students or teachers
- Viewing attendance logs
- Accessing photo logs with password protection
- Adding students and teachers to database file
- Viewing Instructions



3.5 Web Portal

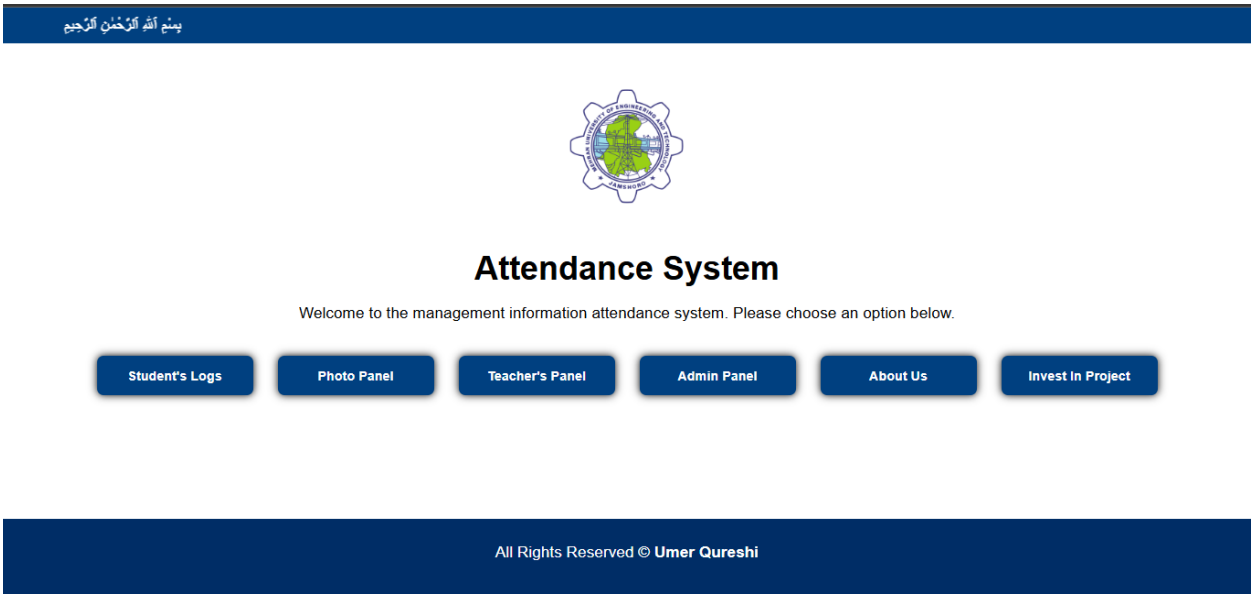
A complete Website through which students, teachers and admin can access or view attendance logs, Photos, Records and anyone can provide feedback, Invest in Project, And view developers page.

- Students Text logs
- Students Photos logs
- Teachers Portal
- Admin Portal
- Invest In Project
- About Us

- Contact Us
- Developers Page

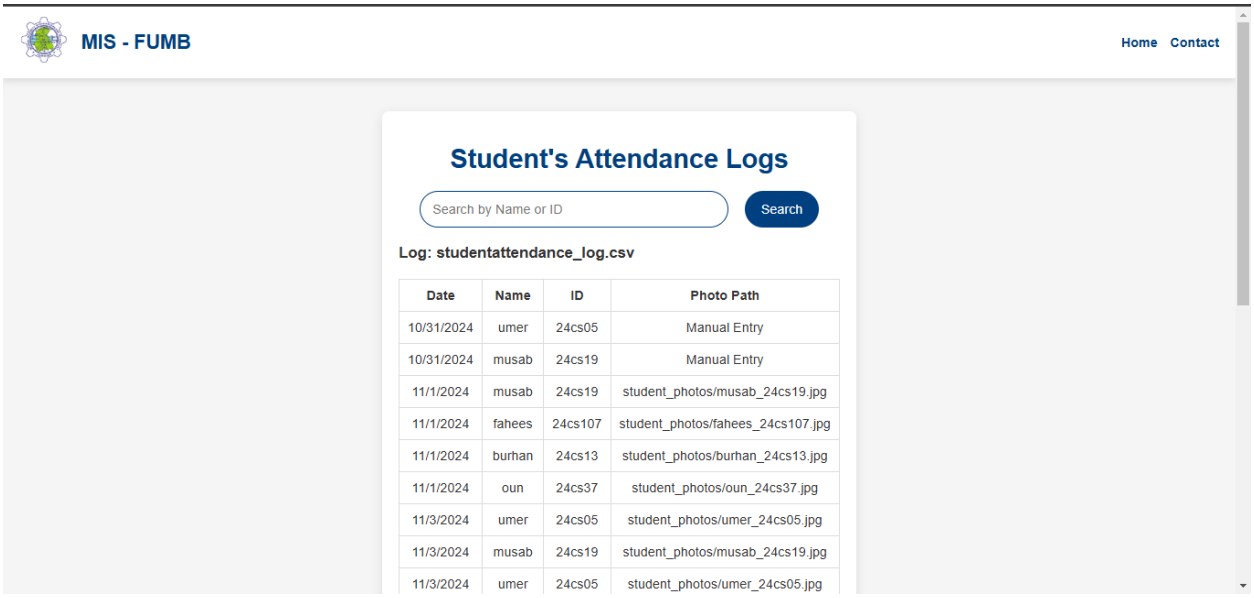
3.5.1 Main Page

The main page of the portal displays various options such as students and teacher log, teachers and admin panel, about us and option to invest in the project.



