**Visvesvaraya Technological University**

"Jnana Sangama", Machhe, BelagaVI – 590018



**Mini Project Report**

**on**

**POLICE STATION DATABASE MANAGEMENT SYSTEM**

Submitted in partial fulfillment of the requirements for the V semester

**Bachelor of Engineering**

in

**Computer Science and Engineering**

of

Visvesvaraya Technological University, Belagavi

by

**Ms. Lakshmi Lavanya S (1CD21CS066)**

**Ms. Priyanshi Prisha (1CD21CS117)**

**Ms. Rutu M belki (1CD21CS128)**

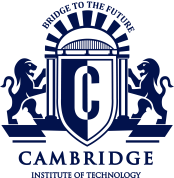
**Ms. Bhavana R (1CD21CS023)**

**Under the Guidance of**

Ms. Priyadarshini Ms. Bhavana

Assistant Professor, Assistant Professor,

Dept of CSE,CITech Dept of CSE,CITech

****

**Department of Computer Science and Engineering**

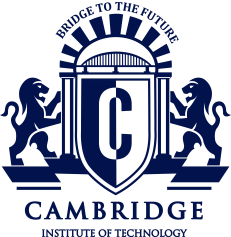
**Cambridge Institute of Technology, BEngalUrU** – **560036**

**2023-2024**

**Cambridge Institute of Technology**

K.R. Puram, Bengaluru – 560036

**Department of COMPUTER Science & Engineering**



**CERTIFICATE**

Certified that **Ms. Lakshmi Lavanya S, Ms. Priyanshi Prisha, Ms. Rutu M Belki, Ms. Bhavana R,** bearing USN **1CD21CS066, 1CD21CS117, 1CD21CS128, 1CD21CS023,** respectively are bonafide students of **Cambridge Institute of Technology,** has successfully completed Mini Project entitled “**Police Station Database Management System**” in partial fulfillment of the requirements for V semester **Bachelor of Engineering** in **Computer Science and Engineering** of **Visvesvaraya Technological University, Belagavi** during academic year 2023 - 2024. It is certified that all Corrections / Suggestions indicated for Internal Assessment have been incorporated in the report. The Database Project report has been approved as it satisfies the academic requirements in respect of Project Work prescribed for the said semester.

**Internal Guides: Head of the Dept.**

1. **Ms. Priyadarshini Dr. Shashikumar D. R.**

**Dept. of CSE., CiTech. Dept. of CSE., CiTech.**

1. **Ms. Bhavana**

**Dept. of CSE., CiTech.**

**Examiners:**

1. **2)**

**ACKNOWLEDGEMENT**

We are extremely thankful to **Dr. Indumathi G,** Principal, CITech., Bengaluru, for providing us the academic ambience and everlasting motivation to carry out this work and shaping our careers.

We express our sincere gratitude to **Dr. Shashikumar D. R.,** HOD, Dept. of Computer Science and Engineering, CITech., Bengaluru, for his stimulating guidance, continuous encouragement and motivation throughout the course of present work.

We also wish to extend our thanks to **Priyadarshini,** Assistant Professor Dept. of Computer Science and Engineering, CITech., Bengaluru and **Bhavana,** Assistant Professor, Dept. of Computer Science and Engineering, CITech., Bengaluru, for their expert guidance and constructive suggestions to improve the quality of this work.

We would also like to thank all other teaching and technical staffs of Department of Computer Science and Engineering, who have directly or indirectly helped us in the completion of this Project Work.

And lastly we would hereby acknowledge and thank our parents who have been a source of inspiration and also instrumental in the successful completion of this project.

**Lakshmi Lavanya S (1CD21CS066)**

**Priyanshi Prisha (1CD21CS117)**

**Rutu M Belki (1CD21CS128)**

**Bhavana R (1CD21CS023)**

**ABSTRACT**

Police station database management system is a project that is designed with the aim of maintaining all the records and details related to a police station in order to increase efficiency. As a result of making it easier to manage and administer a police station, this record management system makes the management and administration of a police station easier and more effective. A major purpose of this police station database management system is to automate the functions involved in the management of a police station by utilizing computer technology. Here, the effort is put forward to convert the current data management system of the police station into a computerized system, so that a faster, more secure, error-free and more efficient computer system can be implemented, so that the police station's data can be managed better. Identifying the problem and collecting details of the problem are the first steps. The process involves gathering information about the existing system's problems, determining how the problem should be solved, and determining the cost estimates and implementation specifications of the police station management system.

