# CRIME MANAGEMENT SYSTEM

## A MINI-PROJECT REPORT

*Submitted by*

**NAVEEN PRABHU S K 241901069**

**NEVILLE P 241901072**

*in partial fulfillment of the award of the degree of*

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**RAJALAKSHMI ENGINEERING COLLEGE**

**An Autonomous Institute**
**CHENNAI**
**NOVEMBER 2025**

# BONAFIDE CERTIFICATE

Certified that this project **"CRIME MANAGEMENT SYSTEM"** is the bonafide work of **"NAVEEN PRABHU S K, NEVILLE P"** who carried out the project work under my supervision.

**SIGNATURE**

**Ms. R. RUPMALA**

**ASSISTANT PROFESSOR**

Dept. of Computer Science and Engineering (Cyber

Rajalakshmi Engineering College

Chennai

This mini project report is submitted for the viva voce examination to be held on _____

**INTERNAL EXAMINER** **EXTERNALEXAMINER**

# ABSTRACT

In our state, maintaining law and order plays a major role. Even though there are various national-level crime tracking and reporting systems, the local departments often lack an efficient and hassle-free application to manage crime-related data. To address this drawback in the local law enforcement system, our team developed a database system to help police departments and related agencies maintain and organize crime records efficiently.

The main objective of this project is to record, monitor, and analyze crime incidents to assist in investigation and prevention. This system helps in maintaining details about reported crimes, suspects, victims, and ongoing investigations within a local jurisdiction. By enabling quick access to reliable information and better coordination among officers, this system allows the department to enhance its efficiency and improve public safety.

# ACKNOWLEDGEMENT

We express our sincere thanks to our beloved and honorable chairman **MR. S. MEGANATHAN** and the chairperson DR. **M. THANGAM MEGANATHAN** for their timely support and encouragement.

We are greatly indebted to our respected and honorable principal **Dr. S.N. MURUGESAN** for his able support and guidance.

No words of gratitude will suffice for the unquestioning support extended to us by our Head of The Department Mr. **Benedict Jayaprakash Nicholas** for being ever supporting force during our project work

We also extend our sincere and hearty thanks to our internal guide **Ms. R. RUPMALA,** for her valuable guidance and motivation during the completion of this project.

Our sincere thanks to our family members, friends and other staff members of computer science engineering.

**1.NAVEEN PRABHU S K**

**2.NEVILLE P**

# TABLE OF CONTENTS

**1. INTRODUCTION** **7**

1.1 INTRODUCTION 7

1.2 SCOPE OF THE WORK 7

1.3 PROBLEM STATEMENT 7

1.4 AIM AND OBJECTIVES OF THE PROJECT 8

**2. SYSTEM SPECIFICATIONS** **9**

2.1 HARDWARE SPECIFICATIONS 9

2.2 SOFTWARE SPECIFICATIONS 9

**3. MODULE DESCRIPTION** **10**

**4. CODING** **11**

**5. SCREENSHOTS** **14**

**6. CONCLUSION AND FUTURE ENHANCEMENT** **18**

**REFERENCES** **19**

## LIST OF FIGURES


| F I G U R E NO. | TITLE | PAGE NO. |
|---|---|---|
| 5.1 | LOGIN PAGE | 14 |
| 5.2 | USER REGISTRATION DETAILS | 15 |
| 5.3 | ADMIN CRIME LOG | 16 |
| 5.4 | USER CRIME LOG | 16 |
| 5.5 | ADDITION OF CRIME | 17 |
| 5.6 | USER DETAILS | 17 |

# CHAPTER 1

# INTRODUCTION

## 1. INTRODUCTION

The project helps people and law enforcement authorities to know the necessary information and details of crime records within the locality. It enables users to access and analyze data related to reported crimes, suspects, and investigation status in an organized manner. The necessary information about crime incidents, case progress, and officer assignments will be maintained and displayed according to user convenience. This system ensures that information is easily accessible, accurate, and helps in assessing the safety and crime trends in the area.

## 2. SCOPE OF THE WORK

The Crime Management System will help law enforcement authorities and the public access important crime-related information within their locality. It aims to provide quick and easy accessibility to data regarding reported crimes, suspects, and case updates. This system enhances coordination between departments and ensures that crime information is efficiently managed and readily available for analysis, thereby improving public safety and decision-making.

## 3. PROBLEM STATEMENT

The need for this project arises from the fact that, although national and international crime tracking systems exist, they often do not cater effectively to local areas. Many local police stations and agencies still rely on manual record-keeping methods, making it difficult to access and share crime-related information efficiently. This lack of an organized and accessible system causes delays in investigations, poor data management, and inconvenience to both officers and the public. Hence, there is a strong need for a localized and efficient Crime

Management System to streamline the process and improve law enforcement effectiveness.