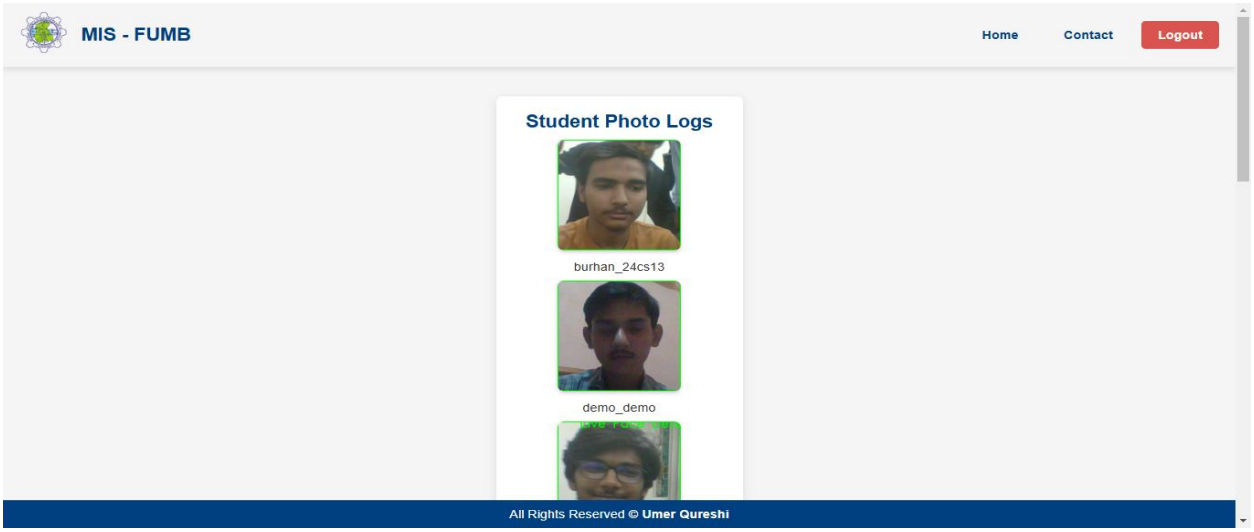
3.5.2 Students Text Logs

Student attendance log provides the user to view and search attendance of student by name, roll number or even by the date.



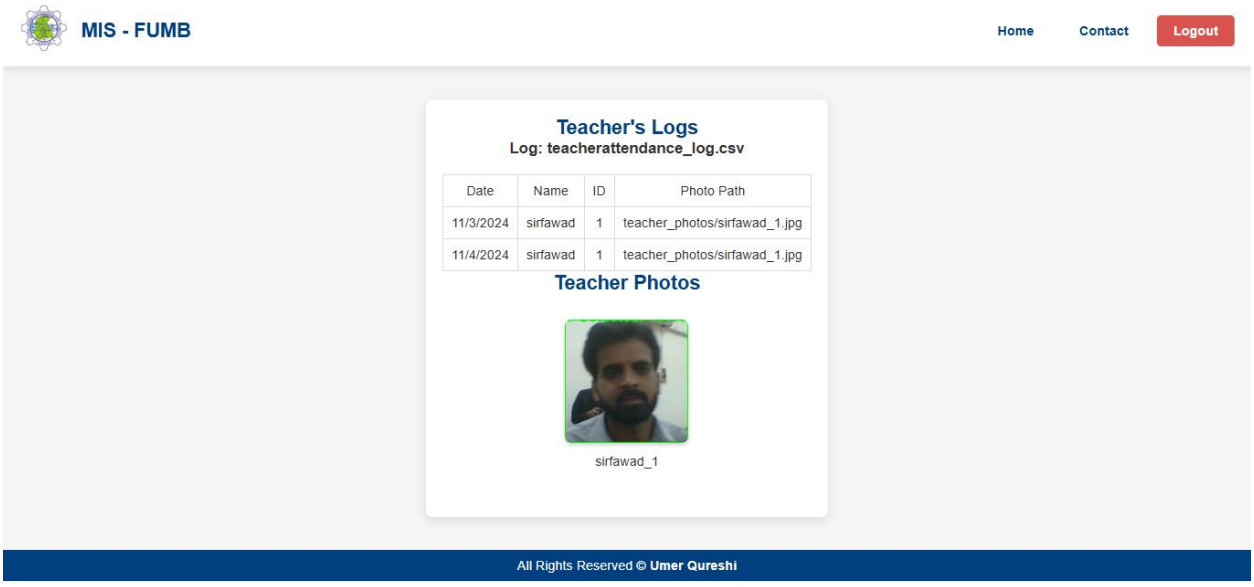
3.5.3 Students Photos Portal

Student photos log provides the photo log of the students who marked their attendance along with their roll numbers.



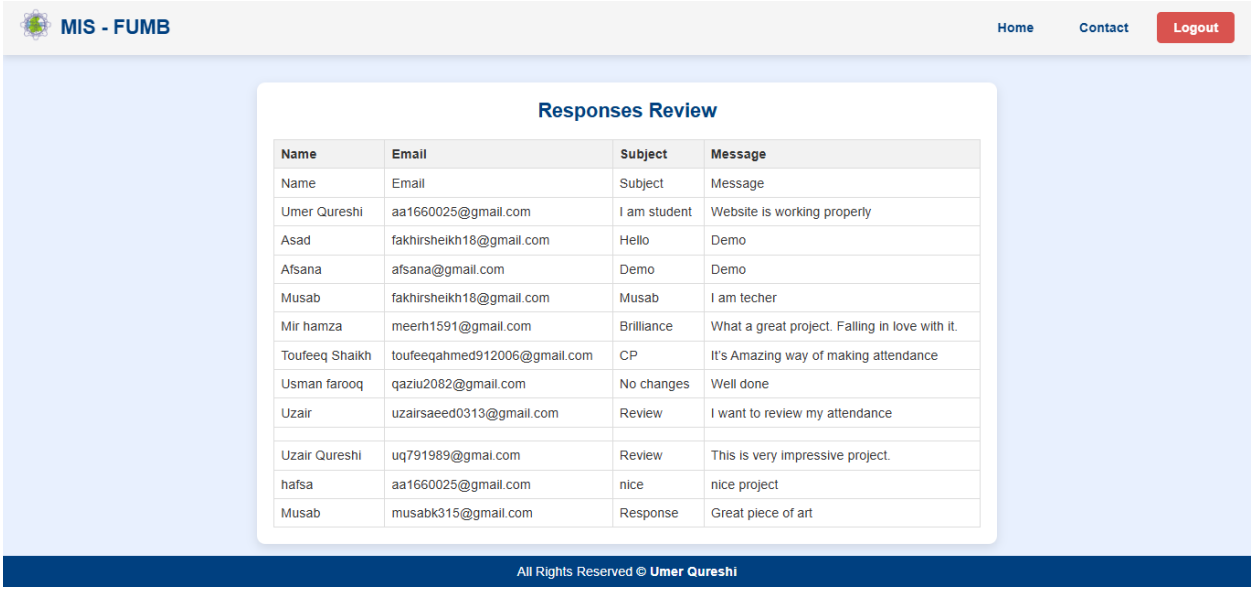
3.5.4 Teachers Portal

Teachers' portal which is secured by password allows admin to access picture and attendance logs of teacher to view.



