A

PROJECT REPORT

ON

"AI-Powered Assistant for Classroom Attendance Using Face Recognition"

Submitted to

INTEL UNNATI INDUSTRIAL TRAINING 2025

Submitted by: ZONAL TECH

AKANKSHA G HOLI	(SG22AID004)
H VAISHNAVI	(SG22AID019)
SRIPAD S JUJAR	(SG22AID046)
VIDYASHREE	(SG22AID056)

Under the Guidence of:

Dr. LAXMI MATH

DEPARTMENT OF

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

FACULTY OF ENGINEERING AND TECHNOLOGY[CO-ED]

SHARNBASVA UNIVERSITY KALBURAGI

2024-2025

CERTIFICATE

The team members of **Zonal-Tech**, Akanksha G Holi (SG22AID004), H Vaishnavi (SG22AID019), Sripad S Jujar (SG22AID046), and Vidyashree (SG22AID056), students of the Artificial Intelligence and Data Science Department, have successfully completed the project entitled **"AI-Powered Assistant for Classroom Attendance Using Face Recognition"** Under the valuable guidance of **Dr. LAXMI MATH** as part of the **INTEL UNNATI INDUSTRIAL TRAINING 2025**.

This project showcases the practical application of AI technologies, utilizing facial recognition, Python programming, and real-time computer vision to automate attendance systems. The work demonstrates innovation, technical proficiency, and a commitment to solving real-world challenges using artificial intelligence.

GUIDE **CHAIRPERSON**

Centenary Celebrated Sharnbasveshwar Vidya Vardhaka Sangha's

ಶರಣಬಸವ ವಿಶ್ವವಿದ್ಯಾಲಯ
SHARNBASVA UNIVERSITY

A State Private University approved by Govt. of Karnataka vide Notification No. ED 144 URC 2016 dated 29-07-2017
Recognised by UGC under Section 2f vide No. F.8-29/2017 (CPP-I/PU), dated 20-12-2017 & AICTE, CoA, PCI New Delhi

ACKNOWLEDGEMENT

Any achievement, be scholastic or otherwise does not depend solely on individual efforts but also on the guidance, encouragement, and cooperation of intellectuals, elders, and friends. Several personalities, in their capacities, have helped me in carrying out this project. I would like to take this opportunity to thank them all.

I extend my sincere thanks to Dr. Shivakumar Jawaligi, Dean, Faculty of Engineering & Technology (Co-Ed), Sharnbasva University, Kalaburagi, for his help and inspiration during the tenure of the course.

I also extend my sincere thanks to Dr. Gajendran Malshetty, Chairman, Department of Artificial Intelligence & Data Science, Faculty of Engineering and Technology (Co-Ed), for his constant encouragement.

I also extend my sincere thanks to my Guide Dr. LAXMI MATH, Department of Artificial Intelligence & Data Science, Faculty of Engineering and Technology (Co-Ed), for her valuable suggestions and encouragement.

I also extend my sense of gratitude and sincere thanks to all the faculty members of the Artificial Intelligence and Data Science Department, Faculty of Engineering and Technology (Co-Ed), for their constant support.

AKANKSHA G HOLI (SG22AID004)
H VAISHNAVI (SG22AID019)
SRIPAD S JUJAR (SG22AID046)
VIDYASHREE (SG22AID056)

ABSTRACT

This project focuses on creating a face recognition-based attendance system using Python, designed to run through the command prompt. Traditional methods like manual registers or roll calls are often slow, prone to errors, and can be misused through proxy attendance. To solve these problems, the system uses facial recognition, offering a secure and automatic solution suitable for schools, colleges, and offices.

The system is developed using Python libraries such as **OpenCV** for video and image processing, and the **face_recognition** library (based on dlib) for accurate face detection and recognition. A major feature of this project is its simplicity—it runs completely from the command prompt, making it easy for beginners and users who prefer simple, lightweight setups without using complex software.

The workflow is straightforward. After downloading and extracting the project, users install the necessary packages and run the main.py file through the command prompt. The system first captures 100 images of a user's face (in RGB and grayscale). These are stored and used to create a unique facial encoding for each person.

During attendance, the webcam captures live video. When a known face is detected, the system displays the user's name and records their attendance automatically in a CSV file. Each entry includes the ID, name, date, and time, which can be used later in Excel or other software for reporting and analysis.

The system is modular and includes parts for face image capture, recognition, user input (via command line), and attendance logging. Each step is easy to follow, and no advanced knowledge is needed to use it.

In conclusion, this project is a useful real-world application of AI and computer vision. It improves accuracy, prevents fake attendance, and is easy to use. Its simple design and cross-platform support make it ideal for both learning and practical use.