A new computerized system will be implemented to replace the existing manual system that is currently in use. Reducing the delay can make the data receiving process more efficient. There is no doubt that computer technology has become an essential part of our daily lives in this new world. There are almost no fields in which the use of computers has not simplified procedures. As a result of this police station record management system project, almost all of the facilities for entering the details in the Police Station, which are currently performed manually, have been provided.

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTERS** |  | **PAGE NO** |
| Chapter 1 | Introduction | 1 |
| Chapter 2 | Requirements | 4 |
| Chapter 3 | Entity Relationship Diagram | 5 |
| Chapter 4 | Schema Diagram | 7 |
| Chapter 5 | Implementation | 8 |
| Chapter 6 | Snapshots | 17 |
|  | Conclusion | 20 |
|  | References | 21 |

ii

v

**CHAPTER 1**

**INTRODUCTION**

The proposed system applies to all Police stations across the country and specifically looks into the subject of Crime Records Management. It is well understood that Crime Prevention, Detection and Conviction of criminals depend on a highly responsive backbone of Information Management. The efficiency of the police function and the effectiveness with which it tackles crime depend on what quality of information it can derive from its existing records and how fast it can have access to it. It is proposed to centralize Information Management in Crime for the purposes of fast and efficient sharing of critical information across all Police Stations across the territory. Initially, the system will be implemented across Cities and Towns and later on, be interlinked so that a Police detective can access information across all records in the state thus helping speedy and successful completion to cases. The System would also be used to generate information for proactive and preventive measures for fighting crime.

The project has been planned to be having the view of distributed architecture, with centralized storage of the database. The application for the storage of the data has been planned. Using the constructs of SQL server. The standards of security and data protective mechanisms have been given a big choice for proper usage. The application takes care of different modules and their associated parts, which are produced as per the applicable strategies and standards that are put forward by the administrative staff.

**1.1 Existing System**

The existing system contains about all the police stations that are registered as per the jurisdiction of the system. It also gets integrated with the employees who are working in these stations along with their designation.

**1.1.2 Problem statement**

The existing system doesn’t have system security. That means, the user can login into the system anywhere in the world. But the data in this system is not for the public.

**1.1.3 Proposed System**

The system after careful analysis has been identified to be presented with the following modules:

Police stations registration module: This module maintains the information about all the police stations that are registered as per the jurisdiction of the system. It also gets integrated with the employees who are working in these stations along with their designation.

Investigating evidence registration module:

This module makes a collection of information related to all the evidence that become categorically important under the normal sequence of the investigation, this module dynamically concentrates upon the changes that take place while the system of investigation is under process.

* 1. **Need**

The proposed system applies to all Police stations across the country and specifically looks into the subject of Crime Records Management. It is well understood that Crime Prevention, Detection and Conviction of criminals depend on a highly responsive backbone of Information Management. The efficiency of the police function and the effectiveness with which it tackles crime depend on what quality of information it can derive from its existing records and how fast it can have access to it.

**1.2.1 Soft wares used and implementation :**

1. Database Design with MySQL Workbench:

* Begin by designing the database schema using MySQL Workbench. Define the tables to represent entities such as books, authors, borrowers, transactions, etc. Determine the attributes for each entity and establish relationships between tables using foreign keys.
* Use MySQL Workbench to create the database, tables, and relationships according to the designed schema. This involves writing SQL scripts or using the visual interface of MySQL Workbench to define the database structure.

2. Python:

* Python being a high-level language provides support for various databases. We can connect and run queries for a particular database using Python and without writing

raw queries in the terminal or shell of that particular database, we just need to have that database installed in our system.

