PRACTICAL NO.: 2

Aim:

To state and analyse the problem definition for the Mini Project.

Project Title:

Face Recognition Based Attendance System

---

1. Introduction

In modern educational institutions and workplaces, attendance tracking is an essential administrative task. Traditional methods such as manual registers, RFID cards, or fingerprint scanners often suffer from inefficiencies, inaccuracies, and security concerns. To address these challenges, we propose a **Face Recognition-Based Attendance System** that leverages **Artificial Intelligence (AI) and Machine Learning (ML)** to automate and streamline attendance management.

This system utilizes computer vision techniques to recognize and verify individuals based on their facial features. By integrating **Machine Learning (ML) modal and libraries**, our solution ensures a contactless, secure, and efficient attendance tracking process. The model is trained using a dataset of facial images and can accurately identify students or employees even in varied lighting and environmental conditions.

2. Objectives

The main objectives of the Face Recognition Based Attendance System:

- **Automate Attendance Marking**
    - Develop a system that eliminates manual attendance processes and ensures seamless, contactless attendance tracking.
- **Enhance Accuracy & Efficiency**
    - Utilize AI/ML-based face recognition to reduce human errors and improve attendance accuracy.
- **Real-Time Face Detection & Recognition**
    - Implement real-time facial recognition using deep learning models to identify individuals quickly and accurately.
- **Improve Security & Prevent Proxy Attendance**
    - Integrate anti-spoofing techniques to prevent fraudulent attendance marking using photos or videos.
- **Develop a User-Friendly System**

    - Ensure the system has an intuitive interface for easy access by students, employees, and administrators.

- **Enable Multi-User & Scalable Deployment**
    - Design the system to handle multiple users efficiently and allow integration with existing attendance databases.
- **Optimize for Different Environments**
    - Ensure the system functions well in different lighting conditions, angles, and real-world scenarios.
- **Integrate with Attendance Records & Reports**
    - Provide automatic attendance logs, real-time data storage, and generate reports for administrators.
- **Support Various Camera Options**
    - Allow integration with CCTV, IP cameras, webcams, or mobile devices to capture images effectively.
- **Enhance Privacy & Data Security**
    - Implement encryption and secure storage techniques to protect biometric data and user identities.

3. Advantages

Advantages of the Face Recognition-Based Attendance System

- **Automation & Time-Saving**
    - Eliminates the need for manual attendance marking, reducing administrative workload and saving time.
- **High Accuracy & Reliability**
    - AI/ML-based face recognition ensures precise identification, minimizing errors compared to traditional methods.
- **Contactless & Hygienic**
    - Unlike fingerprint scanners or ID cards, face recognition is touch-free, reducing the risk of spreading infections.
- **Prevents Proxy Attendance & Fraud**
    - Eliminates the chances of students or employees marking attendance on behalf of others (buddy punching).
- **Real-Time Processing**

- Instantly detects and recognizes faces, allowing quick attendance marking without delays.

- **User-Friendly & Convenient**
  - Requires no physical interaction—users simply need to appear in front of the camera for automatic attendance.
- **Scalability & Multi-User Support**
  - Can handle multiple users efficiently, making it ideal for classrooms, offices, and large organizations.
- **Seamless Integration with Existing Systems**
  - Can be integrated with databases, HR systems, and school/university attendance records.
- **Reduces Costs in the Long Run**
  - Eliminates the need for RFID cards, biometric scanners, or manual registers, reducing operational costs over time.
- **Works in Various Environments**
  - Can be deployed using CCTV, IP cameras, or webcams, ensuring flexibility in different locations.
- **Enhances Security & Data Protection**
  - Uses encryption and secure storage methods to protect biometric data and prevent unauthorized access.
- **Generates Automated Reports & Analytics**
  - Provides real-time attendance logs, summary reports, and analytics for better monitoring and decision-making**.**

4.Conclusion

The **Face Recognition-Based Attendance System** is a modern, AI-driven solution that automates attendance tracking efficiently and securely. By leveraging **machine learning and computer vision**, this system eliminates the limitations of traditional attendance methods, such as manual registers, RFID cards, or fingerprint scanners.

With its **high accuracy, real-time processing, and contactless operation**, the system enhances convenience for both users and administrators. Additionally, it prevents **proxy attendance**, improves **security**, and seamlessly integrates with existing attendance management systems.

This project demonstrates the power of **AI and ML in real-world applications**, showcasing how technology can optimize daily operations in educational institutions, workplaces, and

other environments. By ensuring **scalability, flexibility, and data security**, the system can be effectively deployed across various sectors, making attendance tracking more reliable and efficient.

Roll No.: 12,16,17

PRACTICAL NO.: 3

Aim:

Design Software Requirement Specification document for Mini Project.

Project Title:

AI-Based Face Recognition Attendance System

SOFTWARE REQUIREMENT SPECIFICATION (SRS)

1. Introduction

1.1 Purpose

The AI-Based Face Recognition Attendance System aims to automate and streamline the attendance-taking process by leveraging facial recognition technology. This system eliminates the need for traditional manual attendance methods such as roll calls or RFID-based systems. It provides an efficient, accurate, and tamper-proof solution suitable for educational institutions, workplaces, and other organizations requiring attendance tracking.

1.2 Scope

The system captures and processes facial images of individuals to mark their attendance. It utilizes AI and machine learning algorithms to recognize and authenticate faces in real-time. The system is designed for:

- Educational institutions to track student attendance.
- Corporate offices to manage employee attendance.
- Secure access control in restricted areas.
- Any organization that requires an automated attendance system.

1.3 Definitions, Acronyms, and Abbreviations

- **AI (Artificial Intelligence):** Simulation of human intelligence in machines.
- **ML (Machine Learning):** A subset of AI that enables systems to learn from data.
- **Face Recognition:** The process of identifying or verifying individuals using face.
- **Database:** A structured system for storing and retrieving attendance records.