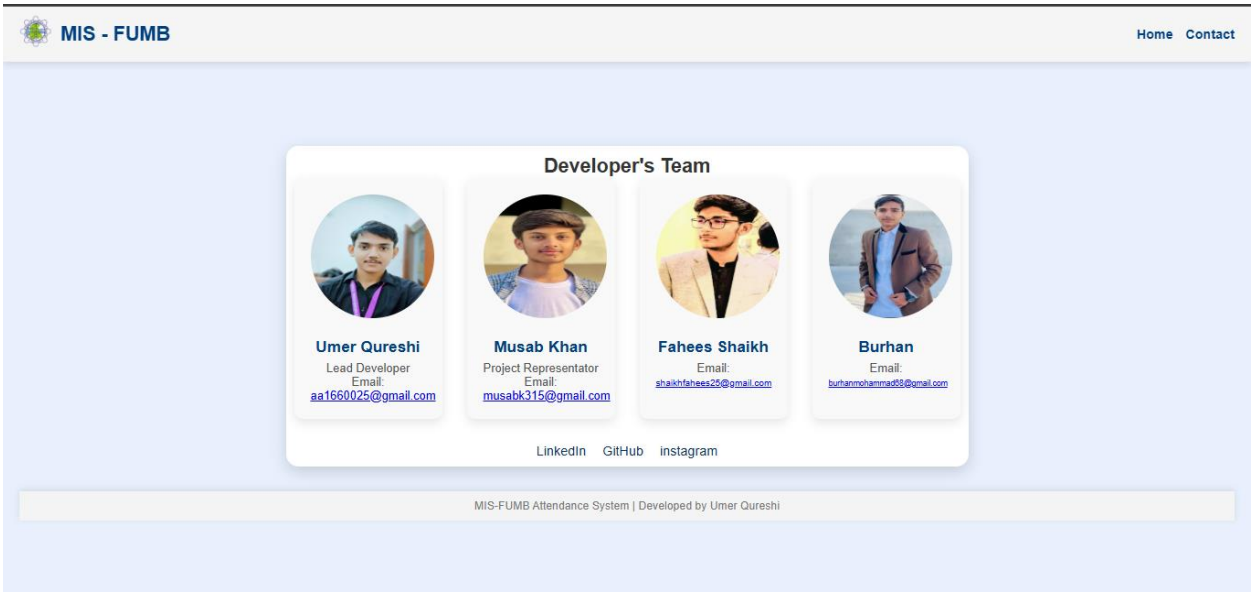
3.5.5 Admin Portal

Admin portal allows admin to view all feedbacks provided by users along with their gmail ID.



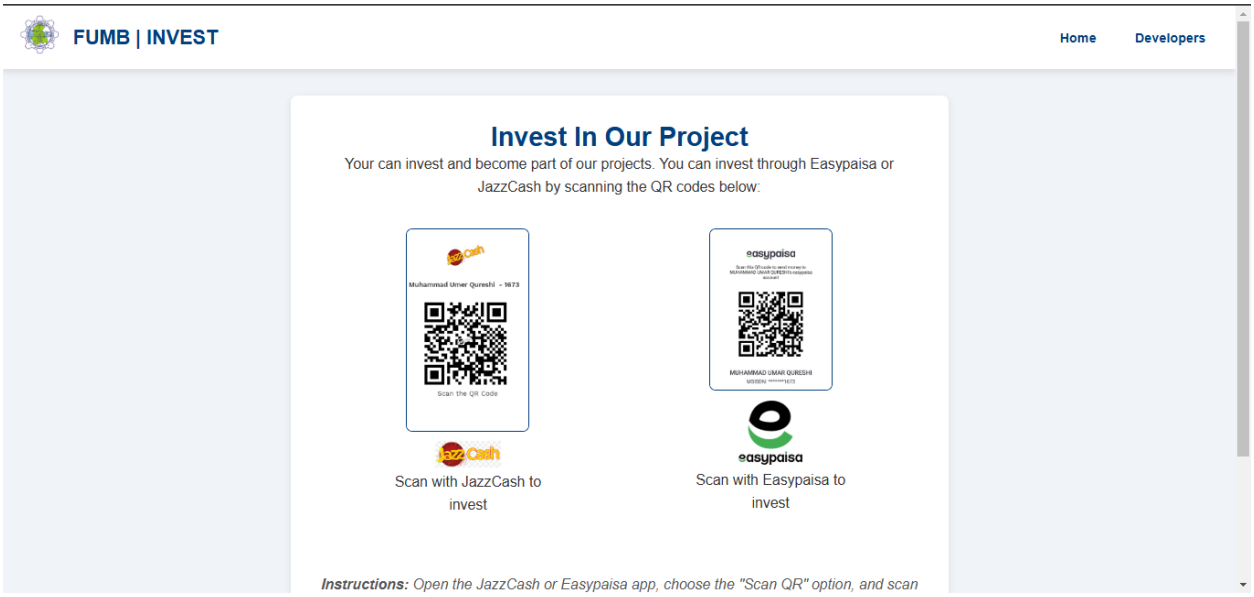
3.5.6 Developers Page

The Developers page displays the developers along with their pictures, email, and with insta, GitHub and LinkedIn profile.



