**CHAPTER 4**

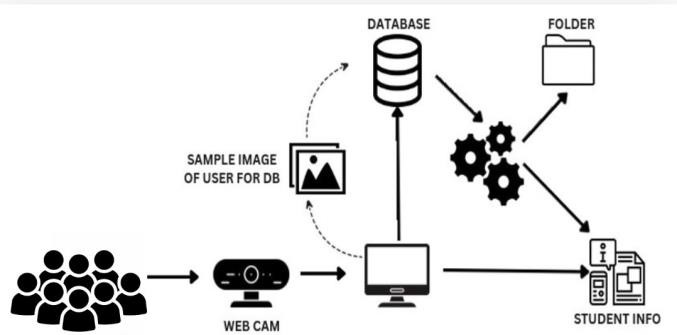
**SYSTEM DESIGN**

**IV. CHAPTER   
SYSTEM DESIGN**

The system design of **EDUCARE** outlines the architecture and the interaction between various components that work together to automate the student attendance process using **facial recognition technology**. The system is divided into the following key components:

### ****4.1 System Overview****

The **EDUCARE system** captures group images of students from the classroom using a webcam. It then performs facial recognition using **OpenCV and TensorFlow** to identify the students. Once identified, their attendance is automatically marked in the database. The administrator can then access the records through a web-based application.



### ****4.2 System Components****

The major components of the EDUCARE system include:

| **Component** | **Description** |
| --- | --- |
| **Webcam** | Captures real-time images of students in the classroom. |
| **Computer System** | Processes the images using Python, OpenCV, and TensorFlow. |
| **Database** | Stores student details, attendance records, and login information. |
| **Folder (File Storage)** | Stores the sample images of students for facial recognition. |
| **Student Info** | Contains the student's personal and academic details. |
| **User Interface (UI)** | Provides an interface for the administrator to view attendance records and monitor logs. |

### ****4.3 Working Flow of the System****

The system workflow can be explained in the following steps:

1. **Image Capturing:** The **webcam** captures group photos of students in the classroom.
2. **Facial Detection:** The captured image is processed using **OpenCV** to detect individual student faces.
3. **Image Comparison:** The detected faces are compared with the sample images stored in the database.
4. **Identification:** If a match is found, the system identifies the student and marks their attendance.
5. **Data Storage:** The attendance records are then stored in the **database** for future reference.
6. **Access and Monitoring:** The admin or faculty can log in to the web-based application and view the attendance records.

### ****4.4 Data Flow Diagram (DFD)****

The following is the **Data Flow Diagram (DFD)** for the EDUCARE system:

**Level 0 (Context Diagram):**

* **Input:** Student group image from the webcam.
* **Process:** Image processing, face detection, and identification.
* **Output:** Attendance marked in the database

**Level 1 DFD (Detailed Process):**

* **Webcam captures image → System processes image → Face detected and matched → Attendance marked in the database.**

### ****4.5 Entity Relationship Diagram (ERD)****

The **Entity Relationship Diagram (ERD)** shows the relationship between different entities involved in the EDUCARE system.

#### **Entities and Their Relationships**

1. **Student Entity:**
   * Contains details of the student such as Name, Roll Number, Department, Year, and Image ID.
2. **Attendance Entity:**
   * Stores the attendance status of each student (Present/Absent).
3. **Admin Entity:**
   * Provides access to the administrator to manage student records and attendance logs.
4. **Database Entity:**
   * Acts as the central storage unit to store attendance data and student information.

### ****4.6 Database Tables****

The database design for the EDUCARE system consists of three primary tables: **attendance table**, **admin table**, and **image table**. These tables ensure efficient data storage, retrieval, and management of attendance records, admin credentials, and student images.

#### **Table 1: Attendance Table**

The attendance table is responsible for storing daily attendance records of each student. It records the student's identification, attendance date, and status (present or absent).

| **field name** | **data type** |  | **constraints** |
| --- | --- | --- | --- |
| attendance\_id | int |  | primary key, auto increment |
| student\_id | int |  | foreign key, references student\_id from image table |
| date | date |  | not null |
| status | enum('present', 'absent') |  | default: 'absent' |

* attendance\_id: This field acts as a primary key and automatically increments with each record.
* student\_id: This field is a foreign key that references the student from the image table.
* date: The date when the attendance was recorded..

#### **Table 2: Image Table**

The image table stores student images and their respective details.

| **field name** | **data type** | **constraints** |
| --- | --- | --- |
| student\_id | int | primary key, auto increment |
| name | varchar(100) | not null |
| image\_path | varchar(200) | not null |
| roll\_number | varchar(20) | unique, not null |

* student\_id: Unique identification number for each student.
* name: Full name of the student.
* image\_path: The file path of the student’s image stored on the server.
* roll\_number: Unique roll number assigned to each student.

### ****4.8 System Flow Chart****

The flow chart below represents the overall working process of the EDUCARE system.

1. **Start:** System initializes and captures group images.
2. **Process:** OpenCV detects faces and compares with the database.
3. **Match Found:** Attendance is marked in the database.
4. **Match Not Found:** The system skips unidentified faces.
5. **Report Generation:** Admin can generate and export reports.
6. **End:** The process ends once the attendance is recorded.