3. Data Population and Management:

* Once the database schema is created, populate the tables with sample data or import existing data if available. This can be done using SQL INSERT statements or through the import functionality provided by MySQL Workbench. Implement procedures for managing data within the database, such as adding new data, updating existing records, and handling all other data.

4. Integration of Python and MySQL:

* Establish a connection between the Python application and the MySQL database. Write Python code to interact with the database, such as executing SQL queries to retrieve data based on user input, updating database records, and handling transactions.
* Implement event handling in Python to respond to user actions such as button clicks, text field entries, etc., and trigger corresponding database operations.

5. Testing and Debugging:

* Test the integrated system to ensure that the GUI functions correctly and interacts seamlessly with the MySQL database. This involves testing various scenarios.

**CHAPTER 2**

**REQUIREMENTS**

**2.1 Software Requirement Specifications**

* Operating System Front End :
* Python
* Back End Server Documentation :
* MySQL Workbench (local assistance , MySQL80 )

**2.2 Hardware Requirement Specifications**

* Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz 2.11 GHz
* 8.00 GB (7.83 GB usable)
* 64-bit operating system, x64-based processor

**CHAPTER 3**

**ENTITY RELATIONSHIP DIAGRAM**

An Entity Relationship Diagram is a diagram that represents relationships among entities in a database. It is commonly known as an ER Diagram. An ER Diagram in DBMS plays a crucial role in designing the database. Today’s business world previews all the requirements demanded by the users in the form of an ER Diagram. Later, it's forwarded to the database administrators to design the database. An Entity Relationship Diagram (ER Diagram) pictorially explains the relationship between entities to be stored in a database. Fundamentally, the ER Diagram is a structural design of the database. It acts as a framework created with specialized symbols for the purpose of defining the relationship between the database entities. ER diagram is created based on three principal components: entities, attributes, and relationships.

**Why Use ER Diagrams in DBMS?**

* ER Diagram helps you conceptualize the database and lets you know which fields need

to be embedded for a particular entity

* ER Diagram gives a better understanding of the information to be stored in a database
* It reduces complexity and allows database designers to build databases quickly
* It helps to describe elements using Entity-Relationship models
* It allows users to get a preview of the logical structure of the database

**Symbols Used in ER Diagrams**

* Rectangles: This Entity Relationship Diagram symbol represents entity types
* Ellipses: This symbol represents attribute
* Diamonds: This symbol represents relationship types
* Lines: It links attributes to entity types and entity types with other relationship types
* Primary key: Here, it underlines the attributes
* Double Ellipses: Represents multi-valued attributes

