

Intelliface

A Face Recognition based attendance system

Presented by -

Jagruti Piprade

(202151067)

Khushi Saxena

(202151078)

Ideation process

KNN Classifier

we investigate face recognition with the adaptable K Nearest Neighbors (KNN) method. KNN classifies incoming data according to how similar it is to preexisting datasets. We actively develop and evaluate the KNN for face recognition and the Haarcascade for face detection.

Convolutional neural networks

The necessity for improved accuracy drove us to use an empirical technique and, as a backup, the CNN ResNet model.

Key Features

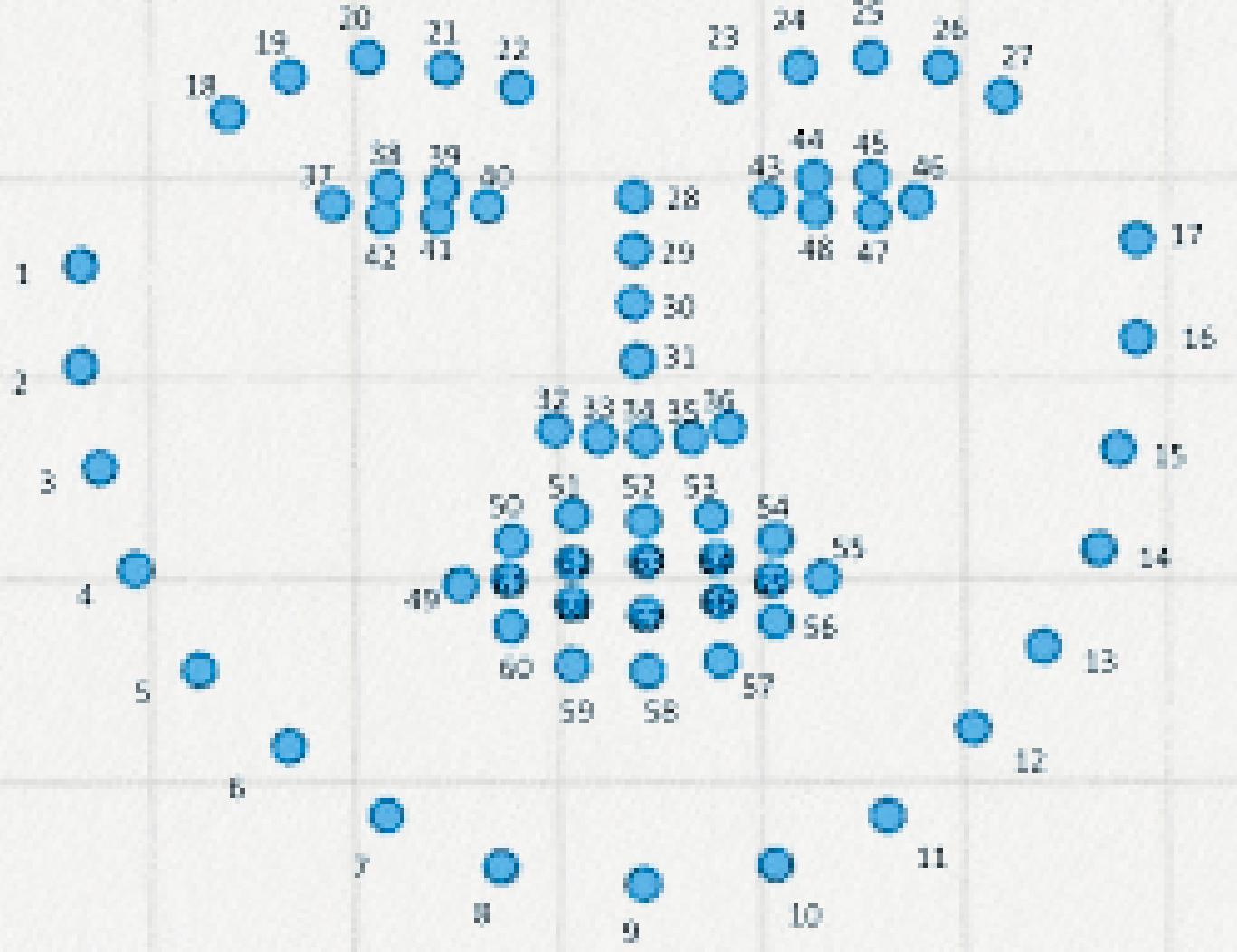
1 **Adaptability**

2 **Real-time tracking**

3 **User-friendly Interface**

4 **Identify unknown faces**

Shape predictor 68 face landmarks



"Shape predictor with 68 face landmarks" usually describes a particular kind of facial landmark detection model that recognizes 68 important points on a face. These points line up with different aspects of the face, like the mouth, nose, eyes, and facial contour.

