# Report on Creating the Database System

**Database Type:** We used MySQL Workbench 8.0 for this.

## 1. Database Setup

What We Created:

*Sql queries:-*

```
CREATE DATABASE attendance_system;
CREATE TABLE employee (id INT, name VARCHAR(255), designation
VARCHAR(255));
CREATE TABLE attendance (id INT AUTO_INCREMENT, employee_id INT, date
DATE, time_in TIME);
```

- ➢ Employee Table: Stores employee details (ID, Name, Job)
- ➢ Attendance Table: Automatically records check-in times

## Tables Created:

**Employees Table:**

```
CREATE TABLE employee (
   id INT PRIMARY KEY,
   name VARCHAR(255),
   designation VARCHAR(255);
```

- ➢ It stores employee IDs, names, and job roles
- ➢ Example: (4, 'Swastik', 'Data Science Student')

**Attendance Table:**

```
CREATE TABLE attendance (
   id INT AUTO_INCREMENT PRIMARY KEY,
   employee_id INT,
   date DATE,
   time_in TIME);
```

- ➢ Automatically records date/time when employees are recognized.
- ➢ Links to employees using their ID.

## Insertion:-

```
INSERT INTO employee VALUES
(0,'Arslan','Data Researcher'),
(1,'Praful','Senior Data Analyst'),
(2,'Sanket','Data Analyst'),
(3,'Rizwan','Data Analyst'),
(4,'Swastik','Data Science Student');
```

## Part 2: Connecting Python to Database

**Database Configuration:**

*In python code:-*

*Pip install*

```
DB_CONFIG = {
    'host': '128.0.01',
    'user': 'root',
    'password': 'root',
    'database': 'attendance_system',
    'port': 3306
}
```

Its like setting up an address to connect to the database,

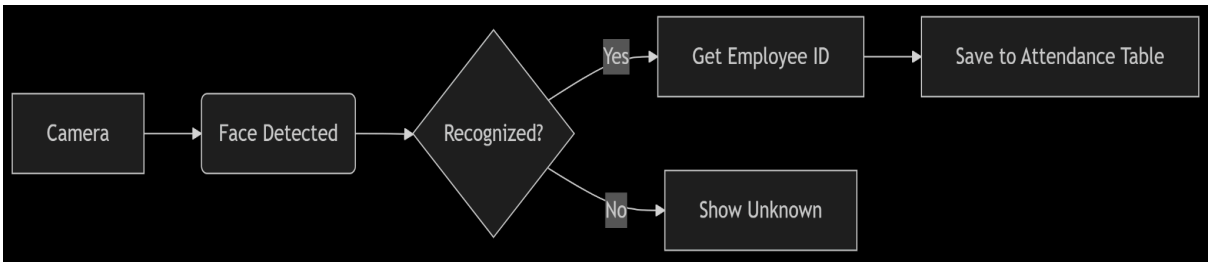
**Mark Attendance:-**

```
def mark_attendance(employee_id):
    # Checks if already marked today
    INSERT INTO attendance VALUES (employee_id, current_date, current_time)
```

View Attendance:

```
SELECT e.name, a.date, a.time_in
FROM attendance a
JOIN employee e ON a.employee_id = e.id
```

## Part 3: How It Works Together




## Key Features:

**Smart Checks:**

- ➢ Cooldown Period: Won't mark same person within 5 seconds
- ➢ Duplicate Prevention: Only 1 entry per day per employee
- ➢ Recognition Threshold: 70% confidence required (RECOGNITION_THRESHOLD = 0.7)

## Technical Components:-

| *Component* | *Purpose* | *Technology Used* |
|---|---|---|
| ***Face Detection*** | *Finds faces in video* | *YOLOv8 (AI model)* |
| ***Face Recognition*** | *Identifies employees* | *Facenet (AI embeddings)* |
| ***Database*** | *Stores records* | *MySQL* |
| ***Attendance Logic*** | *Manages check-ins* | *Python/SQL* |

## Attendance Report:

*Shows entries like:*

Praful  2023-05-20  09:15:00
Swastik  2023-05-20  09:16:30

## ***Queries used :-***