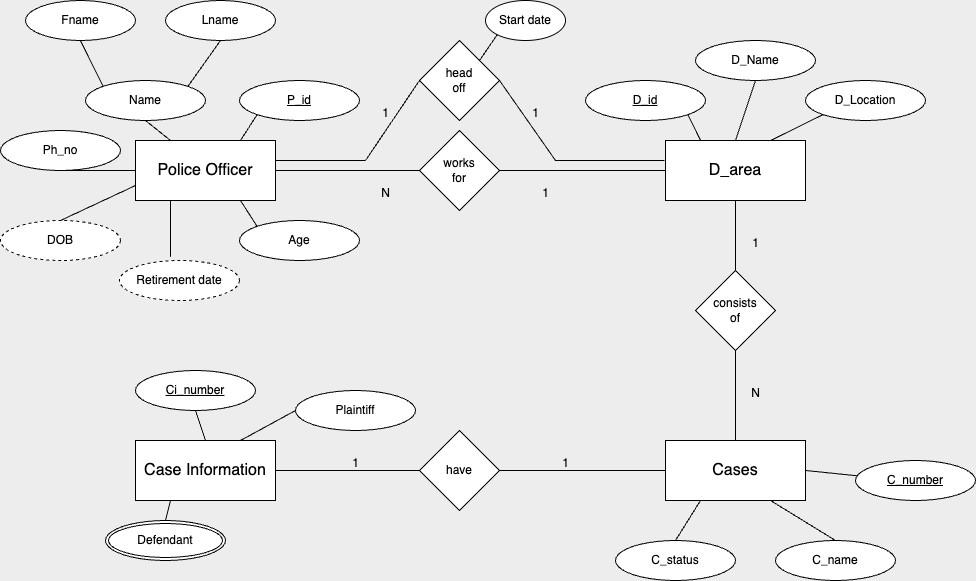


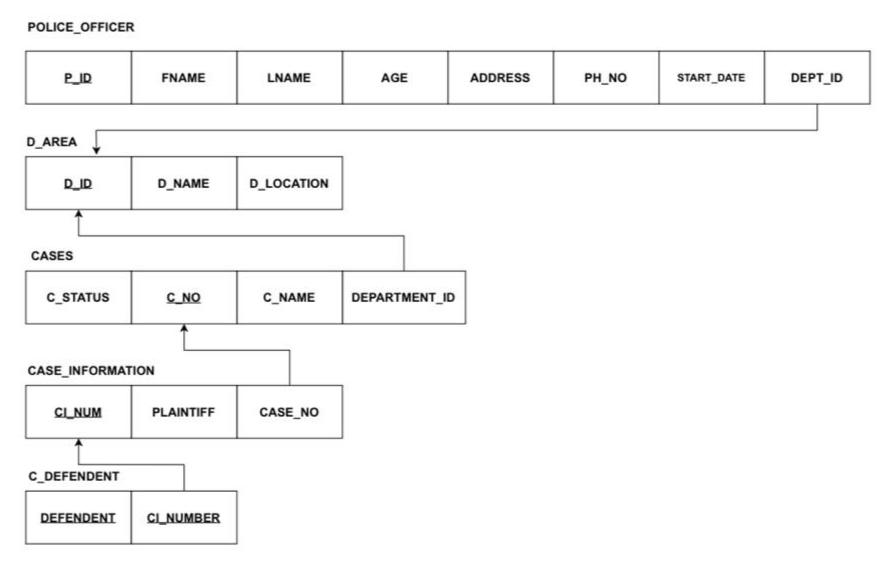
Fig.3.1 Entity Relationship Diagram

**CHAPTER 4**

**SCHEMA DIAGRAM**

In the realm of database management, a schema serves as the blueprint or structure that defines the organization, relationships, and constraints of the data stored within a database. Specifically, when discussing the schema of a police station database, we delve into the intricate design that governs the storage and management of information pertinent to crime operations. This comprehensive explanation not only elucidates the components and entities comprising a police station database schema but also underscores the significance of this framework in facilitating efficient data organization, retrieval, and management within police station database management systems.

**4.1 SCHEMA DIAGRAM**

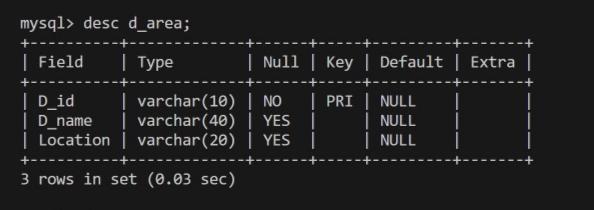


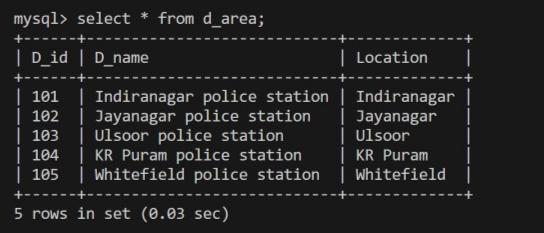
**CHAPTER 5**

**IMPLEMENTATION**

**5.1 Backend Implementation (**database)

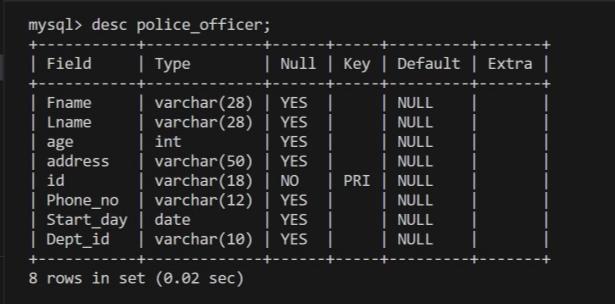
**D\_AREA TABLE:**

CREATE TABLE D\_AREA( D\_ID VARCHAR(10), D\_NAME VARCHAR(48), LOCAT VARCHAR(28),PRIMARY KEY(D\_ID));VARCHAR(20) UNIQUE);