## 1.4 AIM AND OBJECTIVES OF THE PROJECT

The main objective of this project is to record, track, and manage crime-related information efficiently within a local area. This system helps to maintain detailed records of reported crimes, victims, suspects, and investigation progress in a structured database. It aims to assist law enforcement agencies in quick data retrieval, better coordination, and effective decision-making. By ensuring accurate and organized information management, this system enables local authorities to improve their efficiency in solving cases and maintaining public safety.

# CHAPTER 2

# SYSTEM SPECIFICATIONS

## 2.1 HARDWARE SPECIFICATIONS

| | | |
|---|---|---|
| Processor | : | Intel Pentium |
| Memory Size | : | 1GB (Minimum) |
| HDD | : | 20MB (Minimum) |

## 2.2 SOFTWARE SPECIFICATIONS

| | | |
|---|---|---|
| Operating System | : | MacOS, Linux, Windows |
| Front – End | : | Java |
| Back - End | : | Java, H2 Embedded |
| Language | : | SQL, Java (Swing for UI) |

# CHAPTER 3

# MODULE DESCRIPTION

This application consists of two modules. When the program runs, it will ask for confirmation to the login window. The person who interacts can login as an Administrator or as a User. The description of the modules is as follows:

1. **Admin login**

   When the person who interacts tries to login as Admin then he needs to login with his username and password. The administrator only has the power to change and manipulate the data in the database.

2. **User login**

   When the person tries to login as a user then he/she will be prompted to enter the number of symptoms and the result will be printed in the form of table.