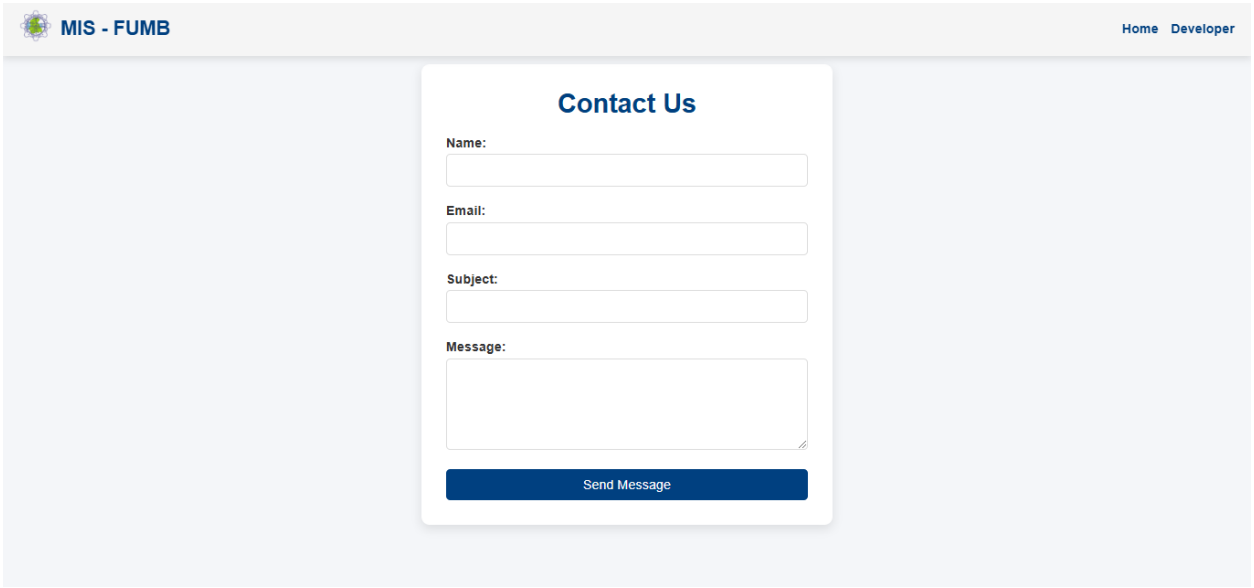
3.5.7 Invest in Our Project

Invest page allows user to invest in our project through easypaisa or jazz Cash which in future would be helpful to buy domain .



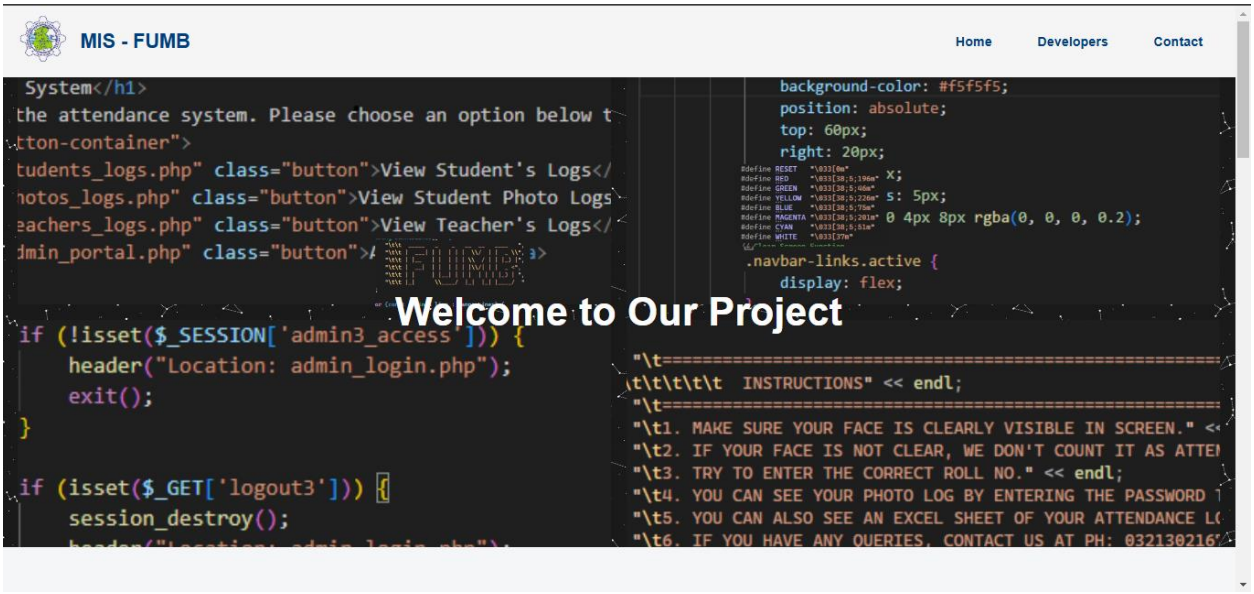
3.5.8 Contact Us

Contact us form allows user to provide admin with feedback or a text of appreciation.



3.5.9 About Us

This page contains overview and abstract of our project.



---

4. Code Analysis

4.1 AttendanceSystem Class

The **AttendanceSystem** class is the heart of the system, encompassing all core functionalities. Below is a detailed breakdown of each method.

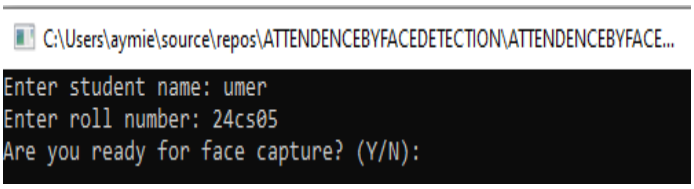
```
class AttendanceSystem {
public:
    void markStudentAttendance(const string& name, const string& rollNumber);
    void markTeacherAttendance(const string& name, const string& regNumber);
    void viewStudentAttendance() const;
    void viewTeacherAttendance() const;
    void viewPhotoLogs(const string& photoDir) const;
    bool detectFace(Mat& frame, CascadeClassifier& faceCascade, Rect& face) const;
    bool detectBlink(Mat& faceROI) const;
    VideoCapture selectCamera() const;

private:
    void markAttendance(const string& name, const string& id, const string& logFileName, const string& photoDir) const;
    void displayAttendance(const string& logFileName) const;
};
```

4.1.1 markStudentAttendance()

Purpose: This method marks the attendance for a student.

Functionality:

1. **Input**: The user is prompted to enter the student's name and roll number.
2. **Face Capture**: The webcam is activated using **OpenCV's VideoCapture** to capture a photo of the student.
3. **Save Photo**: The captured image is saved to a directory named student_photos, with the student's roll number as the file name.
4. **Log Attendance**: The system appends the student's name, roll number, and the timestamp of the attendance to the students_attendance.csv file.