1.4 References

- IEEE Software Engineering Standards.
- Research papers on AI-based facial recognition.

- Machine learning frameworks documentation (e.g., OpenCV, TensorFlow, Keras).

---

2. General Description

2.1 Product Perspective

The system is a standalone application with integration capabilities for various platforms. It consists of a front-end user interface for administration and a backend AI model for facial recognition. It requires a camera-enabled device for capturing images and a database for storing attendance records.

2.2 Product Functions

- Capture and store facial images in a secure database.
- Real-time face detection and recognition.
- Mark attendance automatically upon successful recognition.
- Generate reports and analytics for administrators.
- Allow manual correction in case of recognition failures.

2.3 User Characteristics

The system is designed for:

- **Administrators:** Responsible for managing the database and overseeing system operations.
- **Employees/Students:** End-users whose attendance is recorded using facial recognition.
- **IT Staff:** Ensures system maintenance, data security, and model accuracy.

2.4 Constraints

- Requires a well-lit environment for accurate face detection.
- Dependence on a high-quality camera for precise image capturing.
- Internet or network dependency for cloud-based implementations.

2.5 Assumptions and Dependencies

- Users must have their faces enrolled in the system beforehand.
- The AI model should be trained periodically to enhance recognition accuracy.

- The system should be deployed on devices with sufficient processing power.

---

3. Functional Requirements

3.1 User Authentication

- The system shall authenticate users based on pre-registered facial images.
- It shall prevent unauthorized access by verifying identity before marking attendance.

3.2 Face Detection and Recognition

- The system shall detect and recognize faces in real-time.
- It shall differentiate between multiple users and recognize them individually.
- The recognition process shall be completed within a few seconds.

3.3 Attendance Marking

- The system shall automatically mark attendance when a face is successfully recognized.
- It shall record the timestamp of attendance for each individual.
- Duplicate entries shall be prevented within a specified time frame.

3.4 Database Management

- The system shall store all attendance records securely in a database.
- It shall allow retrieval and export of attendance data.
- The database shall support periodic backups.

3.5 Reporting and Analytics

- The system shall generate daily, weekly, and monthly attendance reports.
- It shall allow filtering of data based on specific users or time periods.
- The reports shall be exportable in formats such as CSV or PDF.

3.6 Manual Override

- Administrators shall be able to manually update attendance records in case of false negatives.
- The system shall log manual modifications for audit purposes.

3.7 Security and Privacy

- The system shall encrypt stored facial images and attendance data.
- It shall comply with data protection regulations to ensure user privacy.
- Access to the system shall be role-based, restricting unauthorized modifications.

---

4. Interface Requirements

The system will need to communicate with several components, both software and hardware. These interactions can include the following:

4.1. **Software Interfaces:**

- **Face Recognition API:** The system will interface with a machine learning model (e.g., OpenCV or a deep learning-based model) to process live CCTV footage for facial recognition. The software interface will include API calls for feeding the video frames, processing them, and receiving the results, such as attendance data.
- **Database Interface:** The system will interface with a SQL or NoSQL database to store and retrieve student attendance data, including time-stamped records for each student's attendance status.
- **User Interface:** A web-based or desktop user interface will allow administrators to monitor the status, view attendance reports, and make adjustments (if necessary).

4.2. **Hardware Interfaces:**

- **CCTV Cameras:** The system will need to interface with CCTV cameras through streaming protocols such as RTSP or direct camera integration, depending on the hardware.
- **Server or Local System:** The system will operate on servers or local machines where processing and storage will take place. The interface will include communication between the software and hardware to capture and process video footage in real-time.

---

5. Performance Requirements

5.1. **Static Requirements:**

- The system should be able to process real-time video feeds from CCTV cameras without major lag or delay.

- The system must be capable of handling a minimum of 10 CCTV camera feeds simultaneously with no degradation in performance.
- The system should support storing at least 3 months of attendance data in the database with minimal access time.

5.2. **Dynamic Requirements:**

- **Processing Speed:** Each frame of video should be processed within 1-2 seconds to ensure real-time attendance taking. Any delay beyond this limit could result in inaccurate attendance data.
- **Accuracy of Facial Recognition:** The model should have an accuracy of at least 95% in recognizing students from live footage under varying lighting and angles.
- **Memory Usage:** The system should use a minimum of 4 GB RAM for processing video feeds and storing attendance data in real-time.
- **Error Rate:** The system should have an error rate of less than 2% in terms of incorrectly marked attendance (false positives/negatives).

---

6. Design Constraints

6.1. **Hardware Constraints:**

- The system will require high-performance hardware for running the machine learning models efficiently, particularly GPUs for faster facial recognition processing. The system should be optimized for use with standard server-grade or high-end personal computers.
- The CCTV cameras must be capable of streaming high-resolution video (at least 1080p) in real-time.

6.2. **Software Constraints:**

- The system must be implemented using Python due to the availability of well-supported libraries for machine learning and image processing (e.g., TensorFlow, OpenCV).
- The facial recognition algorithm should rely on open-source libraries or pre-trained models like OpenCV or Dlib to minimize cost and development time.
- The user interface should be web-based, compatible with common browsers (Chrome, Firefox, etc.), and support basic functionality such as login, monitoring, and reporting.

6.3. **Environmental Constraints:**

- The system must be operable in typical college environments with standard Wi-Fi or LAN networks and minimal internet dependency.
- The system must be robust to changes in lighting conditions, varying student orientations, and partial occlusions.