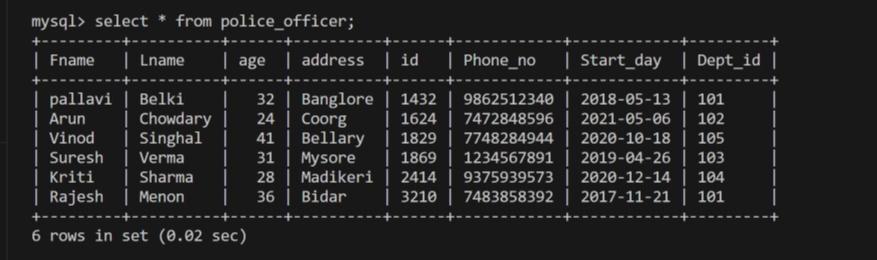
INSERT INTO D\_AREA(D\_ID, D\_NAME, LOCATION) VALUES( '101', 'INDIRANAGAR POLICE STATION', 'INDIRANAGAR'),('102','JAYANAGAR POLICE STATION','JAYANAGAR'),('103','ULSOOR POLICE STATION','ULSOOR'),('104','KR PURAM POLICE STATION','KR PURAM'),('105','WHITEFIELD POLICE STATION','WHITEFIELD')

**POLICE\_OFFICER TABLE**

CREATE TABLE POLICE\_OFFICER(FNAME VARCHAR(28), LNAME VARCHAR(28), AGE INT, ADDRESS VARCHAR(50), ID VARCHAR(18),PHONE\_NO VARCHAR(12), START\_DAY DATE, DEPT\_ID VARCHAR(10) REFERENCES D\_AREA(D\_ID) ON DELETE CASCADE,PRIMARY KEY (ID));

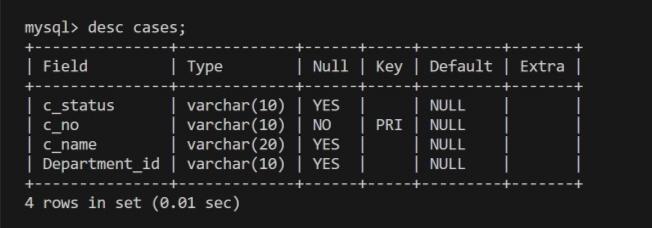


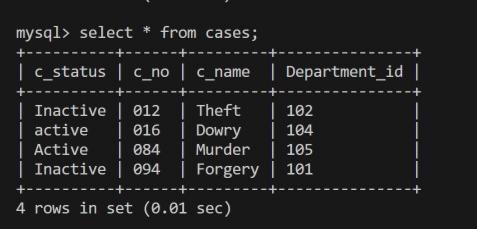
INSERT INTO POLICE\_OFFICER (FNAME,LNAME,AGE,ADDRESS,ID , PHONE\_NO, START\_DAY,DEPT\_ID)VALUES('PALLAVI','ROY',32,'BANGLORE','1432','9862512340','2018-05-13','101'),('SURESH','VERMA',31,'MYSORE','1869','1234567891','2019-04-26', '103'),('KRITI','SHARMA',28,'MADIKERI','2414','14254828484','2020-12-14','104'),('RAJ ESH','MENON',36,'BIDAR','3210','7483858392','2017-11-21','101'),('ARUN','CHOWDARY' ,24,'COORG','1624','92949294940','2021-05-06','102'),('VINOD','SINGHAL',41, 'BELLARY','1829','7748284944','2020-10-18','105');