CONTENTS

SL.NO.	PARTICULARS	PAGE NO.
1.	INTRODUCTION	7
	1.1 Application	8
	1.2 Key feature	8
2.	PROBLEM STATEMENT	9
3.	OBJECTIVES	10
4.	SYSTEM SPECIFICATIONS	11
	4.1 Hardware Specifications	11
	4.2 Software Specifications	12
5.	ADVANTAGES AND DISADVANTAGES	13
6.	IMPLEMENTATION	14
	6.1 System Architecture	14
	6.2 Flowchart	15
	6.3 Workflow	16
7.	OUTPUT	17-18
	Result	19
	CONCLUSION	20

LIST OF FIGURES

Figure 5.1. System Architecture.....**14**

Figure 5.2. Flow Diagram of face recognition-based attendance system.....**15**

Figure 7.1. Output (Data Entry Of Student)……**17**

Figure 7.2. Output (Image capturing)..... **17**

Figure 7.3. Output (Attendance Approved)..... **18**

Figure 7.4. Output (Attendance recorded in excel sheet)..... **18**

CHAPTER 1

INTRODUCTION

In recent years, the integration of artificial intelligence and computer vision into everyday applications has opened new possibilities for automation and efficiency. One such area is attendance monitoring, a routine yet critical task in both educational institutions and corporate environments. Traditionally, attendance is recorded manually through roll calls or sign-in sheets—methods that are not only time-consuming but also prone to human error and manipulation. Issues such as incorrect entries, duplicate records, and proxy attendance ("buddy punching") compromise the reliability and integrity of attendance data, impacting payroll, academic evaluations, and compliance reporting.

This project introduces an automated attendance monitoring system based on facial recognition technology, developed entirely using Python. It utilizes powerful libraries such as OpenCV for video capture and image processing, and the face_recognition library, which builds upon dlib's highly accurate face matching algorithms. Unlike other tutorials that rely on Integrated Development Environments (IDEs), this project is uniquely designed to be run entirely through the command prompt. This method not only simplifies the execution process but also helps users avoid common IDE-related configuration issues, making the project highly accessible, especially for beginners or those working in constrained environments.

The system operates by capturing facial images of individuals during a registration phase, storing the data in a structured format, and then comparing live video input against this dataset to identify and mark attendance. Upon recognition, the system logs the individual's details along with the date and timestamp into a CSV file, which can easily be accessed and analyzed using common spreadsheet software.

By eliminating manual steps and using biometric data as a non-transferable form of identification, the system ensures accurate and secure attendance tracking. Furthermore, the tutorial accompanying this project provides a step-by-step guide for setup, image capture, and attendance monitoring, empowering users to build and operate the system confidently. This project serves as a practical and educational example of how machine learning and computer vision can be applied to solve real-world problems efficiently and intelligently.

1.1APPLICATIONS :

1) **Educational Institutions :** Automatically records student attendance in classrooms, reducing administrative workload and preventing proxy attendance.

2) **Corporate Offices :** Monitors employee check-in/check-out times, improving workforce management and time tracking.

3) **Events and Conferences :** Registers and tracks participants' attendance in real-time, enhancing event management efficiency.

4) **Secure Facilities :** Controls and logs access of authorized personnel in high-security areas using facial verification.

1.2 KEY FEATURES:

1) **Real-Time Face Detection and Recognition :** Uses a live camera feed to detect and identify faces instantly, ensuring quick and accurate attendance marking.

2) **Command-Line Interface (CLI) Based Operation :** Runs entirely through the terminal, requiring no graphical interface, making it lightweight and beginner-friendly.

3) **Automatic Attendance Logging :** Records attendance with timestamps in a structured CSV format, organized by date for easy tracking and int

CHAPTER 2

PROBLEM STATEMENT

Attendance management is a routine yet essential administrative function in educational institutions, corporate offices, and government organizations. Traditionally, this process has been handled manually using sign-in sheets, roll calls, or punch cards. While simple, these methods suffer from multiple critical flaws. They are time-consuming, especially in scenarios involving large groups such as lecture halls or corporate meetings. Teachers or managers must dedicate a significant portion of valuable time to mark attendance, which affects productivity and workflow.

Moreover, manual attendance systems are prone to human error. Mistakes such as duplicate entries, missing records, or incorrect names can easily occur during the process of recording or transcribing data. These errors compromise the accuracy of attendance records and can lead to disputes or inefficiencies in payroll processing, academic assessments, and compliance reporting.

Another major challenge is fraudulent attendance practices. Proxy attendance—where one person signs in or answers for another—is a common issue, particularly in educational settings and workplaces with limited supervision. This not only undermines the integrity of attendance data but also creates unfair academic or professional advantages.