---

7. Non-Functional Attributes

The AI-Based Face Recognition Attendance System must adhere to several non-functional attributes to ensure its effectiveness and efficiency:

- **Reliability:** The system must have an uptime of at least 99.5% to ensure continuous availability, reducing downtime and improving user trust.
- **Scalability:** The system should support multiple institutions, each with its own database, ensuring flexibility and growth. As user demand increases, the system should be capable of handling additional workload without performance degradation.
- **Usability:** The user interface should be intuitive, requiring minimal training for administrators and end-users. The system should provide a seamless experience across different devices and screen sizes.
- **Maintainability:** The system should be designed to allow easy updates and patches without disrupting ongoing attendance tracking. A structured approach to software maintenance should be in place to ensure long-term sustainability.
- **Security:** All data transmissions and storage must be encrypted, complying with data privacy regulations. Robust authentication mechanisms should be implemented to prevent unauthorized access.
- **Performance:** The system should process attendance in real-time, ensuring responses within 1-2 seconds. High processing speed is crucial to prevent bottlenecks and delays in attendance tracking.

---

8. Preliminary Schedule and Budget

8.1.Preliminary Schedule

The estimated timeline for the development and deployment of the system is as follows:

- **Requirements Gathering (1 Month):** Identify system specifications and user needs. Conduct surveys and interviews with stakeholders.

- **System Design (1 Month):** Develop architecture, UI, and database design. Define system workflows and infrastructure requirements.
- **Development (2 Months):** Implement core functionalities, including facial recognition and database integration. Perform initial testing.
- **Testing & Validation (1 Month):** Perform unit testing, system testing, and user acceptance testing. Identify and fix bugs to ensure smooth operation.
- **Deployment & Training (1 Month):** Deploy system and train administrators. Provide documentation and user manuals for easy onboarding.

8.2.Preliminary Budget

The estimated budget for the development and deployment of the system is as follows:

- **Software Development:** ₹12,50,000
- **Hardware (CCTV Cameras, Servers):** ₹8,30,000
- **Cloud Services (Database & Hosting):** ₹4,15,000/year
- **Miscellaneous Expenses:** ₹1,65,000
- **Maintenance & Support:** ₹1,00,000/year
- **Total Estimated Cost: ₹27,60,000**

---

9. Appendices

9.1. Facial Recognition Accuracy Factors

The accuracy of the facial recognition model depends on:

- Lighting conditions, which affect image clarity and recognition precision.
- Camera resolution and quality, ensuring higher accuracy in facial detection.
- The diversity of training data to reduce bias and improve recognition across different demographics.
- Angle and orientation of faces, as extreme angles can reduce detection reliability.

9.2. Legal and Ethical Considerations

- Compliance with data protection regulations such as GDPR and local privacy laws.
- Ensuring user consent before capturing facial images to maintain ethical standards.

- Secure storage and encryption of biometric data to prevent unauthorized access and data breaches.
- Regular audits to ensure adherence to security and compliance guidelines.

9.3. System Deployment Diagram

A high-level deployment diagram includes:

- Frontend interface (web-based or desktop application), providing an intuitive user experience.
- Backend AI engine processing live camera feeds, ensuring accurate facial recognition.
- Database storing attendance records, maintaining a secure and scalable repository.
- Cloud-based or on-premises server for computation and storage, depending on institutional preferences and budget constraints.

PRACTICAL NO.: 4

Aim:

Design Data Flow- Diagram for Mini Project.

Project Title:

AI Based Face Recognition Attendance System

Group details:

Group no. 3

Members:

12 – Sejal Wadibhasme

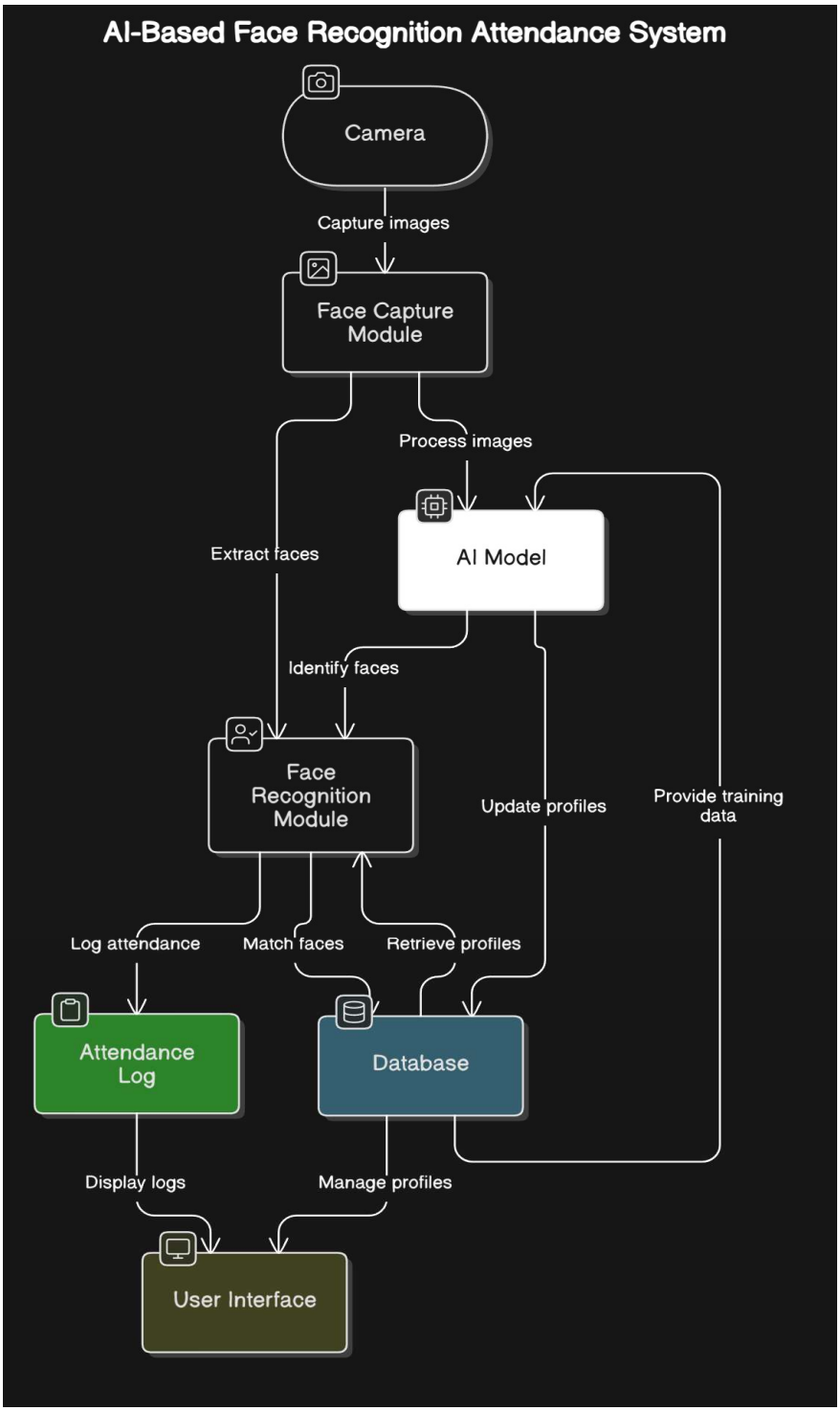
16 – Aayush Bisht

17 – Aditya Jaiswal

Entities/Modules/Functions/Data Types Used:

1. **Face Capture Module:**
   - **Entity**: Camera
   - **Function**: Captures real-time images or video frames of individuals in the classroom or work environment.
   - **Explanation**: The face capture module is responsible for obtaining a clear image of each individual in the frame, which will then be processed by the recognition algorithm.
2. **Face Recognition Module:**
   - **Entity**: AI Model (Pre-trained CNN, or any other face recognition model)
   - **Function**: Recognizes the face from the captured image and compares it with stored face data.
   - **Explanation**: The recognition system uses machine learning models to identify or match faces based on previously stored data, typically in the form of face embeddings.
3. **Database/Storage:**
   - **Entity**: Database (SQL/NoSQL, such as MySQL, MongoDB)
   - **Function**: Stores face data, student or employee profiles, attendance logs.
   - **Explanation**: The database stores images, unique identifiers for individuals, and their attendance logs. It acts as a data repository for the system.

4. **Attendance Log:**
    - **Entity**: Attendance records
    - **Function**: Logs the time and identity of the recognized individual.
    - **Explanation**: The attendance log records each successful face recognition event, including the timestamp and user details, ensuring accurate attendance tracking.
5. **User Interface:**
    - **Entity**: UI (Admin or Teacher Interface)
    - **Function**: Interface for viewing attendance data, managing profiles, and reviewing logs.
    - **Explanation**: The user interface allows administrators, teachers, or managers to interact with the system, manage attendance records, and handle system settings.


AI-Based Face Recognition Attendance System
Camera
Capture images
Face Capture Module
Process images
Extract faces
AI Model
Identify faces
Face Recognition Module
Update profiles
Provide training data
Log attendance
Match faces
Retrieve profiles
Attendance Log
Database
Display logs
Manage profiles
User Interface

PRACTICAL NO.: 5

Aim:

Design Entity Relationship - Diagram for Mini Project.

Project Title:

AI Based Face Recognition Attendance System

Group details:

Group no. 3

Members:

12 – Sejal Wadibhasme

16 – Aayush Bisht

17 – Aditya Jaiswal

Entities and Attributes:

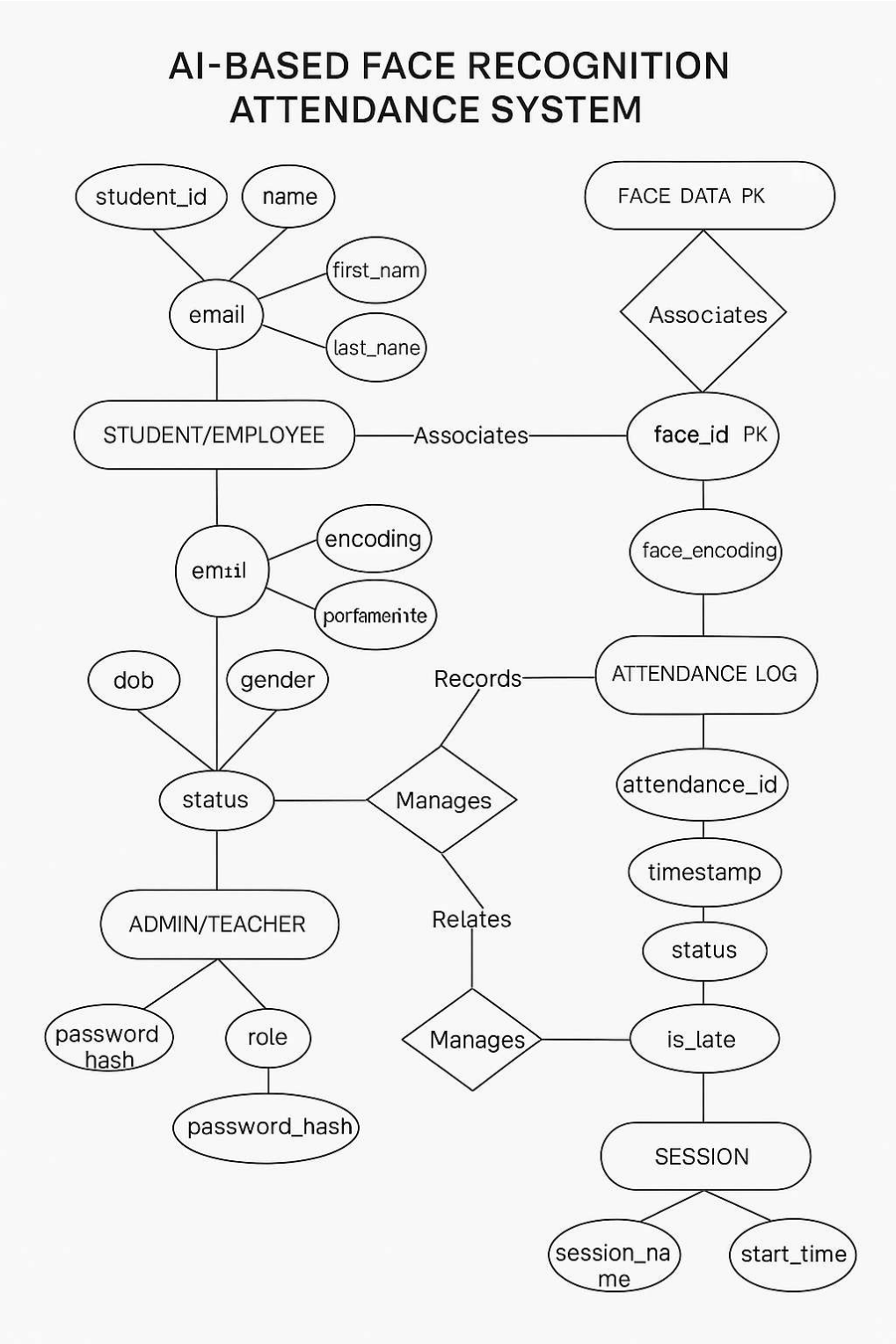
1. **Student/Employee (Entity)**:
    - **Attributes**:
        - student_id (Primary Key)
        - first_name
        - last_name
        - email
        - profile_picture (Image)
        - department
2. **Face Data (Entity)**:
    - **Attributes**:
        - face_id (Primary Key)
        - student_id (Foreign Key)
        - face_encoding (Data related to the unique face features)
3. **Attendance Log (Entity)**:
    - **Attributes**:
        - attendance_id (Primary Key)
        - student_id (Foreign Key)

        - timestamp
        - status (Present/Absent)
4. **Admin/Teacher (Entity)**:
    - **Attributes**:
        - admin_id (Primary Key)
        - first_name
        - last_name
        - email
        - role (Admin/Teacher)
5. **Session (Entity)**:
    - **Attributes**:
        - session_id (Primary Key)
        - session_name
        - start_time
        - end_time

Relationships:

1. **Student/Employee to Face Data**:
    - **Relationship**: One-to-One
    - **Explanation**: Each student or employee has one set of face data associated with them for recognition.
2. **Student/Employee to Attendance Log**:
    - **Relationship**: One-to-Many
    - **Explanation**: A student or employee can have multiple attendance records over time.
3. **Admin/Teacher to Attendance Log**:
    - **Relationship**: One-to-Many
    - **Explanation**: Admin or Teacher can view or manage multiple attendance records.
4. **Session to Attendance Log**:
    - **Relationship**: One-to-Many

- **Explanation**: Each session (class, meeting, etc.) can have multiple attendance logs for the students/employees.



PRACTICAL NO.: 6

Aim:

Design Use Case - Diagram for Mini Project.

Project Title:

Face Recognition Attendance System

---

1. Actors:

- Student
- CCTV Cameras
- Server
- AI Model
- Face Recognition API
- Database
- Faculty
- Cloud Services

2. Preconditions:

- CCTV cameras must be operational and continuously stream video.
- The server should have sufficient storage and processing power to handle video streams.
- The AI model should be trained with a dataset containing authorized individuals.
- The database should be accessible and capable of storing attendance records.
- Cloud services must be configured for data backup and access.

3. Basic Flow:

1. CCTV cameras stream real-time video footage to the server.
2. The server processes the video and extracts relevant frames.
3. The backend AI engine uses the AI model and Face Recognition API to analyze the frames.
4. If a match is found, attendance is recorded in the database.
5. The attendance records are backed up to the cloud database.
6. Faculty members can access attendance records through the frontend interface.

4. Alternative Flow:

- If no match is found, the system discards the frame and continues monitoring.
- If video processing fails, the server logs an error and alerts the administrator.
- If recognition fails multiple times, students must manually verify attendance.

5. Postconditions:

- Attendance is successfully marked for recognized students.
- The database and cloud storage are updated with attendance records.
- Faculty members can retrieve attendance reports via the frontend interface.