**CASES TABLE**

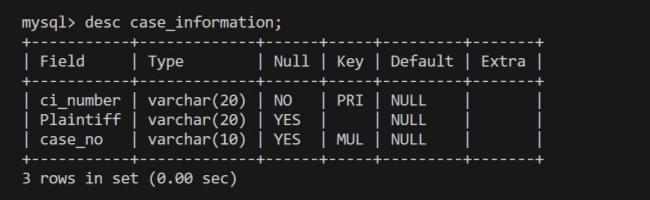
CREATE TABLE CASES (C\_STATUS VARCHAR(10), C\_NO VARCHAR(10),C\_NAME VARCHAR(20),DEPARTMENT\_ID VARCHAR(10) REFERENCES D\_AREA(D\_ID) ON DELETE CASCADE, PRIMARY KEY (C\_NO));

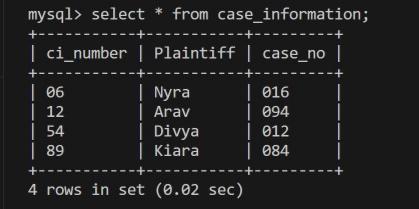


INSERT INTO CASES (C\_STATUS, C\_NO,C\_NAME, DEPARTMENT\_ID) VALUES ('ACTIVE','002', 'COUNTERFEIT','103'),('ACTIVE','016','DOWRY','104'), ('ACTIVE','084','MURDER','105'),('INACTIVE','094', 'FORGERY','101'), ('INACTIVE','012','THEFT','102');

**CASE\_INFORMATION TABLE**

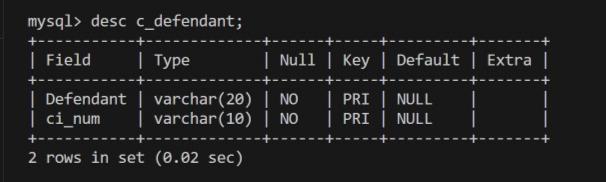
CREATE TABLE CASE\_INFORMATION (CI\_NUMBER VARCHAR(20), PLAINTIFF VARCHAR(20), DEFENDANT VARCHAR(20), CASE\_NO VARCHAR(10) REFERENCES CASES (C\_NO) ON DELETE CASCADE, PRIMARY KEY (CI\_NUMBER));

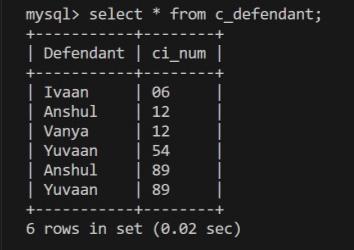
****

INSERT INTO CASE\_INFORMATION (CI\_NUMBER, PLAINTIFF, DEFENDANT, CASE\_NO) VALUES ('12','ARAV','VANYA','094'),('54','DIVYA','YUVAAN','012'), ('89', 'KIARA','ANSHUL','084'),('10','DHRUV','DIVIT','002'),('06','NYRA','IVAAN','016');

**C\_DEFENDENT TABLE**

CREATE TABLE C\_DEFENDANT (DEFENDANT VARCHAR(20), CI\_NUM VARCHAR(10) REFERENCES CASE\_INFORMATION (CI\_NUMBER) ON DELETE CASCADE,PRIMARY KEY (DEFENDANT, CI\_NUM));



INSERT INTO C\_DEFENDANT (DEFENDANT, CI\_NUM)VALUES ('YUVAAN', '54'), ('VANYA','12'),('DIVIT','10'),('IVAAN','06'),('ANSHUL','89');

**5.2 Frontend Implementation**

**Code for Login into the Window**

from tkinter import \*

import tkinter.messagebox

from tkinter import ttk

from tkinter import font

import mysql.connector

conn = mysql.connector.connect(

                host='localhost',

                user='root',

                password='xyz@007',

                database='police\_dept' )

print("DATABASE CONNECTION SUCCESSFUL")

class Cases\_menu:

    def \_\_init\_\_(self,master):

        self.master = master

        self.master.title("Case management")

        self.master.geometry("1500x700+0+0")

        self.master.config(bg="#bcdbf7")

        self.frame = Frame(self.master,bg="#bcdbf7")

        self.frame.pack()

        self.insert\_status = StringVar()

        self.c\_status=StringVar()

        self.c\_no=StringVar()

        self.c\_name=StringVar()

        self.ci\_number=StringVar()

        self.plaintiff=StringVar()

        self.Defendant=StringVar()

        self.case\_no=StringVar()

        self.ci\_num=StringVar()

        self.c\_status=StringVar()

 self.Department\_id=StringVar()

self.lblTitle = Label(self.frame,text = "CASE RECORDS", font="Helvetica 20 bold",bg="#bcdbf7")

self.lblTitle.grid(row =0 ,column = 0,columnspan=2,pady=50)