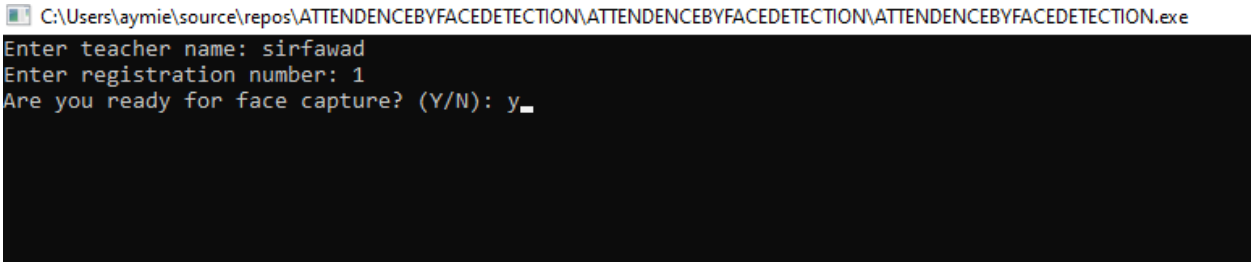


4.1.2 markTeacherAttendance()

Purpose: Similar to markStudentAttendance(), but for teachers.

Functionality:

- Prompts the user for the teacher's **name** and **registration number**.
- Captures the teacher's photo and logs the attendance in the teachers_attendance.csv file.

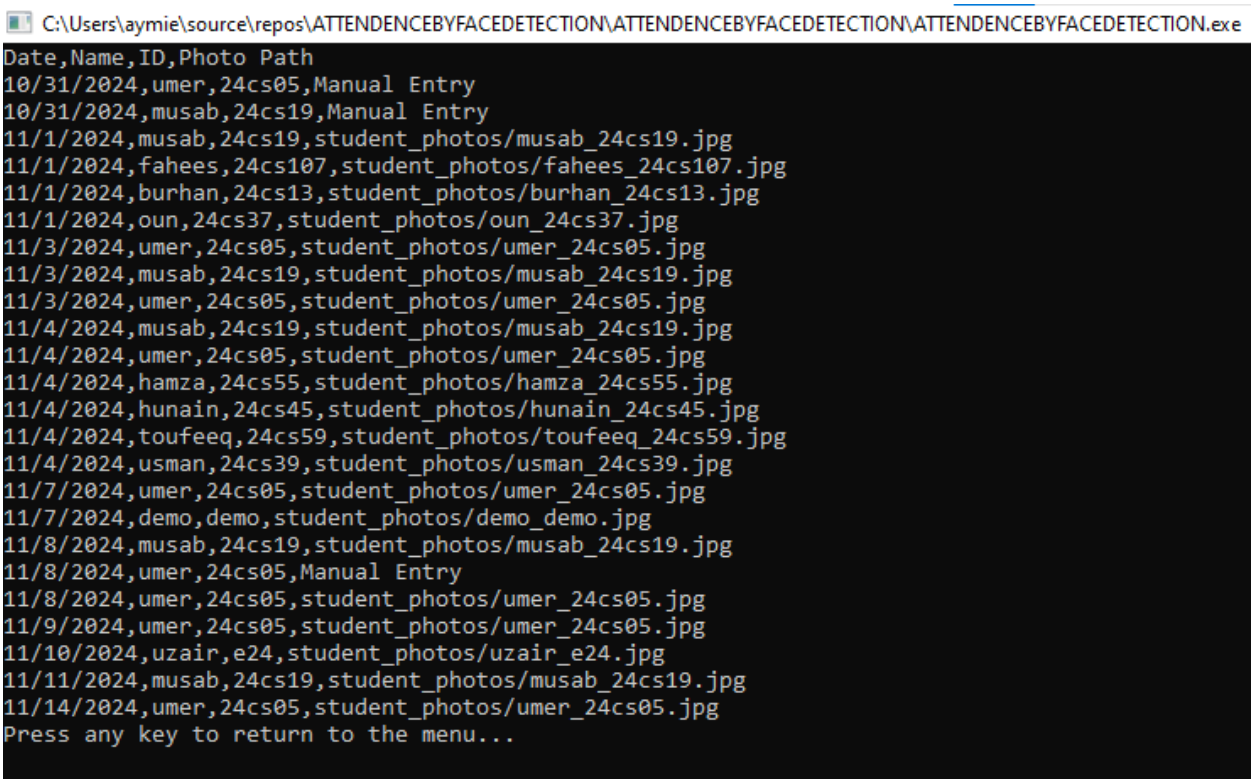


4.1.3 viewStudentAttendance()

Purpose: Displays the attendance records for students.

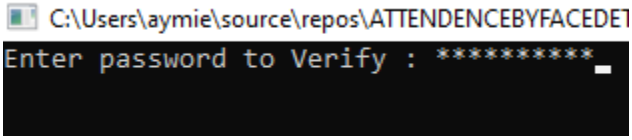
Functionality:

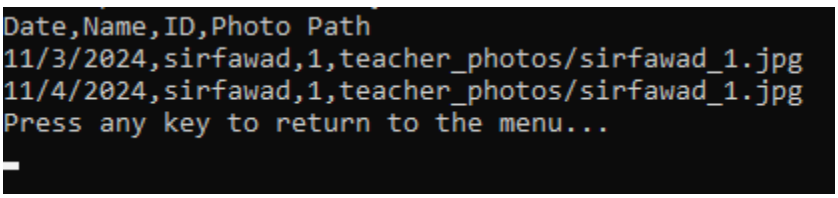
- Opens the students_attendance.csv file and reads its content line by line.
- Prints the student's attendance, including their name, roll number, and timestamp.



4.1.4 viewTeacherAttendance()

Purpose: Similar to viewStudentAttendance(), but secured with password protection displays teacher attendance records.





4.1.5 displayAttendance()

Purpose: A helper function to display any attendance log, either student or teacher.

Functionality:

- Accepts the file name as input (e.g., students_attendance.csv) and displays its contents.