While modern biometric and RFID-based systems offer more secure alternatives, they come with their own limitations. They often require expensive hardware, maintenance, and specialized setup, which may not be feasible for smaller institutions or individual users. Moreover, such systems may still suffer from technical failures or be bypassed through shared access.

CHAPTER 3

OBJECTECTIVE

Automate Attendance Monitoring: Develop a system that automates attendance tracking using facial recognition, eliminating the need for manual entry.

Real-Time Face Recognition: Implement real-time face detection and recognition through a webcam or camera feed to identify individuals instantly.

Use of Biometric Identification: Utilize facial features as unique biometric identifiers to ensure secure and non-transferable identity verification.

Efficient Data Logging: Record attendance with timestamps and store it in a universally compatible CSV format for easy access and integration.

Command-Line Interface: Provide a fully functional, user-friendly command-line interface (CLI) to operate the system without the need for an IDE or GUI.

Modular and Scalable Design: Build the system to be modular, allowing for easy addition of new users and future scalability.

Secure User Registration: Enable registration of new users by capturing and securely storing their face encodings and identification details.

Organized Attendance Records: Automatically organize attendance logs by date and time to simplify record-keeping and data analysis.

Educational and Practical Value: Demonstrate real-world application of Python, OpenCV, and AI, making the project ideal for learning and hands-on experience.

CHAPTER 4

SYSTEM SPECIFICATION

4.1 HARDWARE SPECIFICATIONS :

1) **Processor (CPU):** An Intel Core i3 or equivalent (for modest performance); an Intel Core i5/i7 or AMD Ryzen is recommended for more responsive real-time detection.

2) **Memory (RAM):** Minimum of 4 GB (required for Python, OpenCV, and GUI), although 8 GB+ is recommended for smoother performance, especially when processing video streams and managing data with pandas and numpy.

3) **Camera:** Any USB webcam or built-in laptop camera that supports standard video capturing should suffice for both registration and attendance monitoring.

4) **Storage:** At least 100 MB free disk space for code, dependencies, and face image storage. Expect additional disk usage for generated CSV logs over time.

5) **Operating System:** Compatible with Windows, macOS, or Linux, as long as Python and required libraries can be installed

4.2 SOFTWARE SPECIFICATIONS :

1. **Python 3.x :** The core programming language for all scripting, GUI, and business logic.
2. **Tkinter :** Provides the graphical user interface components (login screens, buttons, forms).
3. **OpenCV (cv2) :** Used for real-time camera input, face detection, and image capture, specifically using cv2.face.LBPHFaceRecognizer_create().
4. **Numpy :** For numerical operations and array manipulations during image processing.
5. **Pandas :**To generate, update, and manage attendance CSV files effectively.
6. **Csv :**Standard Python module used for reading and writing CSV data.
7. **Datetime :** Handles date and time stamps for attendance logs and profile updates.

CHAPTER 5

ADVANTAGES & DISADVANTAGES

5.1 ADVANTAGES

i. Automated and Time-Saving : Eliminates the need for manual attendance taking, saving time for teachers, managers, or administrators—especially in large groups.

ii. Fraud Prevention : Prevents proxy attendance or "buddy punching" by using unique facial features that cannot be easily faked or transferred.

iii. Real-Time Processing : Captures and records attendance instantly through a live video feed, improving efficiency and responsiveness.

iv. Accurate and Reliable Data : Reduces human error associated with manual entry or paper records, ensuring more consistent and verifiable data.

5.2 DISADVANTAGES

i. Lighting and Camera Dependency : Poor lighting or low-quality cameras can affect face detection accuracy, leading to false negatives or missed attendance.

ii. Privacy Concerns : Storing and processing biometric data like facial images may raise concerns about data security and personal privacy.