**Code for Login Buttons:**

self.lblTitle = Label(self.frame,text = "CASE RECORDS", font="Helvetica 20 bold",bg="#bcdbf7")

self.lblTitle.grid(row =0 ,column = 0,columnspan=2,pady=50)

#==============FRAME==========

self.LoginFrame=Frame(self.frame,width=400,height=80,relief="ridge",bg="#4682b4",bd=9)

self.LoginFrame.grid(row=2,column=0,pady=(0,15))

self.LoginFrame1 = Frame(self.frame,width=300,height=20,relief="ridge",bg="#bcdbf7",bd=0)

self.LoginFrame1.grid(row=1,column=0)

self.LoginFrame2 = Frame(self.frame,width=400,height=80,relief="ridge",bg="#4682b4",bd=5)

self.LoginFrame2.grid(row=2,column=0,pady=(420,0))

**Code for update button:**

#update the officer details

def main():

root = Tk()

app = Cases\_menu(root)

root.mainloop()

if \_\_name\_\_ == "\_\_main\_\_": main()

Code for delete button of customer:

#CLASS FOR DISPLAY MENU FOR DELETE EMPLOYEE

class D\_officer:

def \_\_init\_\_(self,master):

global de1\_case,de

self.master = master

self.master.title("POLICE DEPARTMENT MANAGEMENT SYSTEM")

self.master.geometry("1500x700+0+0")

self.master.config(bg="#bcdbf7")

self.frame = Frame(self.master,bg="#bcdbf7")

self.frame.pack()

self.de1\_case=StringVar()

self.lblTitle = Label(self.frame,text = "DELETE CASE WINDOW", font="Helvetica 20 bold",bg="#bcdbf7")

self.lblTitle.grid(row =0 ,column = 0,columnspan=2,pady=50)

self.LoginFrame = Frame(self.frame,width=400,height=80,relief="ridge",bg="#4682b4",bd=20)

self.LoginFrame.grid(row=1,column=0)

self.LoginFrame2 = Frame(self.frame,width=400,height=80,relief="ridge",bg="#4682b4",bd=20)

self.LoginFrame2.grid(row=2,column=0)

self.lbloffid = Label(self.LoginFrame,text="ENTER CASE NUMBER TO DELETE",font="Helvetica 14 bold",bg="#4682b4",bd=22)

self.lbloffid.grid(row=0,column=0)

self.lbloffid= Entry(self.LoginFrame,font="Helvetica 14 bold",bd=2,textvariable= self.de1\_case)

self.lbloffid.grid(row=0,column=1)

self.DeleteB = Button(self.LoginFrame2, text="DELETE",width =10,font="Helvetica 14 bold",bg="#4682b4",command = self.DELETE\_OFF)

self.DeleteB.grid(row=3,column=1)

#FUNCTION TO DELETE DATA IN EMPLOYEE FORM

def DELETE\_OFF(self):

de = str(self.de1\_case.get())