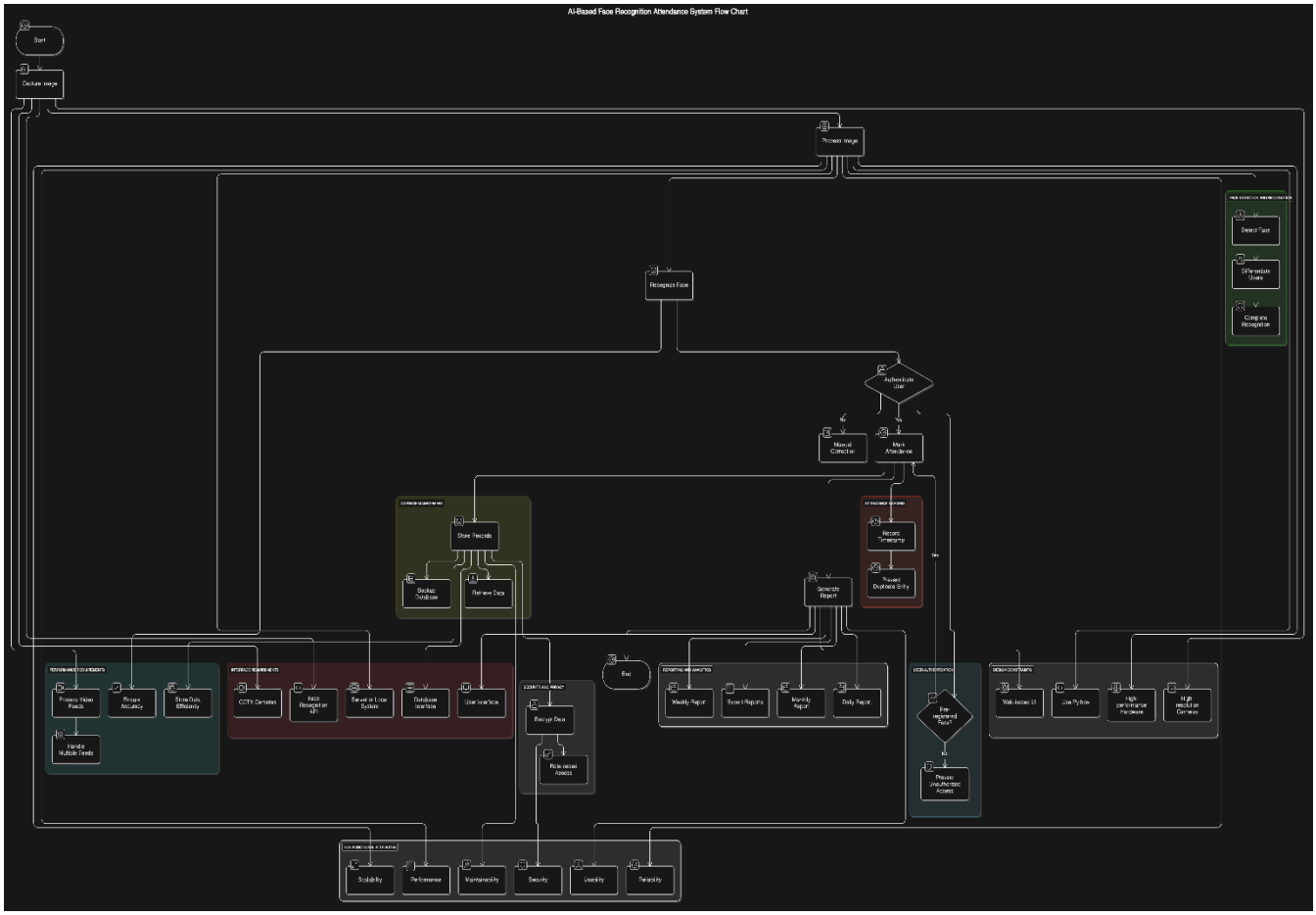
6. Exceptions:

- Low camera resolution may reduce recognition accuracy.
- Network failures may interrupt cloud data synchronization.
- Unauthorized individuals will not be marked present.

7. Assumptions:

- The system has a stable network connection for real-time processing.
- The AI model is regularly updated to improve recognition accuracy.
- Faculty members have the necessary credentials to access attendance records.

Use Case - Diagram



PRACTICAL NO.: 7

Aim:

Design Class - Diagram for Mini Project.

Project Title:

Face Recognition Attendance System

---

1. Classes:

- User *(Abstract Class)*
- Administrator *(Inherits from User)*
- Student/Employee *(Inherits from User)*
- IT Staff *(Inherits from User)*
- FaceRecognizer
- Camera
- MachineLearningModel
- AttendanceManager
- DatabaseManager
- ReportGenerator
- UI
- SecurityManager

---

2. Attributes and Methods:

User (abstract)

- Attributes:
  userID, name, role
- Methods:
  login(), logout()

Administrator *(inherits User)*

- Methods:
  generateReport(), manualOverride()

Student/Employee *(inherits User)*

- Methods:
  authenticateFace(), viewAttendanceRecord()

IT Staff ***(inherits User)***

- Methods:
  maintainSystem(), backupDatabase(), trainModel()

FaceRecognizer

- Attributes:
  model, camera
- Methods:
  captureFace(), detectFace(), recognizeFace()

Camera

- Methods:
  captureImage(), streamVideo()

MachineLearningModel

- Methods:
  trainModel(), predict(image)

AttendanceManager

- Attributes:
  attendanceRecords
- Methods:
  markAttendance(), checkDuplicate(), getAttendanceByDate()

DatabaseManager

- Attributes:
  dbConnection
- Methods:
  storeRecord(), retrieveRecord(), backupDatabase()

ReportGenerator

- Methods:
  generateDailyReport(), generateMonthlyReport(), exportReport(format)

UI

- Methods:
  displayDashboard(), displayReports(), adminControls()

SecurityManager

- Methods:
  encryptData(), decryptData(), authorizeUser()

---

3. Relationships:

- User is an abstract base class with Administrator, IT Staff, and Student/Employee as subclasses.
- FaceRecognizer has an association with Camera and MachineLearningModel.
- AttendanceManager uses DatabaseManager to store records.
- ReportGenerator pulls data from DatabaseManager.
- UI interacts with Administrator and ReportGenerator.
- SecurityManager ensures secure access and encryption throughout the system.