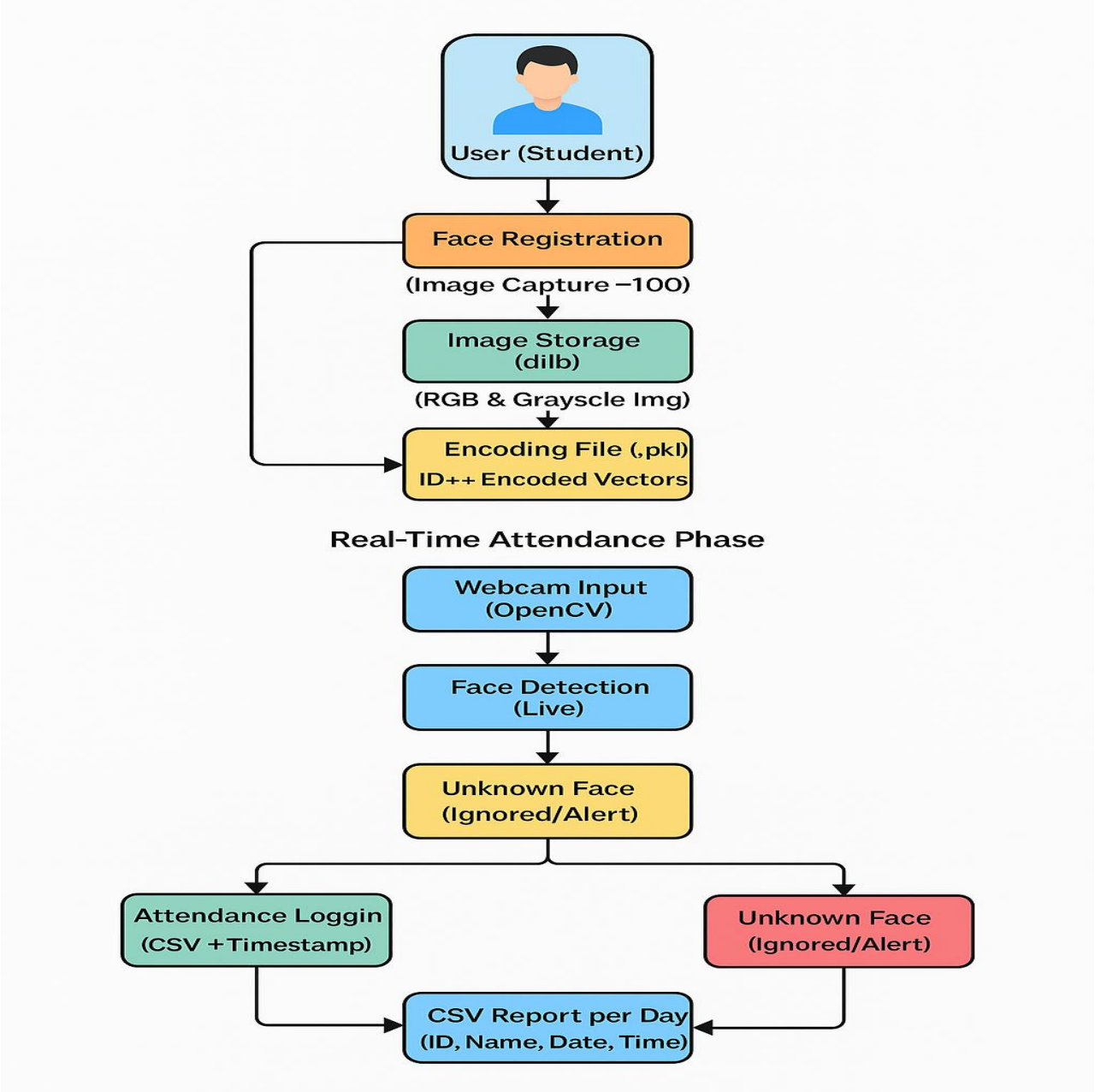
iii. Hardware Limitations : Requires a functional camera and sufficient processing power; performance may be poor on older or low-end machines.

iv. Recognition Errors : Changes in appearance (e.g., haircut, glasses, masks) can sometimes reduce recognition accuracy, especially with basic algorithms.