The choice of 68 facial landmarks is a convention that stems from the specific model provided by the Dlib library for facial landmark detection. These 68 points are strategically placed to capture key features of the face, including the eyes, nose, mouth, and the contour of the face.

W O R K F L O W

01

Initialize essential libraries like OpenCV, Tkinter, Numpy, Pandas, and Dlib. OpenCV enables image processing, Numpy handles numerical operations, Pandas manages data structures, Dlib for face detection and Tkinter for GUI development.

02

OpenCV is employed to capture the video feed, extracting frames as the foundation for subsequent face identification. OpenCV provides robust functionalities for video processing, enabling efficient frame extraction and manipulation.

W O R K F L O W

03

Face Identification: Dlib's face detection capabilities are employed to accurately identify faces within the frames obtained from the video feed captured by OpenCV.

04

Implement GUI interaction through Tkinter, which is a Python library for creating graphical user interfaces (GUIs). Tkinter is utilized to design and create the user interface where users can input their names for attendance in the face recognition project.

05

SQLite, a lightweight relational database management system is used to interact with the database. The integration of SQLite with Python facilitates efficient storage of facial features in a structured manner for subsequent retrieval and analysis.

06

The facial features are extracted from images and systematically stored in a CSV file for record-keeping. The combination of Pandas and Numpy facilitates organized storage of facial features in a CSV format.

W O R K F L O W

07

Utilize the recognition module for face detection, ensuring real-time updating of attendance records on the website. The flask functions as a web framework, creating a user interface for the face recognition system. It handles the communication between the back end (OpenCV and Dlib for face detection) and the front end (project website). This integration allows real-time updating of attendance records on the website when faces are recognized.