# CHAPTER 4

# SAMPLE CODING

```
package com.crime;


import java.sql.*;


public class DBConnection {
                    private static final String URL = "jdbc:h2:./
crime_db;DB_CLOSE_DELAY=-1;MODE=MySQL;AUTO_SERVER=TRUE";
    private static final String USER = "sa";
    private static final String PASS = "";


    static {
        try {
            Class.forName("org.h2.Driver");
        } catch (ClassNotFoundException e) {
            System.err.println("H2 Driver not found!");
            e.printStackTrace();
        }
    }


    public static Connection getConnection() {
        try {
            Connection conn = DriverManager.getConnection(URL, USER, PASS);
```

```
            createTablesIfNotExist(conn);
            return conn;
        } catch (SQLException e) {
            System.err.println("H2 Connection Failed!");
            e.printStackTrace();
            return null;
        }
    }

    private static void createTablesIfNotExist(Connection conn) {
        try (Statement stmt = conn.createStatement()) {
            // Users Table
            stmt.execute("""
                CREATE TABLE IF NOT EXISTS users (
                    id INT AUTO_INCREMENT PRIMARY KEY,
                    username VARCHAR(50) UNIQUE NOT NULL,
                    password VARCHAR(100) NOT NULL,
                    role ENUM('ADMIN', 'USER') NOT NULL,
                    name VARCHAR(100),
                    mobile_no VARCHAR(15),
                    email VARCHAR(100),
                    age INT,
                    gender ENUM('Male', 'Female', 'Other')
                )
                """);

            // Crimes Table
            stmt.execute("""
```

```
            CREATE TABLE IF NOT EXISTS crimes (
                id INT AUTO_INCREMENT PRIMARY KEY,
                crime_type VARCHAR(100) NOT NULL,
                location VARCHAR(150),
                date_reported DATE,
                suspect_name VARCHAR(100),
                status VARCHAR(50) DEFAULT 'Pending',
                description TEXT
            )
            """);

        // Insert Default Admin
        var rs = stmt.executeQuery("SELECT COUNT(*) FROM users WHERE username = 'admin'");
        if (rs.next() && rs.getInt(1) == 0) {
                stmt.execute("INSERT INTO users (username, password, role, name, mobile_no, email, age,
gender) " +
                                    "VALUES ('admin', 'admin123', 'ADMIN', 'System Admin', '9999999999',
'admin@crime.com', 30, 'Male')");
            System.out.println("Default admin created: admin / admin123");
        }

        // Insert 15 Sample Crimes
        rs = stmt.executeQuery("SELECT COUNT(*) FROM crimes");
        if (rs.next() && rs.getInt(1) == 0) {
            insertSampleCrimes(conn);
            System.out.println("15 sample crimes inserted!");
        }

    } catch (SQLException e) {
```

```
            e.printStackTrace();
        }
    }

    // FIXED: Use PreparedStatement to avoid SQL injection & escape issues
    private static void insertSampleCrimes(Connection conn) throws SQLException {
        String sql = "INSERT INTO crimes (crime_type, location, date_reported, suspect_name, status, description) VALUES (?, ?, ?, ?, ?, ?)";
        try (PreparedStatement ps = conn.prepareStatement(sql)) {
            addCrime(ps, "Negligent Homicide at Rally", "Karur", "2025-09-27", "TVK Party Leaders", "Under Investigation", "41 killed in stampede at actor Vijay's TVK rally. Police filed case.");
            addCrime(ps, "Unlawful Assembly", "Chennai", "2025-03-17", "K. Annamalai", "Pending", "BJP leaders arrested during TASMAC protest.");
            addCrime(ps, "Murder", "Madurai", "2025-03-19", "Unknown", "Under Investigation", "DMK worker killed in party feud.");
            addCrime(ps, "Assault", "Tiruppur", "2025-04-14", "Unknown", "Closed", "Dalit activist assaulted at political meeting.");
            addCrime(ps, "Murder", "Sivaganga", "2025-03-19", "Unknown", "Pending", "AIADMK worker murdered.");
            addCrime(ps, "Murder", "Coimbatore", "2025-10-19", "Youth", "Under Investigation", "Man stabbed in hospital maternity ward.");
            addCrime(ps, "Murder", "Unnamed Town", "2025-08-28", "Father", "Under Investigation", "2-year-old girl killed by father.");
            addCrime(ps, "Robbery", "Music College", "2025-08-28", "2 Rowdies", "Closed", "Student robbed after honey trap.");
            addCrime(ps, "Murder", "Unnamed Village", "2025-08-28", "Sanjay", "Pending", "Man faked snake bite to plot murder.");
            addCrime(ps, "Murder", "Mayiladuthurai", "2025-07-30", "Vendors", "Under Investigation", "Two youngsters killed by illicit liquor sellers.");
            addCrime(ps, "Murder", "Pudukkottai", "2025-02-15", "Unknown", "Under Investigation", "Anti-quarrying activist Jagabar Ali murdered.");
            addCrime(ps, "Murder", "Unnamed District", "2025-09-12", "Unknown", "Closed", "Dalit techie killed in honour killing.");
```

```
            addCrime(ps, "Murder", "Udumalpet", "2025-09-12", "Unknown", "Under Investigation", "Police officer Shanmugavel hacked to death.");
            addCrime(ps, "Burglary", "Vellore", "2025-09-12", "Thieves", "Pending", "CRPF personnel home burgled.");
                addCrime(ps, "Murder", "Erode", "2025-03-19", "Unknown", "Closed", "Murder in personal dispute.");
        }
    }

    private static void addCrime(PreparedStatement ps, String type, String loc, String date, String suspect, String status, String desc) throws SQLException {
        ps.setString(1, type);
        ps.setString(2, loc);
        ps.setDate(3, Date.valueOf(date));
        ps.setString(4, suspect);
        ps.setString(5, status);
        ps.setString(6, desc);
        ps.executeUpdate();
    }
}
```