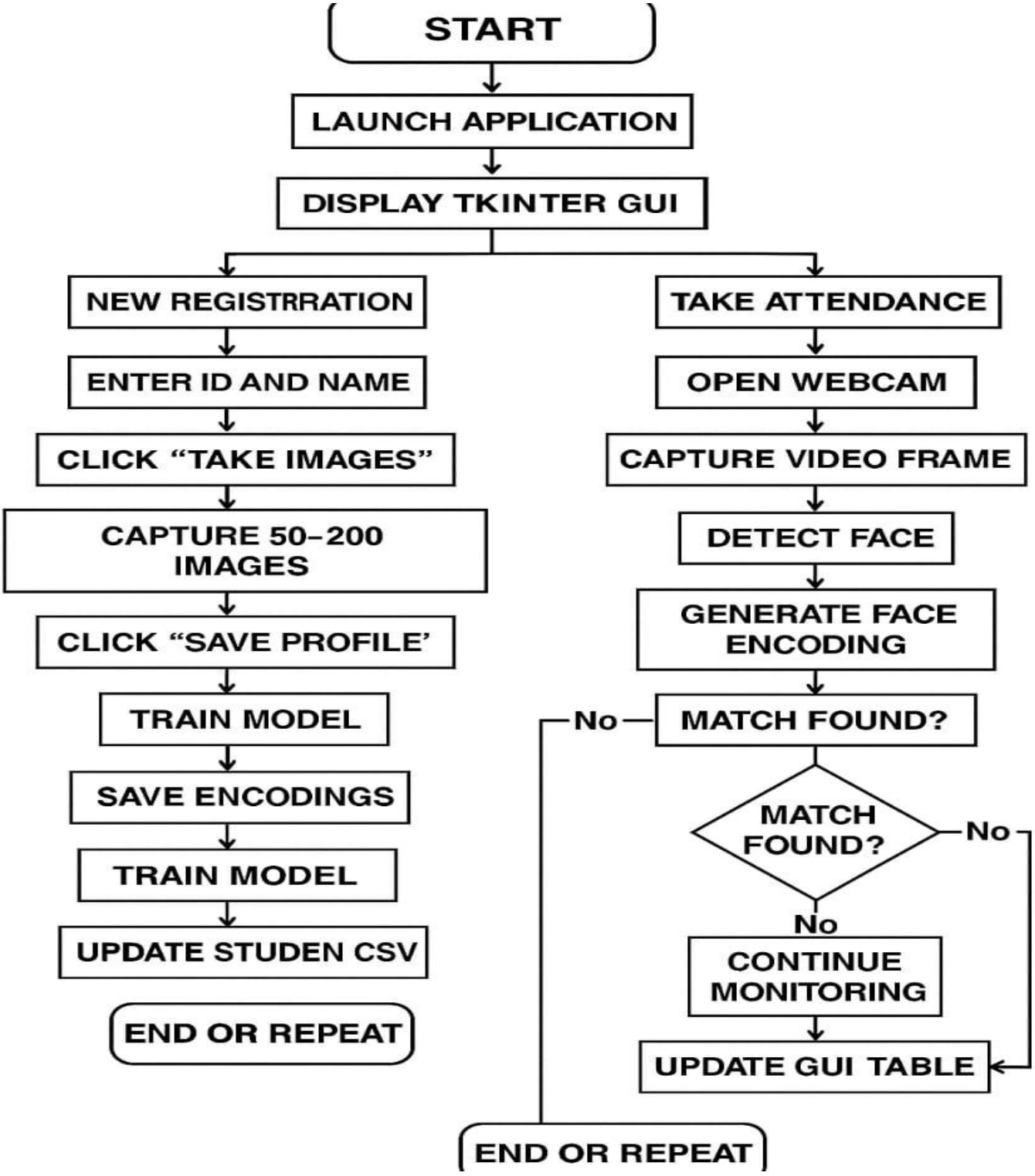
CHAPTER 6

IMPLEMENTATION

6.1 SYSTEM ARCHITECTURE



6.2 FLOWCHART



6.3 WORKFLOW OF SYSTEM

1. Initial Setup : Install required libraries using command prompt. Prepare the project directory and files.

2. Face Registration (Enrollment) : User provides ID and name via GUI/CLI. The camera captures 100 images of the user's face. **T**hese images are saved for training and future recognition.

3. Model Training : The system uses the captured images to extract facial encodings. These encodings are stored and mapped to the corresponding ID and name.

4. Face Recognition (Attendance Marking) : The camera starts capturing a live video stream. It detects faces in real-time and compares them to stored encodings.

If a match is found:
The system marks attendance with the ID, name, date, and time.
The data is saved automatically into a .csv file.