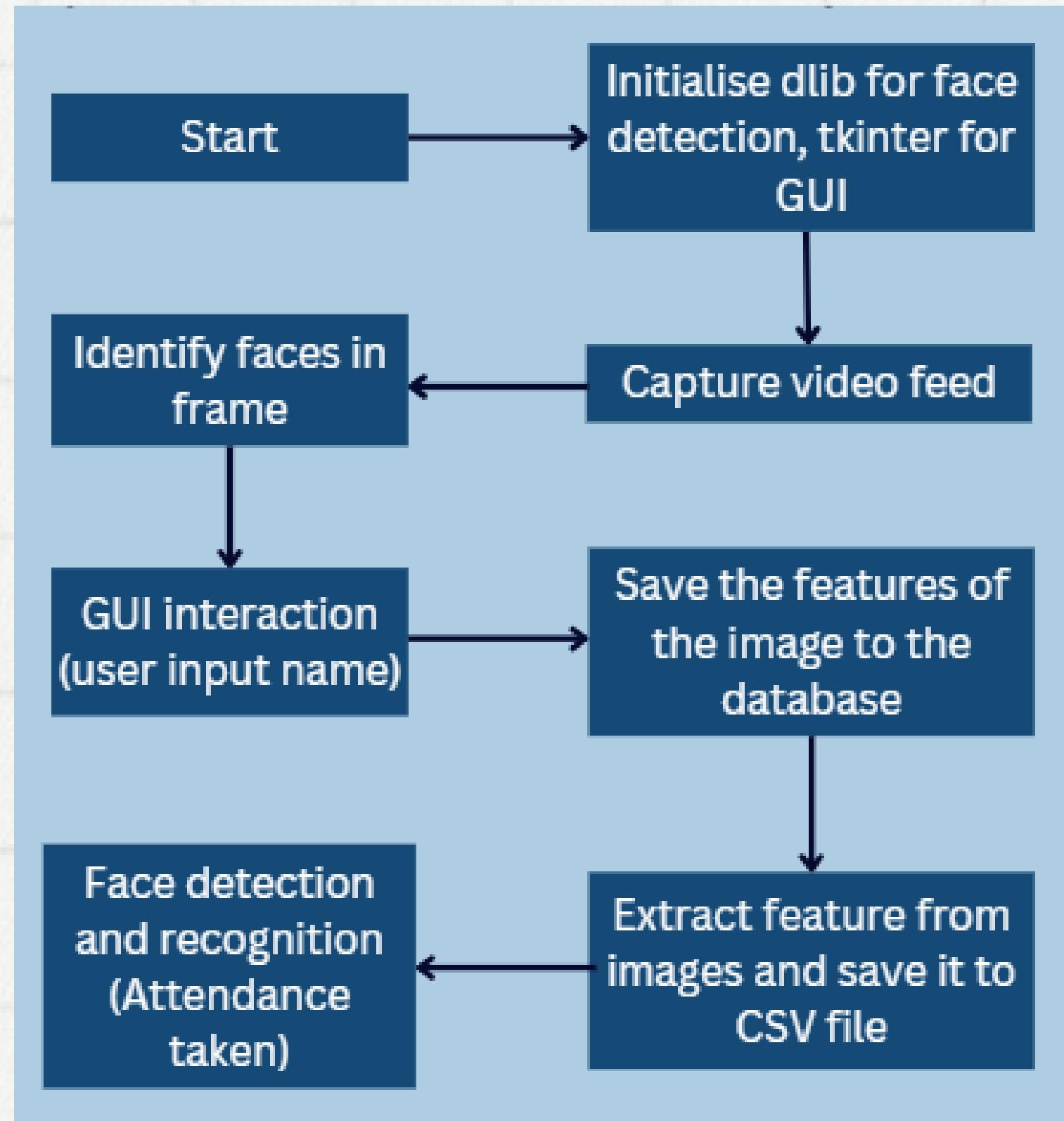
W O R K F L O W



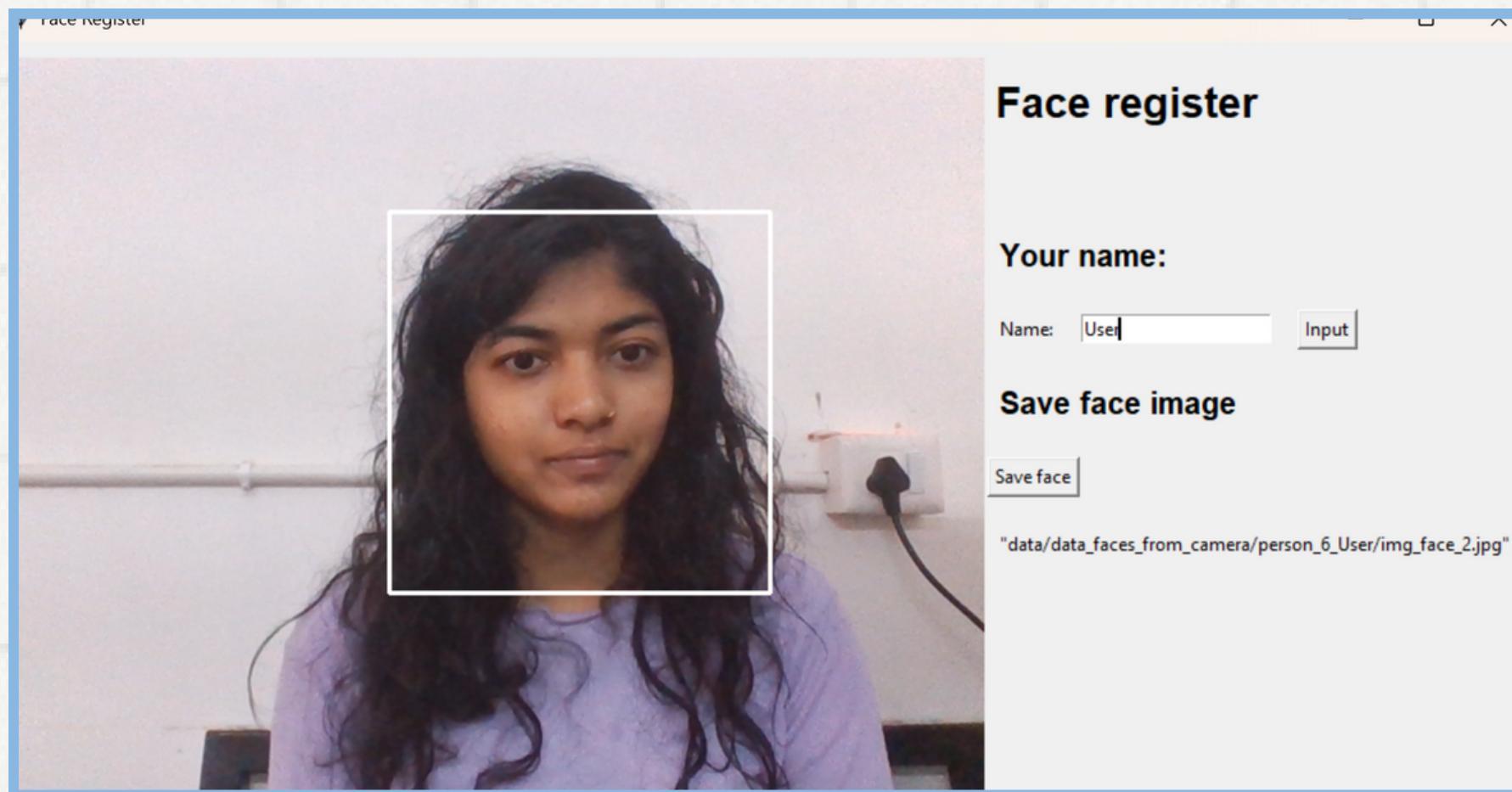
08

Website Access for Students: A user-friendly website interface is developed to provide students with convenient access to their attendance records. HTML structures website content, CSS styles its appearance, and Flask integrates with Python for back-end logic, interacting with the database (SQLite).

