MINI PROJECT: PERSONAL DETAILS MANAGEMENT

Submitted by:

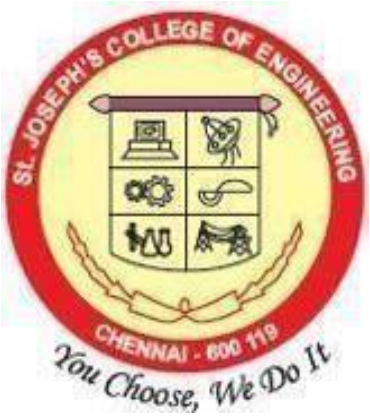
Fuaad Ahmad : 312322205303

Of

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY



St. JOSEPH'S COLLEGE OF ENGINEERING

(An Autonomous Institution)

St. Joseph's Group of Institution

OMR, Chennai 600 119

Table of Contents:

➢ Introduction:

- In the age of digitalization, managing personal information efficiently is crucial. This project aims to develop a Python-based application integrated with MySQL for managing personal details. The system allows users to add, search, and delete records based on Aadhaar number, facilitating organized storage and retrieval of information.

➢ Project Overview:

- The project provides a platform to input personal details including Aadhaar number, name, age, date of birth, phone number, and blood group. These details are stored in a MySQL database, enabling users to search for records based on Aadhaar number and delete them as necessary.

➢ Technologies Used:

- Python: Used for application development and user interface implementation.
- MySQL: Employed as the backend database management system for storing personal details.
- MySQL Connector/Python: Utilized to establish a connection between Python and MySQL for executing database operations.

➢ Database Schema:

- The database schema includes the following fields:

- Aadhaar number
- Name
- Age
- Date of Birth
- Phone number
- Blood group

➢ Functionality:

  - Add Record:
- Users can input personal details through the command-line interface and add a new record to the database.

  - Search Record:
- The system allows users to search for records using Aadhaar number, facilitating quick retrieval of personal details.

  - Delete Record:
- Users can delete records based on Aadhaar number, enabling the removal of outdated or erroneous information from the database.

➢ Implementation Details:

- The project architecture consists of Python scripts for handling user input, executing database operations, and presenting results. The MySQL database is utilized for efficient data storage and retrieval.

➢ User Interface:

- A command-line interface (CLI) is implemented for user interaction, ensuring simplicity and ease of use.

Program:

```
import mysql.connector

def connect_to_database():
    try:
        connection = mysql.connector.connect(
            host="localhost",
            user="username",
            password="password",
            database="databasename"
        )
        return connection
    except mysql.connector.Error as error:
        print("Failed to connect to MySQL:", error)

def add_person(connection, aadhaar_no, name, age, dob, blood_group, phone_no):
    try:
        cursor = connection.cursor()
        sql_query = "INSERT INTO person_details (aadhaar_no, name, age, dob, blood_group,
phone_no) VALUES (%s, %s, %s, %s, %s, %s)"
        data = (aadhaar_no, name, age, dob, blood_group, phone_no)
        cursor.execute(sql_query, data)
        connection.commit()
        print("Person details added successfully!")
    except mysql.connector.Error as error:
        print("Failed to add person details:", error)

def search_person_by_aadhaar(connection, aadhaar_no):
    try:
        cursor = connection.cursor()
        sql_query = "SELECT * FROM person_details WHERE aadhaar_no = %s"
        cursor.execute(sql_query, (aadhaar_no,))
        result = cursor.fetchone()
        if result:
            print("Person details found:")
            print("Aadhaar No:", result[0])
            print("Name:", result[1])
            print("Age:", result[2])
            print("Date of Birth:", result[3])
            print("Blood Group:", result[4])
            print("Phone No:", result[5])
        else:
            print("Person with Aadhaar number {} not found.".format(aadhaar_no))
    except mysql.connector.Error as error:
```

```
        print("Failed to search person details:", error)

def delete_person_by_aadhaar(connection, aadhaar_no):
    try:
        cursor = connection.cursor()
        sql_query = "DELETE FROM person_details WHERE aadhaar_no = %s"
        cursor.execute(sql_query, (aadhaar_no,))
        connection.commit()
        print("Person details with Aadhaar number {} deleted
successfully!".format(aadhaar_no))
    except mysql.connector.Error as error:
        print("Failed to delete person details:", error)
def create_table(connection):
    try:
        cursor = connection.cursor()
        cursor.execute("""
            CREATE TABLE IF NOT EXISTS person_details (
                aadhaar_no VARCHAR(12) PRIMARY KEY,
                name VARCHAR(255),
                age INT,
                dob DATE,
                blood_group VARCHAR(5),
                phone_no VARCHAR(15)
            )
        """)
    except mysql.connector.Error as error:
        print("Failed to create table:",error)
# Main function
def main():
    connection = connect_to_database()
    create_table(connection)
    if connection:
        while True:
            print("\n1. Add person details")
            print("2. Search person by Aadhaar number")
            print("3. Delete person by Aadhaar number")
            print("4. Exit")

            choice = input("Enter your choice: ")

            if choice == "1":
                aadhaar_no = input("Enter Aadhaar number: ")
                name = input("Enter name: ")
                age = int(input("Enter age: "))
                dob = input("Enter date of birth (YYYY-MM-DD): ")
                blood_group = input("Enter blood group: ")
                phone_no = input("Enter phone number: ")
```

```
            add_person(connection, aadhaar_no, name, age, dob, blood_group, phone_no)

        elif choice == "2":
            aadhaar_no = input("Enter Aadhaar number to search: ")
            search_person_by_aadhaar(connection, aadhaar_no)

        elif choice == "3":
            aadhaar_no = input("Enter Aadhaar number to delete: ")
            delete_person_by_aadhaar(connection, aadhaar_no)

        elif choice == "4":
            print("Exiting...")
            break

        else:
            print("Invalid choice!")

    connection.close()

main()
```

➢ Conclusion:

- In conclusion, the Python-MySQL Personal Details Management System represents an effective solution for organizing and managing personal information. By leveraging the capabilities of Python for application development and MySQL for database management, the project provides users with a robust platform for storing, searching, and deleting personal details.

- Through the implementation of functionalities such as adding, searching, and deleting records based on Aadhaar number, the project streamlines the process of managing personal information. Users can easily input their details, search for specific records using Aadhaar numbers, and delete outdated or erroneous information as needed.

```
========== RESTART: C:\Users\varun\OneDrive\Desktop\projectVIKI.py ==========

1. Add person details
2. Search person by Aadhaar number
3. Delete person by Aadhaar number
4. Exit
Enter your choice: 1
Enter Aadhaar number: 123456789123
Enter name: viki
Enter age: 19
Enter date of birth (YYYY-MM-DD): 2003-4-5
Enter blood group: o+ve
Enter phone number: 9738745634
Person details added successfully!

1. Add person details
2. Search person by Aadhaar number
3. Delete person by Aadhaar number
4. Exit
Enter your choice: 2
Enter Aadhaar number to search: 123456789123
Person details found:
Aadhaar No: 123456789123
Name: viki
Age: 19
Date of Birth: 2003-04-05
Blood Group: o+ve
Phone No: 9738745634

1. Add person details
2. Search person by Aadhaar number
3. Delete person by Aadhaar number
4. Exit
Enter your choice: 3
Enter Aadhaar number to delete: 123456789123
Person details with Aadhaar number 123456789123 deleted successfully!

1. Add person details
2. Search person by Aadhaar number
3. Delete person by Aadhaar number
4. Exit
Enter your choice: 4
Exiting...
>>>
```