conn = mysql.connector.connect(

host='localhost',

user='root',

password='PASSWORD',

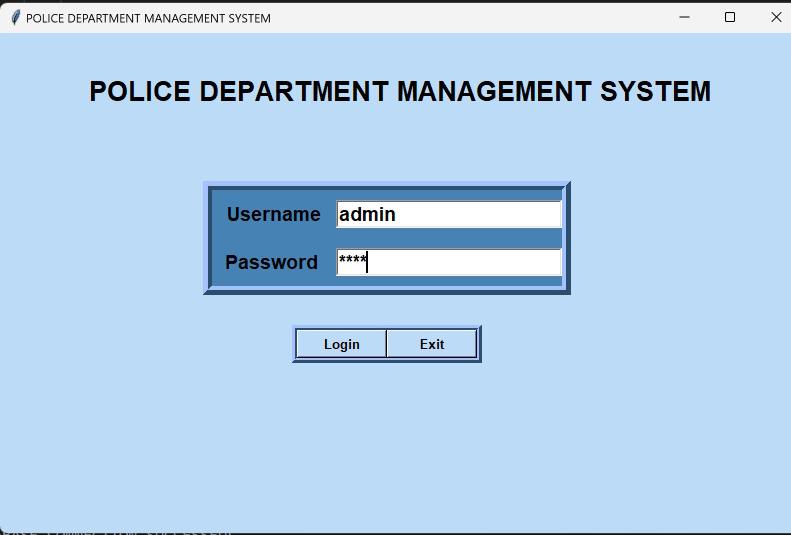
database='police\_dept'

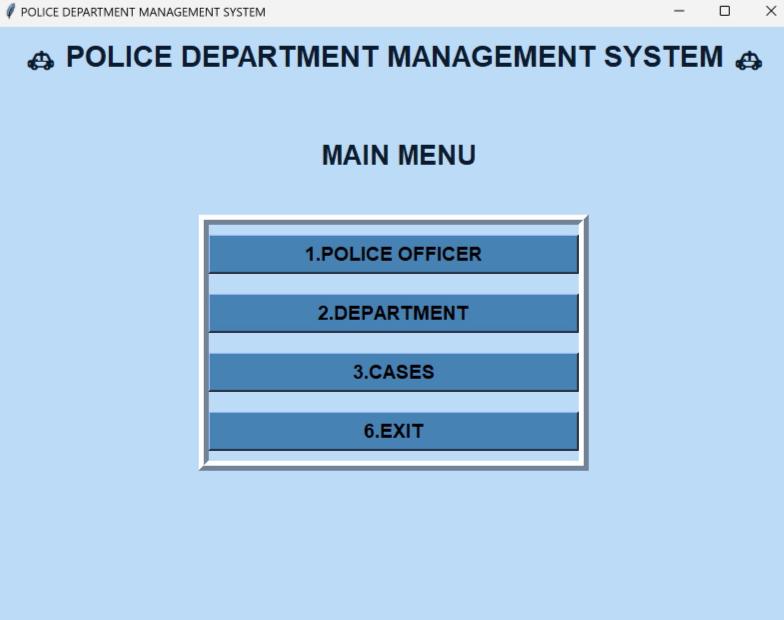
)

my\_cursor = conn.cursor()

**CHAPTER 6**

**SNAPSHOTS**

**Figure 6.1: Police login Window Page**



**Figure 6.2: Police Dashboard Page**



**Figure 6.3: Officer Registration Form**



**Figure 6.4: Police Window Page**

**Figure 6.5: Case Record Window Page**

**CONCLUSION**

Police Station Database Management System is a web-based application for primarily providing training to the employees who provide customized solutions to meet organizational needs intention of the system is the computerization of the existing system. In an existing system all work was done manually with a lot of paperwork involved. The project titled Police Station Management System is exclusively for the authority. It saves a lot of time and manpower and helps the activities to run properly with almost no errors and delay. This application software has been computed successfully and was also tested successfully by taking “test cases”. It is user friendly, and has required options, which can be utilized by the user to perform the desired operations. The software is developed using Python as front end and Sql as back end in Windows environment. The goals that are achieved by the software are:

* Instant access.
* Improved productivity.
* Optimum utilization of resources.
* Efficient management of records.
* Simplification of the operations.
* Less processing time and getting required information.

**REFERENCES**

[1] Herbert Schildt, Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007.

[2] Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.

[3] https ://www.guru99.com/introduction-to- mysql-workbench.html

[4] https://www.youtube.com/watch?v=GODYXAnBDvc

[5] "MySQL 8.0 Reference Manual", Oracle Corporation, <https://dev.mysql.com/doc/refman/8.0/en/>

[6] "Introduction to Relational Database and SQL", Tutorials Point, <https://www.tutorialspoint.com/sql/sql-introduction.html>