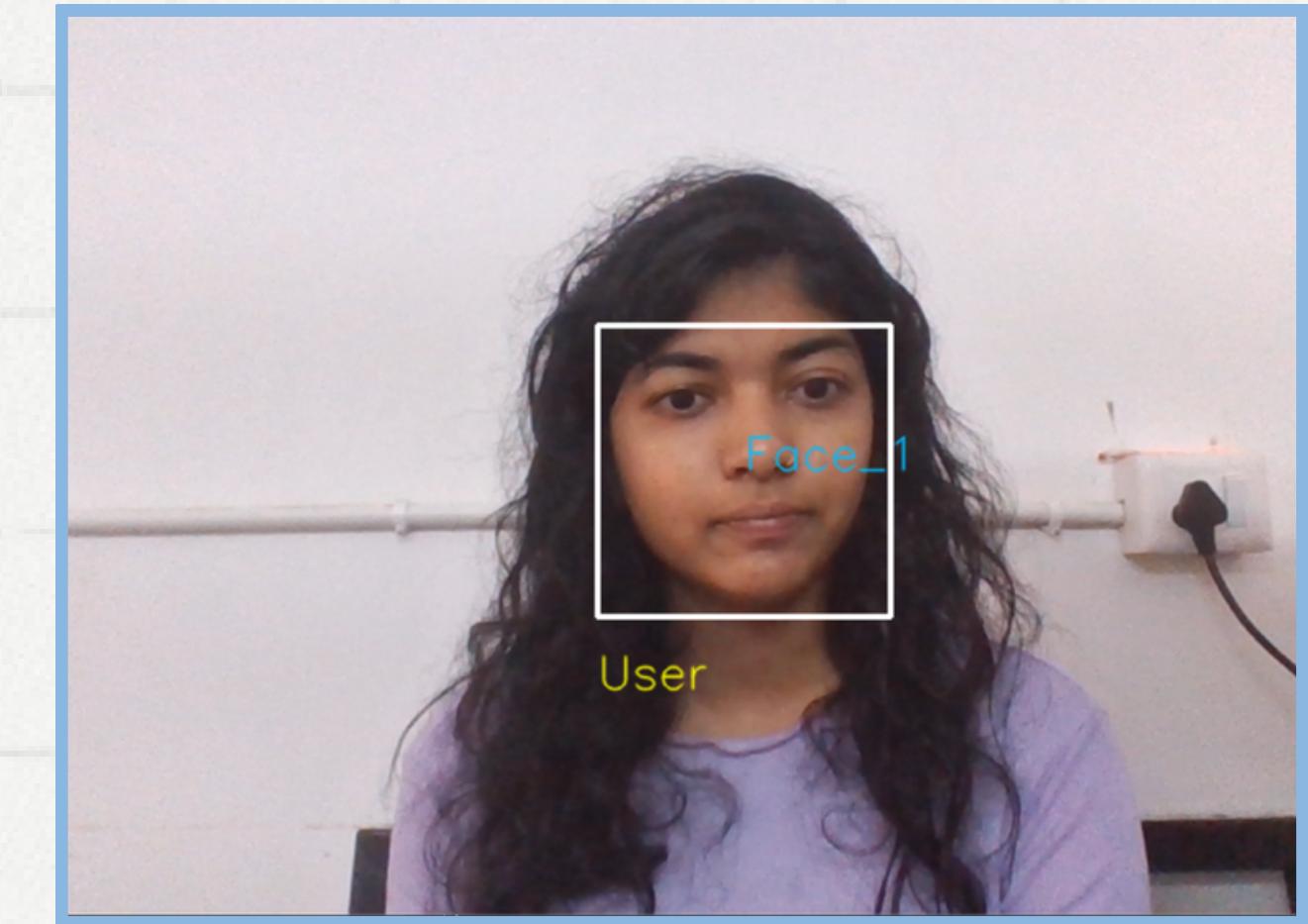
Workflow



Implementation

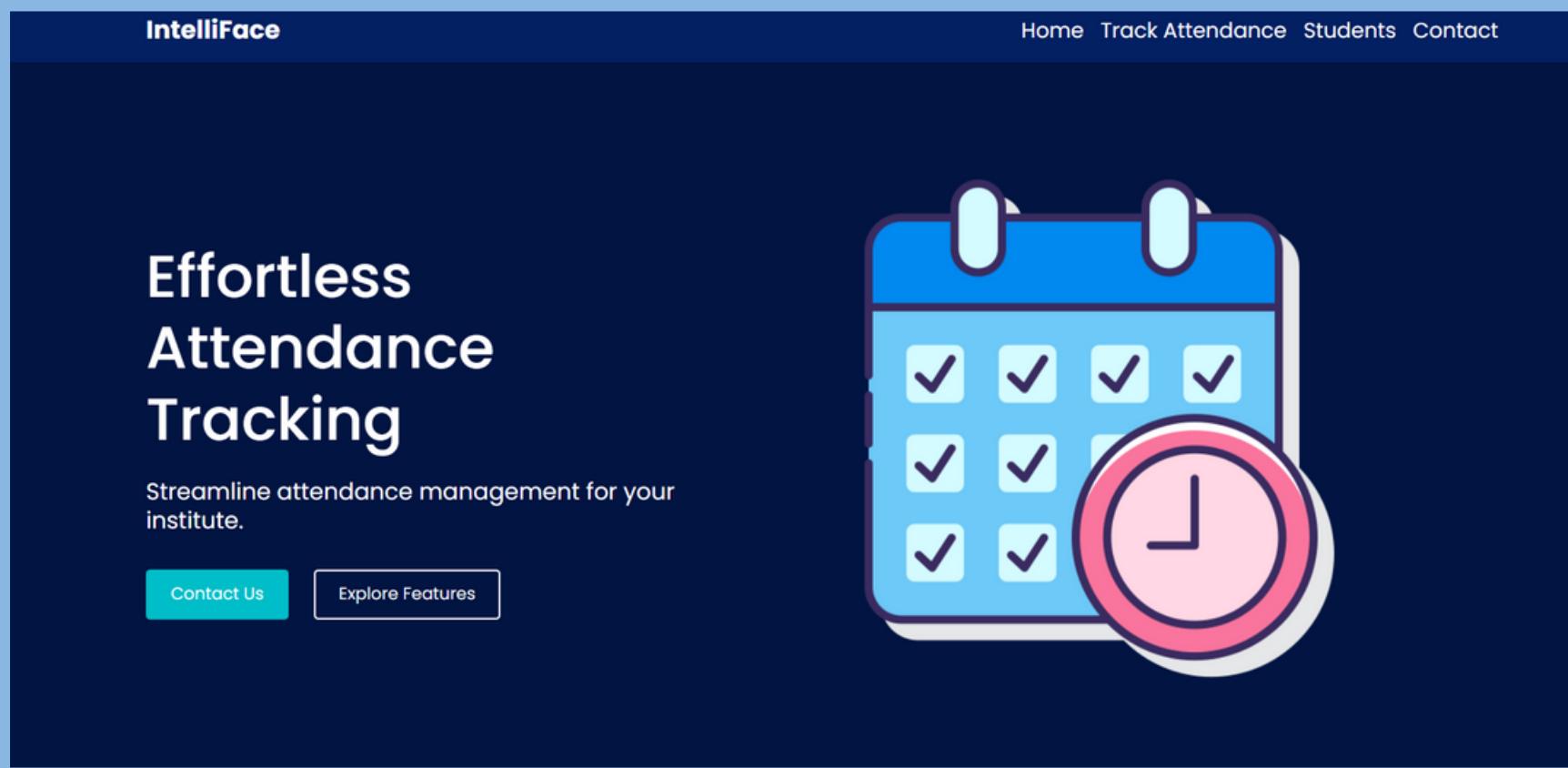


Face Registration



Face Recognition

Implementation



TRACK ATTENDANCE

Select Date:

07-12-2023

Show attendance

Attendance Data Table

Name	Time
Jagruti	00:18:41
Gargee	00:18:58
Khushi	00:19:11
Srivalli	00:19:27
Meenakshi	00:51:53
User	10:20:46

Attendance tracker web application

Final Reflections

- The integration of a ResNet model, Dlib, NumPy, and OpenCV in the Face Recognition-Based Attendance System signifies a notable advancement in attendance tracking technology.