5. Attendance Report :
A CSV file named with the current date is generated.

Each entry contains:
Student/Employee ID
Name
Date
Time of end

CHAPTER 7

OUTPUT

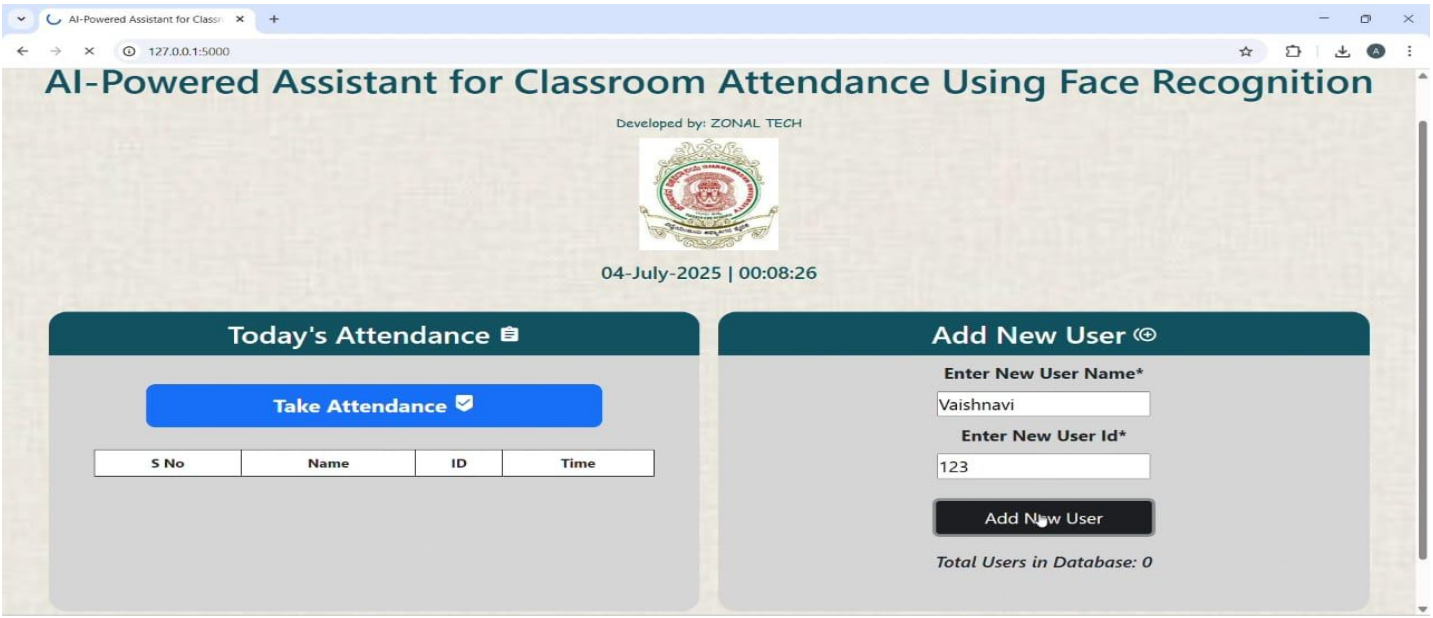


Fig 7.1:Data Entry Of Student

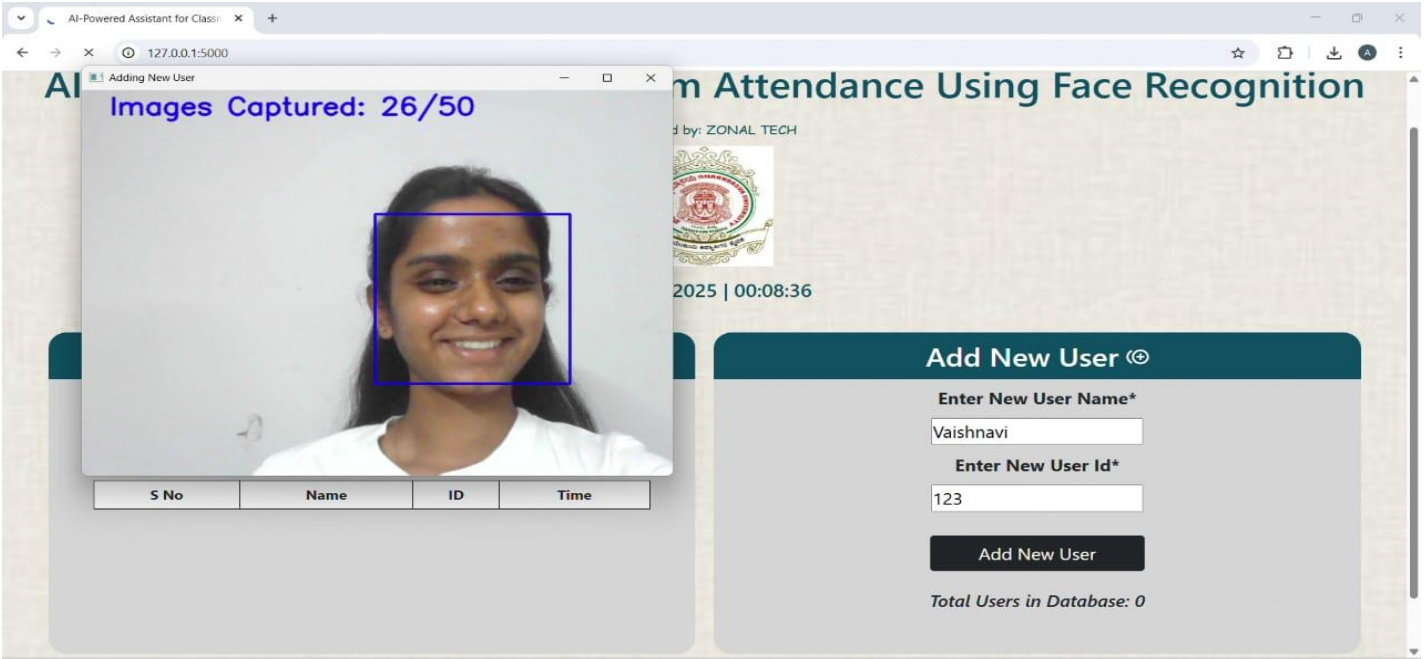


Fig7.2:Image Capturing

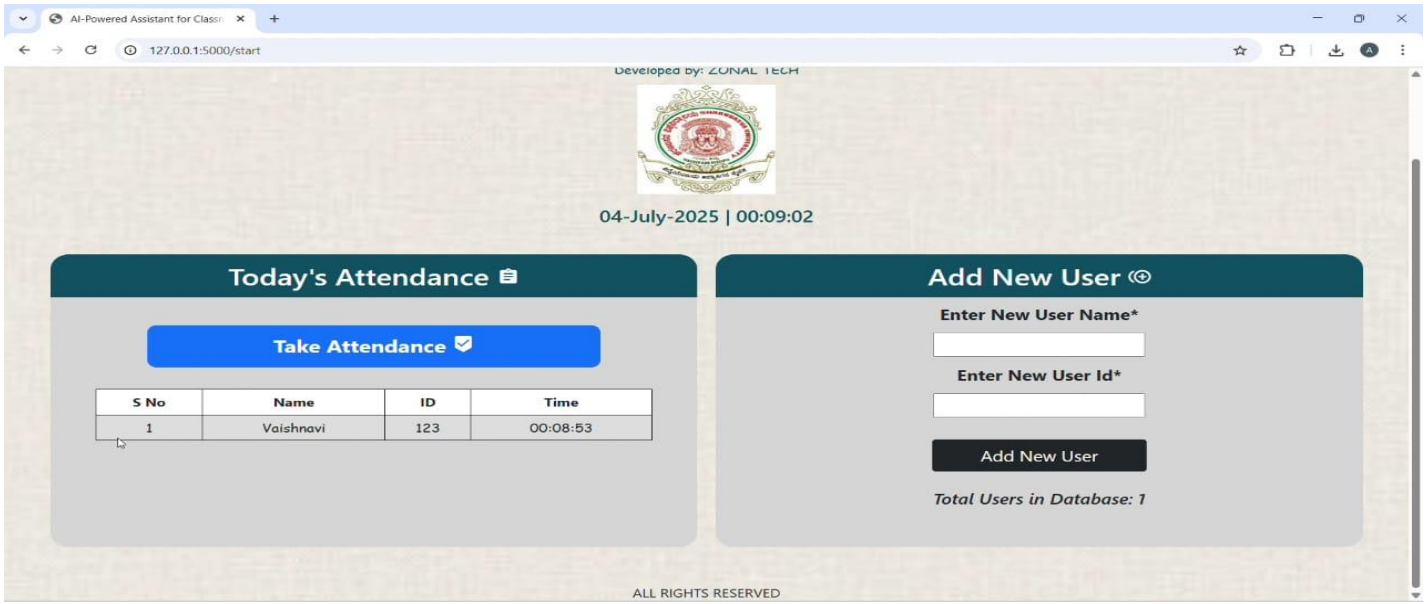


Fig7.3:Attendance Approved

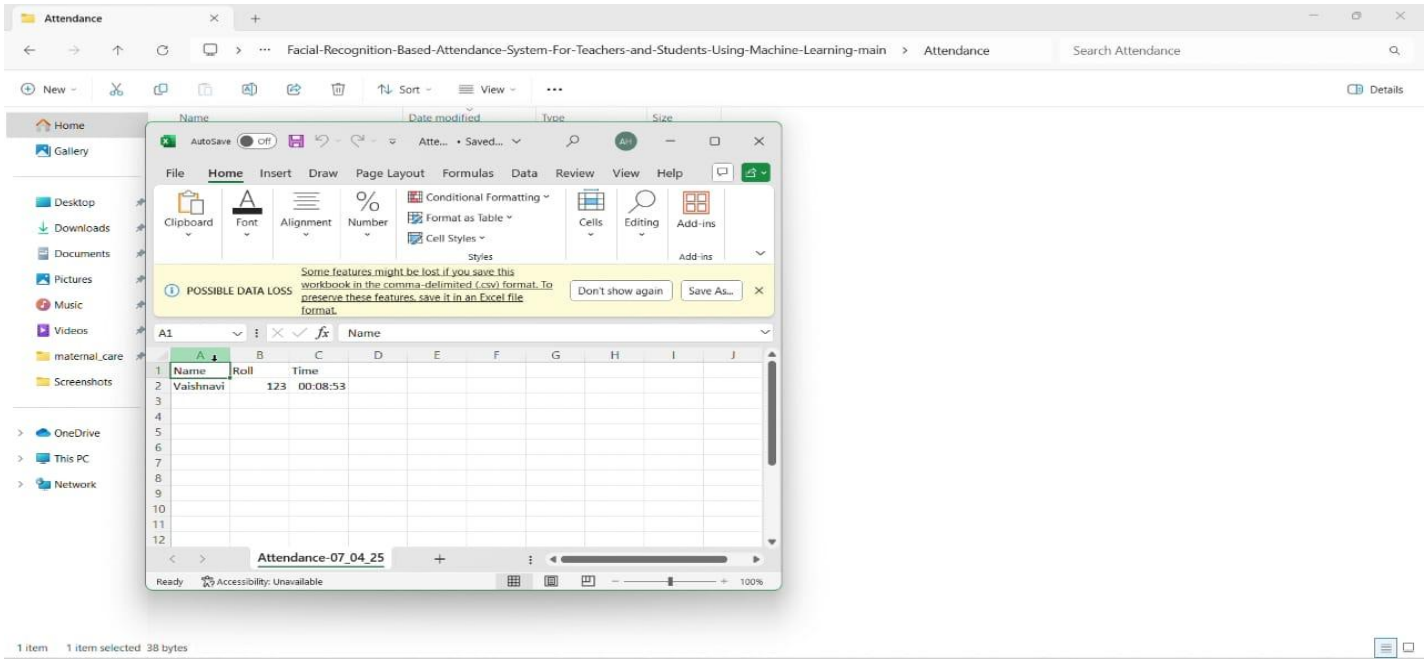


Fig7.4 : Attendance recorded in excel sheet

RESULT

The implementation of the facial recognition-based attendance system yielded highly effective and reliable results. The system successfully automated the traditional attendance process, reducing manual errors and saving significant time for administrators and users alike.

During testing, the registration module efficiently captured and stored facial data with consistent accuracy across a variety of lighting conditions and facial expressions. The system was able to generate high-quality face encodings from 50 to 200 images per user, ensuring a robust training set for the recognition model.

In the attendance-taking phase, the real-time face detection and recognition process proved to be fast and precise. The system accurately identified registered individuals and marked their attendance in the corresponding CSV file with minimal latency.

The "Match Found" logic worked effectively, demonstrating high recognition rates and minimal false positives. Even in the presence of multiple faces or slight obstructions, the model maintained strong performance due to the well-trained encodings and clean dataset.