---

4. Class Diagram Flow (Basic Explanation):

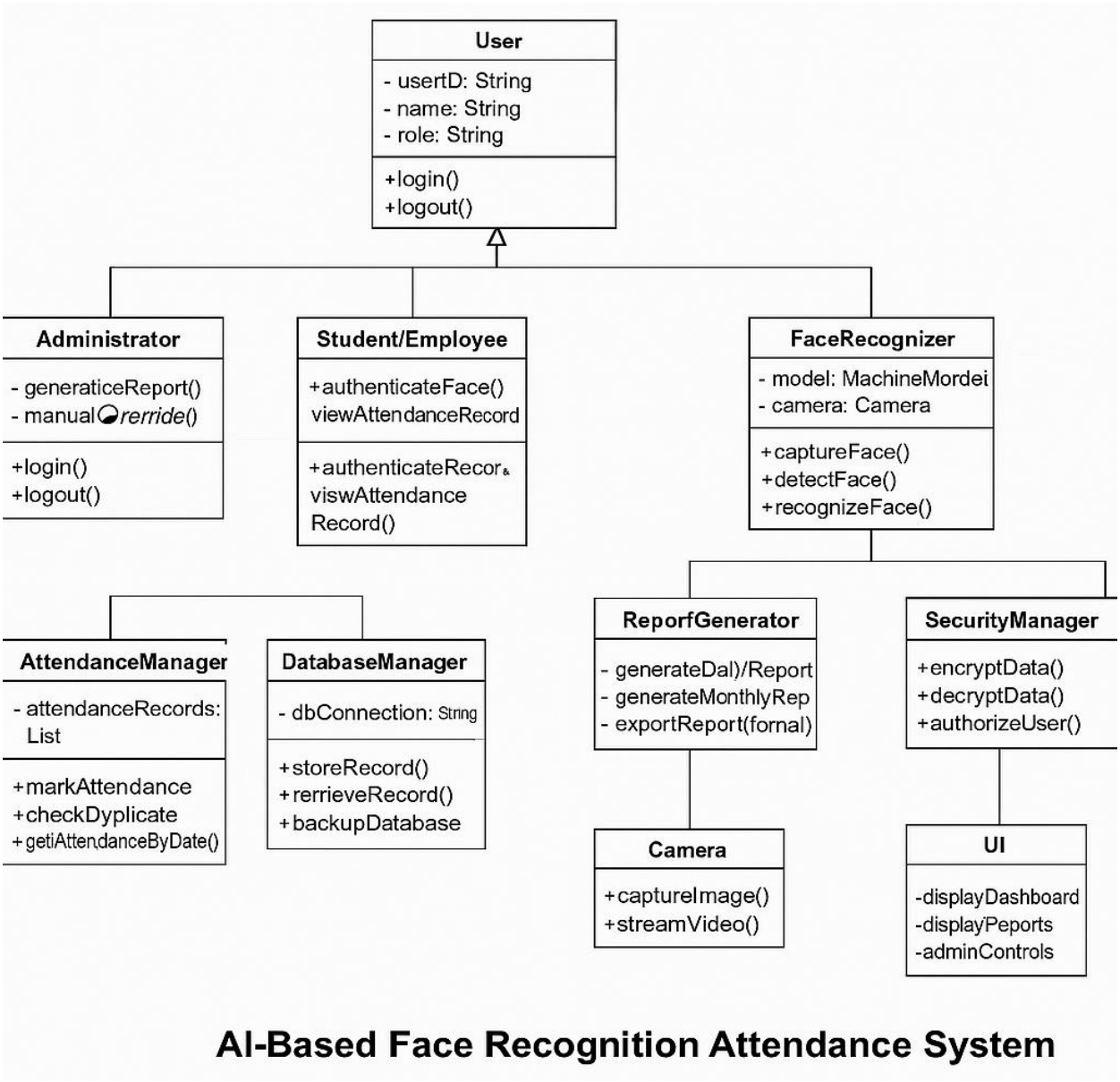
1. The Camera captures video feeds.
2. The FaceRecognizer class interacts with Camera and MachineLearningModel to recognize faces.
3. Once a face is recognized, the AttendanceManager marks attendance and sends it to DatabaseManager.
4. The ReportGenerator pulls data to generate reports.
5. Administrator uses the UI to view reports and manually override attendance if necessary.
6. SecurityManager ensures encryption and role-based access to sensitive data.

---

5. Assumptions:

- All class methods are implemented in a modular and scalable manner.
- The system is deployed on hardware with sufficient processing power.
- The user interface is intuitive and responsive.
- All actors are authenticated via SecurityManager before access.

Class – Diagram



PRACTICAL NO.: 8

Aim:

Design Sequence - Diagram for Mini Project.

Project Title:

AI Based Face Recognition Attendance System

Elements Used in the Sequence Diagram:

1. **Actors (Entities):**
   - Student
   - System
   - Camera
   - AI Engine
   - Database
   - Faculty
2. **Lifelines:**
   - Each module (Student, System, Camera, AI Engine, Database, Faculty) has a lifeline representing its existence throughout the process.
3. **Messages (Arrows):**
   - **Synchronous Messages (Solid Arrows):** Represent direct communication between components (e.g., "Capture Image," "Process Image").
   - **Asynchronous Messages (Dashed Arrows):** Represent responses (e.g., "Recognition Result," "Confirmation").
4. **Alt (Alternative) Blocks:**
   - Used for conditional logic (e.g., "Face Recognized" vs. "Face Not Recognized").
5. **Loop Blocks:**
   - Used for repeated processes (e.g., retrying face recognition if the recognition fails).
6. **Parallel Processing (Par Block):**
   - Allows multiple attendance requests to be processed simultaneously.

This sequence diagram visually represents the AI-based face recognition attendance system's step-by-step execution with alternative and parallel execution paths.

Sequence - Diagram


AI-Based Face Recognition Attendance System
Student
System
Camera
AI Engine
Database
Faculty
Request attendance
Capture image
Image data
loop
until face recognized
Process image
Recognition result
alt
face recognized
Update attendance
Confirmation
Attendance marked
face not recognized
Recognition failed
Retry
par
parallel processing
Request attendance
Capture image
Image data
Process image
Recognition result
alt
face recognized
Update attendance
Confirmation
Attendance marked
face not recognized
Recognition failed
Capture image
Image data
Process image
Recognition result
alt
face recognized
Update attendance
Confirmation
face not recognized
Student
System
Camera
AI Engine
Database
Faculty

PRACTICAL NO.: 9

Aim:

Design Activity- Diagram for Mini Project.