- The system's interface and real-time efficacy position it as a pragmatic tool for educational and corporate environment



What next?

Prevent Frauds

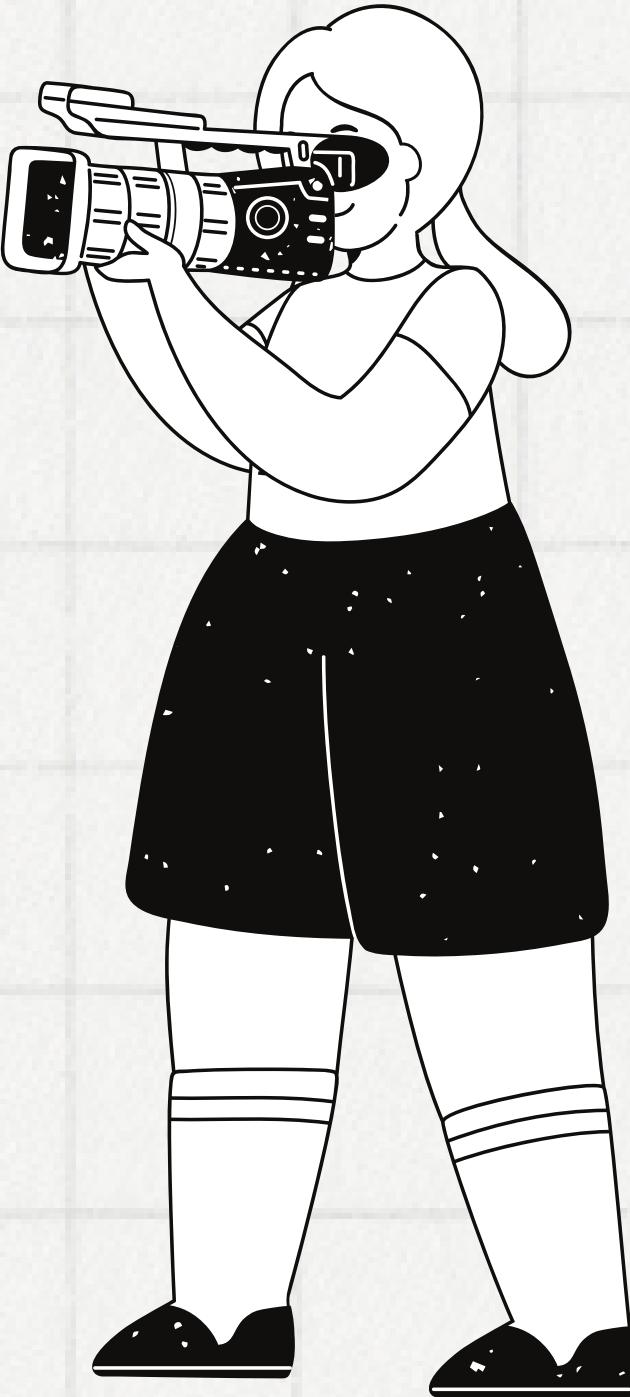
Integrating anti-spoofing methods to prevent photo-based attendance fraud.

Security

Introducing iris detection for increased security.

Addition of more features

introducing automated notifications and reporting features, streamlining administrative tasks through real-time alerts



**Thank you
very much!**