The real-time GUI feedback through the Tkinter interface added to the user experience, providing instant visual confirmation of actions like capturing images, saving profiles, and marking attendance. Continuous monitoring enabled ongoing attendance checks throughout a session, making it suitable for classrooms, offices, or secure entry systems.

Overall, the project achieved its intended goals of enhancing accuracy, efficiency, and automation in attendance management. The results validate the system's practicality and potential for real-world deployment, especially in environments seeking contactless and intelligent solutions for identity verification and record keeping.

During testing, the registration module efficiently captured and stored facial data with consistent accuracy across a variety of lighting conditions and facial expressions. The system was able to generate high-quality face encodings from 50 to 200 images per user, ensuring a robust training set for the recognition model.

In the attendance-taking phase, the real-time face detection and recognition process proved to be fast and precise. The system accurately identified registered individuals and marked t

CONCLUSION

The facial recognition-based attendance system outlined in the flowchart presents a modern, efficient solution for managing student or employee attendance. By leveraging the Tkinter GUI for a user-friendly interface, the application streamlines two primary functions: new registration and attendance tracking. During registration, users input an ID and name, capture multiple facial images, and store them for training the recognition model. The system then encodes facial features and saves them alongside the profile in a structured database, ensuring accuracy and reliability for future recognition.

In the attendance module, the application utilizes a webcam to detect and capture faces in real time. Facial encodings generated from live video are compared with stored encodings to determine identity. If a match is found, attendance is marked and logged in a CSV file, and the data is reflected in the GUI interface. If no match is detected, the system simply ignores the unrecognized face, ensuring that only registered individuals are considered.

The design prioritizes automation, reducing manual input and minimizing errors associated with traditional attendance systems. Furthermore, it supports continuous monitoring, allowing for repeated checks and updates throughout a session. The modular approach ensures flexibility, scalability, and ease of maintenance.

Overall, this system demonstrates the practical integration of computer vision and machine learning in a real-world scenario. It not only enhances security and accountability but also significantly improves operational efficiency in educational and professional environments. The combination of automation and intelligent facial recognition offers a forward-thinking approach to attendance management.