Project Title:

AI Based Face Recognition Attendance System

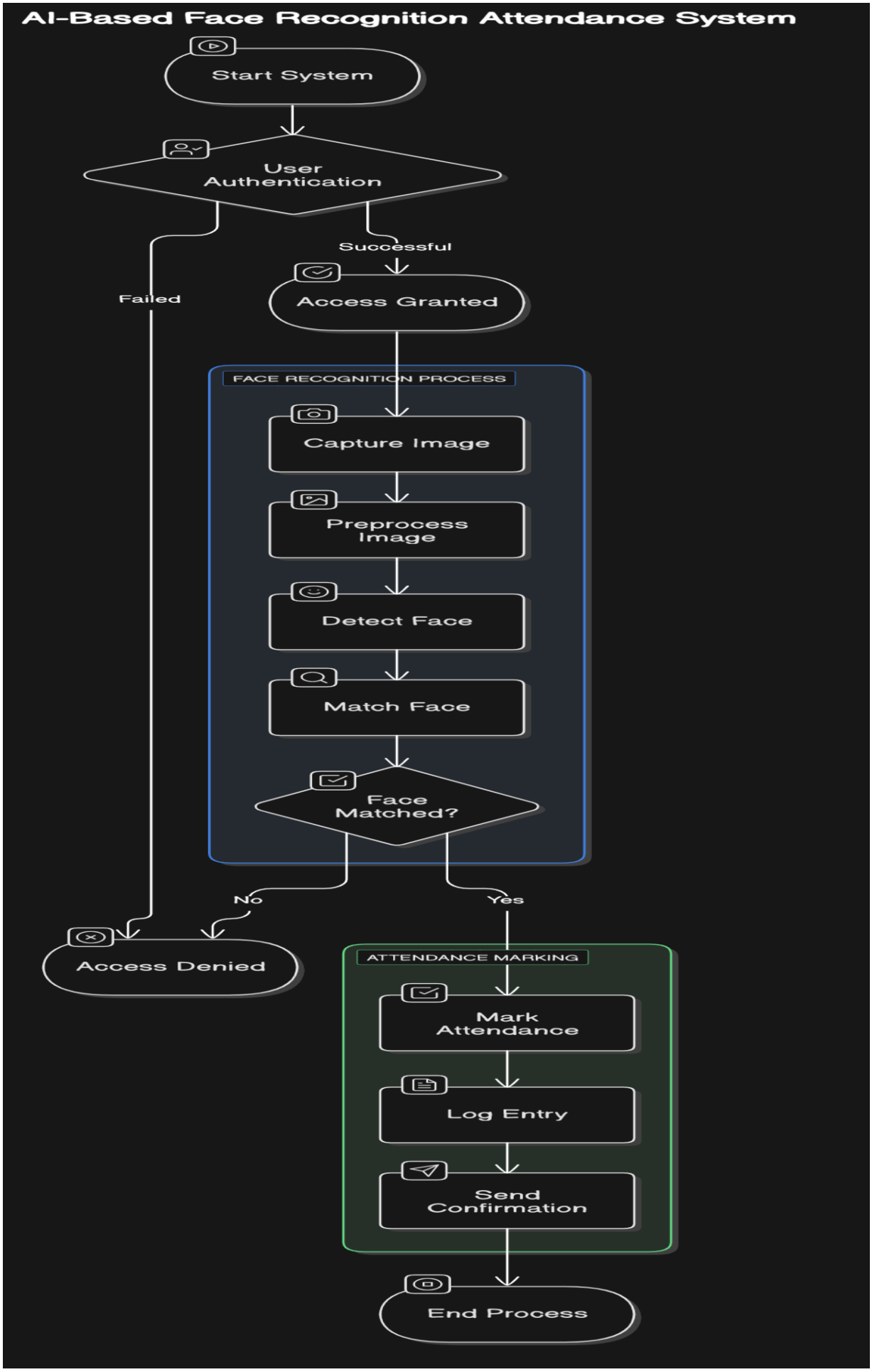
Elements Used in the Activity Diagram:

1. **Start & End Nodes:**
   - **Start Node:** Represents the beginning of the process ("Start System").
   - **End Node:** Marks the termination of the process ("End Process").
2. **Action/Activity Nodes:**
   - Represent tasks performed in the process, such as:
     - **User Authentication**
     - **Capture Image**
     - **Preprocess Image**
     - **Detect Face**
     - **Match Face**
     - **Mark Attendance**
     - **Log Entry**
     - **Send Confirmation**
3. **Decision Nodes (Diamonds):**
   - Used to represent conditional branching:
     - **Authentication Successful/Failed?**
     - **Face Matched?**
     - If "No," the process leads to "Access Denied."
     - If "Yes," the attendance marking process continues.
4. **Swimlanes (Optional Categorization):**
   - The diagram visually categorizes the **Face Recognition Process** and **Attendance Marking** into separate sections using borders.
5. **Control Flow (Arrows):**

- Represent the sequence of execution between nodes.

This activity diagram visually represents the step-by-step workflow of the **AI-Based Face Recognition Attendance System**, ensuring proper user authentication, face recognition, and attendance marking with appropriate decision handling.

Activity- Diagram


AI-Based Face Recognition Attendance System
Start System
User Authentication
Failed
Successful
Access Granted
FACE RECOGNITION PROCESS
Capture Image
Preprocess Image
Detect Face
Match Face
Face Matched?
No
Yes
Access Denied
ATTENDANCE MARKING
Mark Attendance
Log Entry
Send Confirmation
End Process