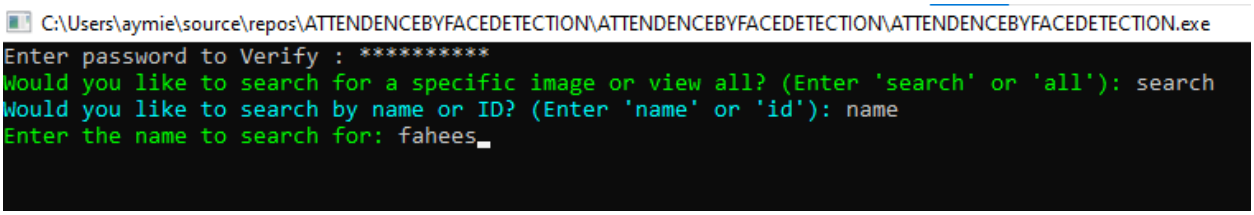
```
void AttendanceSystem::displayAttendance(const string& logFileName) const {

    ifstream log(logFileName);
    string line;

    if (!log.is_open()) {
        cerr << WHITE << "Error opening log file: " << logFileName << RESET << endl;
        return;
    }

    while (getline(log, line)) {
        cout << WHITE << line << RESET << endl;
    }

    log.close();
}
```

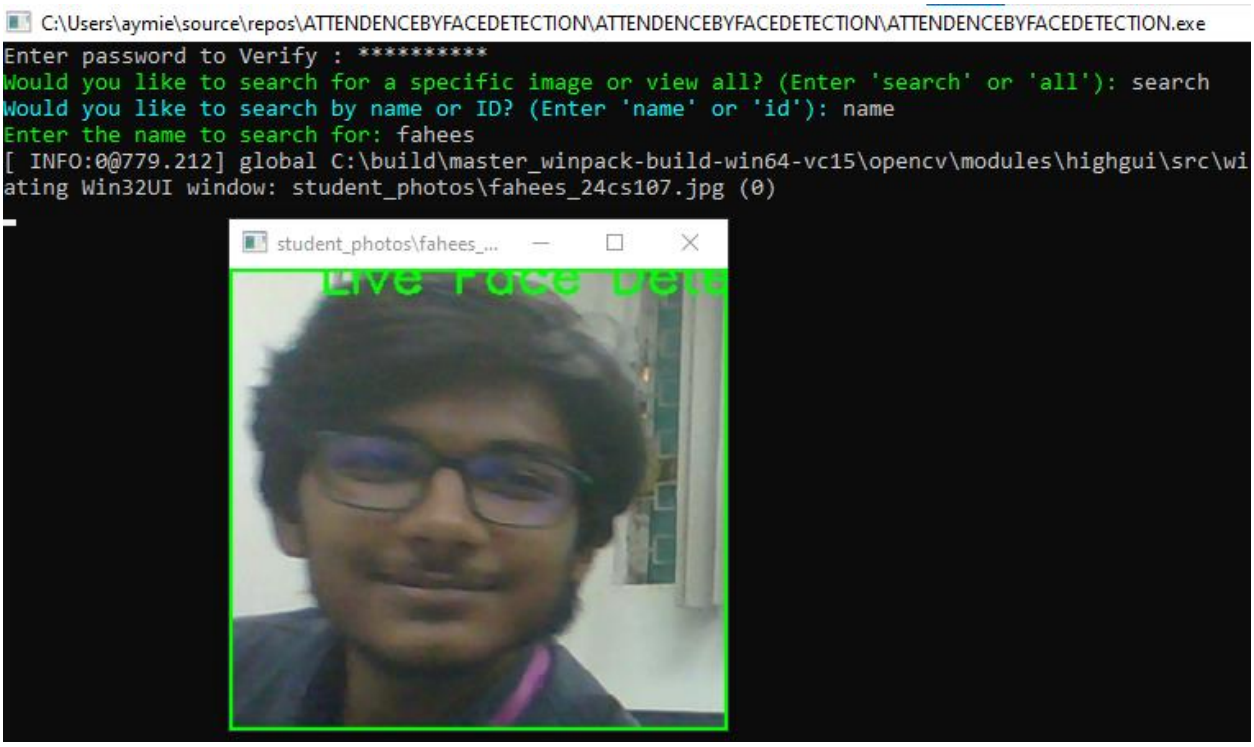
4.1.6 viewPhotoLogs()

Purpose: Allows the user to view saved photo logs.

Functionality:

1. **Password Protection**: Before displaying the photo logs, the user must input the correct password.
2. **Search by Name or ID**: Users can search for photos by entering the **name** or **ID**.
3. **Image Display**: The images are displayed using **OpenCV**.



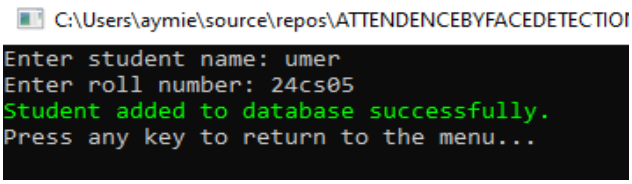


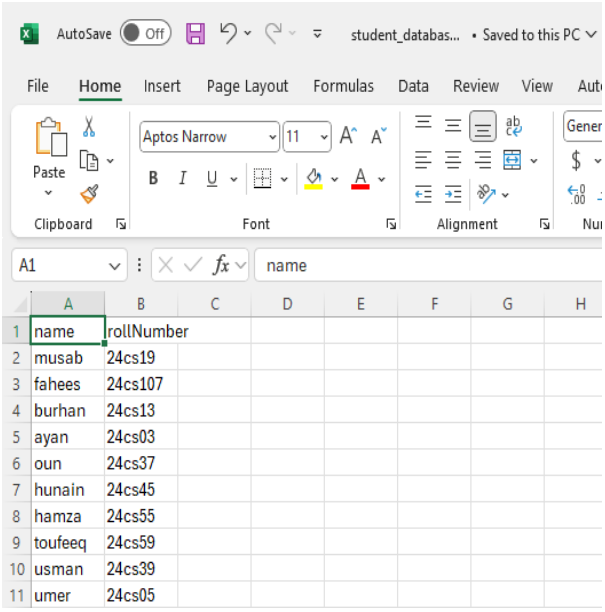
4.1.7 addStudentToDatabase() and addTeacherToDatabase()

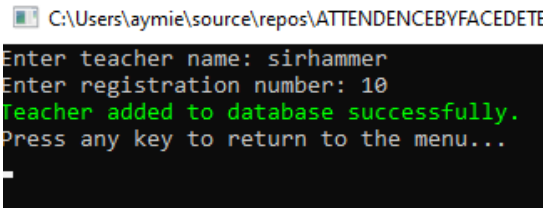
Purpose: These functions allow adding new students and teachers to the respective databases after entering admin password.

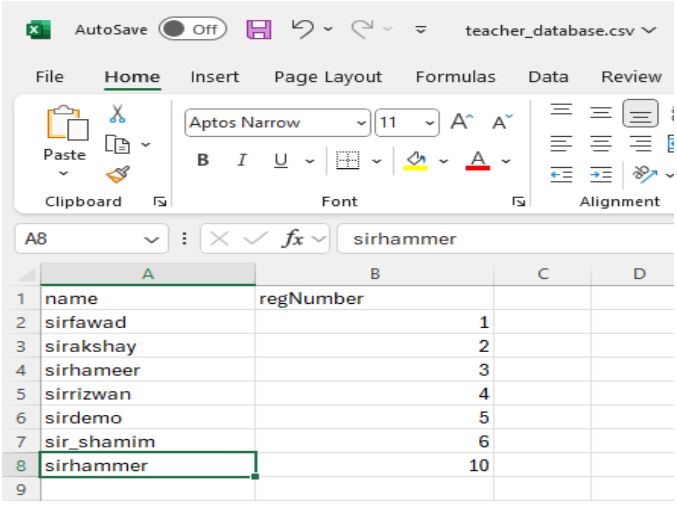
Functionality:

- The system prompts the user to enter the name, roll number, or registration number of the new student or teacher.
- The information is then added to the corresponding CSV file.









	A	B	C	D
1	name	regNumber		
2	sirfawad	1		
3	sirakshay	2		
4	sirhameer	3		
5	sirrizwan	4		
6	sirdemo	5		
7	sir_shamim	6		
8	sirhammer	10		
9				

4.2 Supporting Functions

4.2.1 password()

Purpose: This function provides password protection for sensitive data, particularly photo logs.

Functionality:

- The entered password is compared with a predefined password stored in the system.

```
Enter password to Verify : **********
```

4.2.2 returnToMenu()

Purpose: This function returns the user to the main menu after completing any operation.

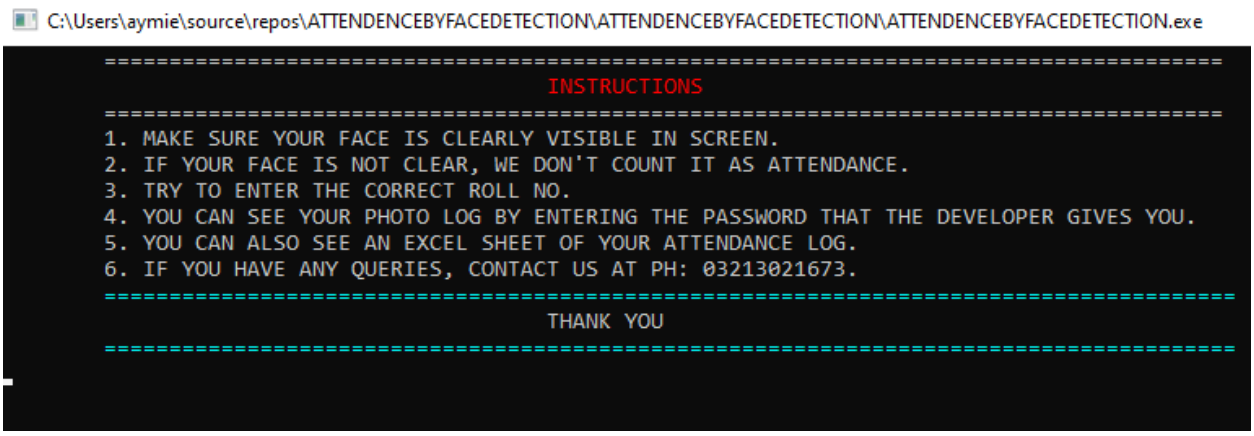
```
Press any key to return to the menu...
```

4.2.3 clearScreen()

Purpose: Clears the console screen to improve user experience and readability.

4.2.4 Instructions()

Purpose: Displays instructions on how to use the system, including steps for marking attendance, viewing logs, and searching photos.



---

5. New Features

5.1 IP Camera Support

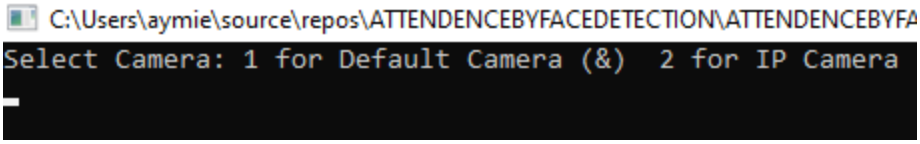
This feature provides a facility to users that they can use their phone camera for face capturing by using app (IP Webcam).

1. Users must install IP Webcam on their smartphone or laptop.
2. Start video server from their device.
3. Then type their server URL on our software then this software will user user's camera for OpenCV face capturing.
4. In case of connection failure our default camera will be used
5. URL example: (e.g. http://192.168.0.38:8080/video)

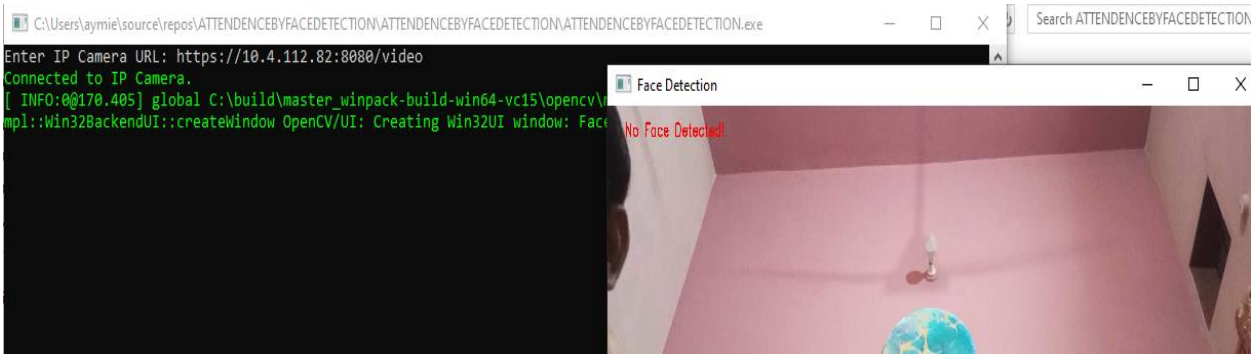
Step 1:

Step 2:

Step 3:



Step 4:



---

6. Main Features

- Cross Platform compatibility (Windows, Linux, MacOS)
- IP Webcam support
- Error handling
- Web Portal Integration
- Search by name or id support

---

7. Technologies Used

- C++ ..... (For Main Program)
- OpenCV ..... (For face detection & capture)
- CSV files ..... (For database and storage of logs)
- PHP ..... (For Web Portal)
- CSS ..... (For Styling)
- HTML ..... (For Core Web Portal Structure)
- JavaScript ..... (For Responsiveness and other functionalities)

---

8. Team Contributions

Umer Qureshi (24CS05)

- Implemented core attendance logging functionalities and CSV-based database management and Designed **Web Portal**.

Burhan (24CS13)

- Integrated **OpenCV** for face capture and photo logging, developed search and view features for logs and photos.

Musab (24CS19)

- Developed the **user authentication** module and implemented password protection for sensitive logs.

Fahees (24CS107)

- Designed the **interactive menu** and improved **user experience**, including error handling and system instructions.

---

9. Conclusion

The **Face Capture-Based Attendance System** effectively addresses the challenges of traditional attendance methods by automating the process using facial detection technology. The system is designed to be secure, scalable, and efficient. We have successfully achieved the objectives outlined in our initial presentation. Currently, the system is in its testing phase and may still have minor glitches. We are actively working to improve its functionality and incorporate new features. In conclusion, this project is ready to resolve the issues highlighted in the problem statement and is suitable for implementation in small-scale organizations.

10. Future Enhancements

- Implementing GUI (Graphical User Interface).
- Creating Database (e.g. MYSQL).
- Increase Scalability

PROJECT ASSESSMENT

Course code:	CS-151	Course Name:	Computer Programmming
Roll No:	24CS05	Student Name:	Muhammad Umer Qureshi

Performance Parameter	Excellent (10)	Good (7)	Satisfactory (5)	Poor (2)	Score (S)
Completeness and Accuracy	The project is complete and accurate in the context of installation and understanding	The project is mostly complete and accurate in the context of installation and understanding	The project is half complete and accurate in the context of installation and understanding	The project in its current state is not complete	
Delivery	Completed 100% of requirements	Completed between 80-95% of the requirements.	Completed at least 75% of the requirements. Delivered on time, and in correct format.	Completed less than 75% of the requirements. Not delivered in time or not in correct format. Does not comply with requirements	
Timeliness	The project is completed in due time	The project is mostly completed in due time	The project is half completed in due time	The project is not completed in due time	

Total Score (TS) (Out of 30):		
Normalized Score (NS) (Out of 1):	***NS = (TS/3 x 10)***	

Teacher Signature: _____ Date: _____

PROJECT ASSESSMENT

Course code:	CS-151	Course Name:	Computer Programmming
Roll No:	24CS13	Student Name:	Muhammad Burhan

Performance Parameter	Excellent (10)	Good (7)	Satisfactory (5)	Poor (2)	Score (S)
Completeness and Accuracy	The project is complete and accurate in the context of installation and understanding	The project is mostly complete and accurate in the context of installation and understanding	The project is half complete and accurate in the context of installation and understanding	The project in its current state is not complete	
Delivery	Completed 100% of requirements	Completed between 80-95% of the requirements.	Completed at least 75% of the requirements. Delivered on time, and in correct format.	Completed less than 75% of the requirements. Not delivered in time or not in correct format. Does not comply with requirements	
Timeliness	The project is completed in due time	The project is mostly completed in due time	The project is half completed in due time	The project is not completed in due time	

Total Score (TS) (Out of 30):		
Normalized Score (NS) (Out of 1):	***NS = (TS/3 x 10)***	

Teacher Signature: _____ Date: _____

PROJECT ASSESSMENT

Course code:	CS-151	Course Name:	Computer Programmming
Roll No:	24CS19	Student Name:	Musab Khan

Performance Parameter	Excellent (10)	Good (7)	Satisfactory (5)	Poor (2)	Score (S)
Completeness and Accuracy	The project is complete and accurate in the context of installation and understanding	The project is mostly complete and accurate in the context of installation and understanding	The project is half complete and accurate in the context of installation and understanding	The project in its current state is not complete	
Delivery	Completed 100% of requirements	Completed between 80-95% of the requirements.	Completed at least 75% of the requirements. Delivered on time, and in correct format.	Completed less than 75% of the requirements. Not delivered in time or not in correct format. Does not comply with requirements	
Timeliness	The project is completed in due time	The project is mostly completed in due time	The project is half completed in due time	The project is not completed in due time	

Total Score (TS) (Out of 30):		
Normalized Score (NS) (Out of 1):	***NS = (TS/3 x 10)***	

Teacher Signature: _____ Date: _____

PROJECT ASSESSMENT

Course code:	CS-151	Course Name:	Computer Programmming
Roll No:	24CS107	Student Name:	Muhammad Fahees Shaikh

Performance Parameter	Excellent (10)	Good (7)	Satisfactory (5)	Poor (2)	Score (S)
Completeness and Accuracy	The project is complete and accurate in the context of installation and understanding	The project is mostly complete and accurate in the context of installation and understanding	The project is half complete and accurate in the context of installation and understanding	The project in its current state is not complete	
Delivery	Completed 100% of requirements	Completed between 80-95% of the requirements.	Completed at least 75% of the requirements. Delivered on time, and in correct format.	Completed less than 75% of the requirements. Not delivered in time or not in correct format. Does not comply with requirements	
Timeliness	The project is completed in due time	The project is mostly completed in due time	The project is half completed in due time	The project is not completed in due time	

Total Score (TS) (Out of 30):		
Normalized Score (NS) (Out of 1):	***NS = (TS/3 x 10)***	

Teacher Signature: _____ Date: _____