# CHAPTER 5

# SCREEN SHOTS

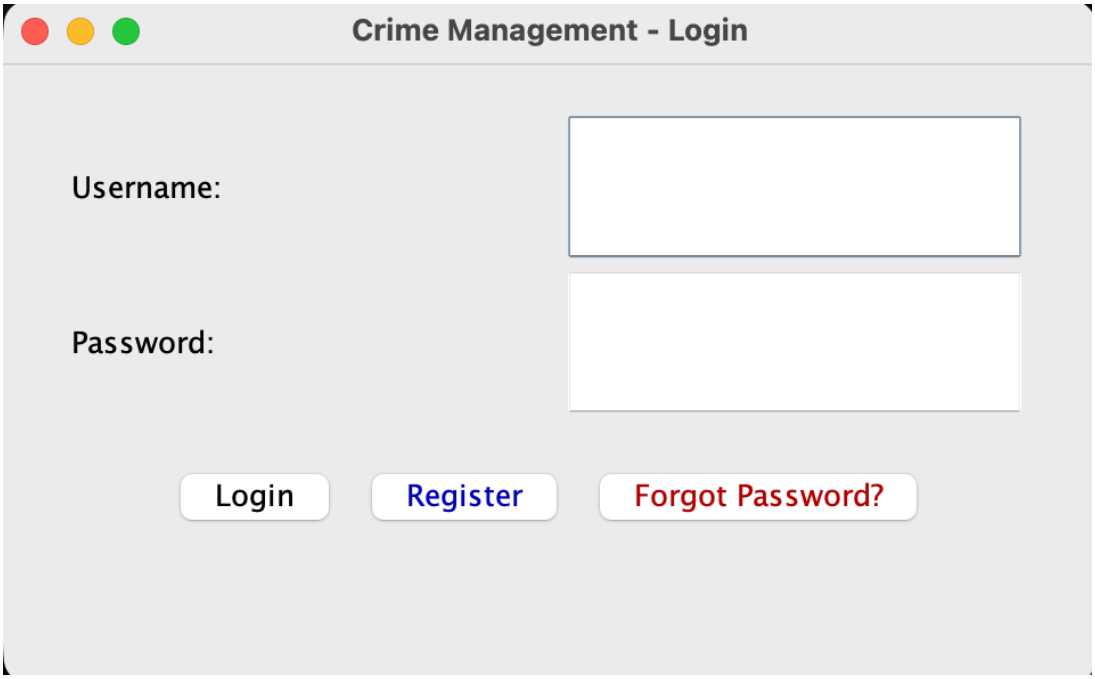


**Fig 5.1 Login page**

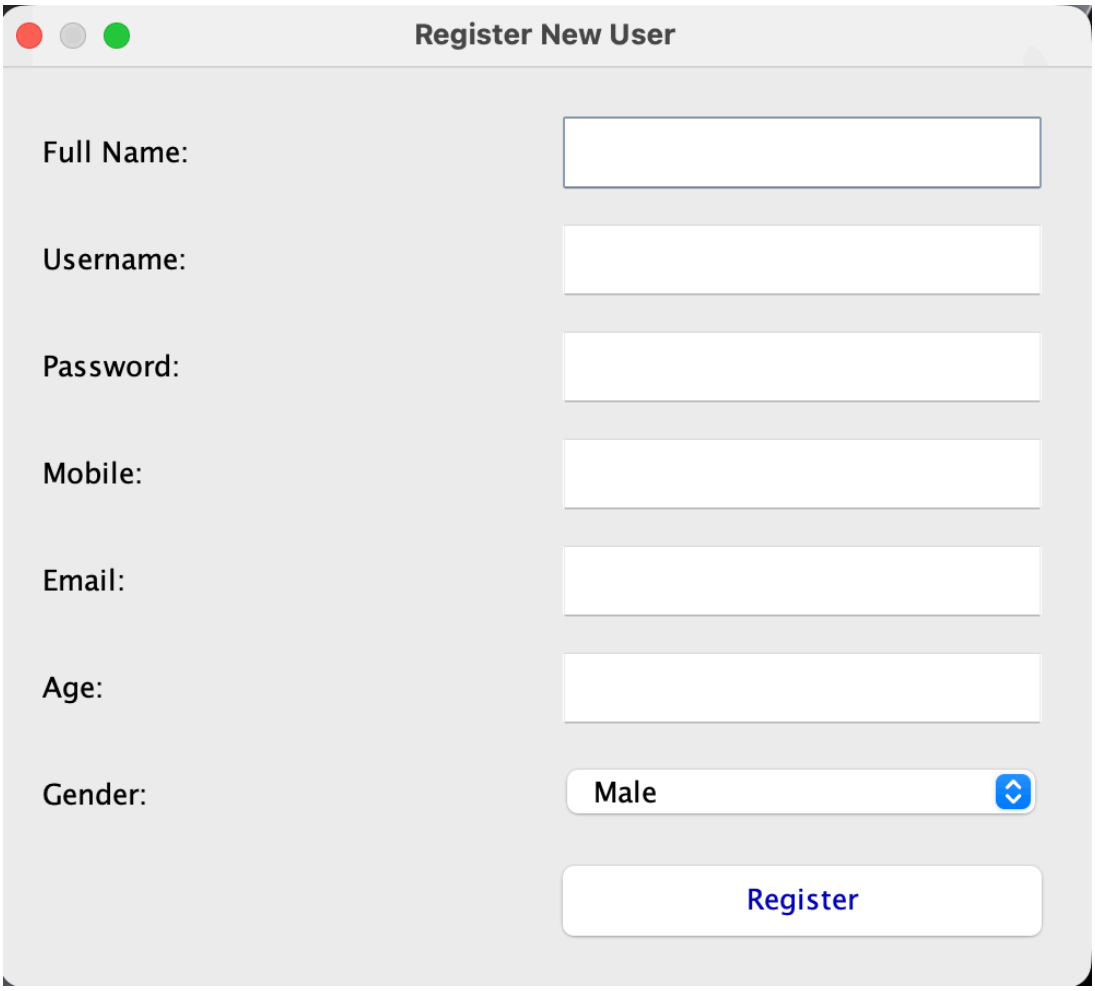


**Fig 5.2 User registration details**

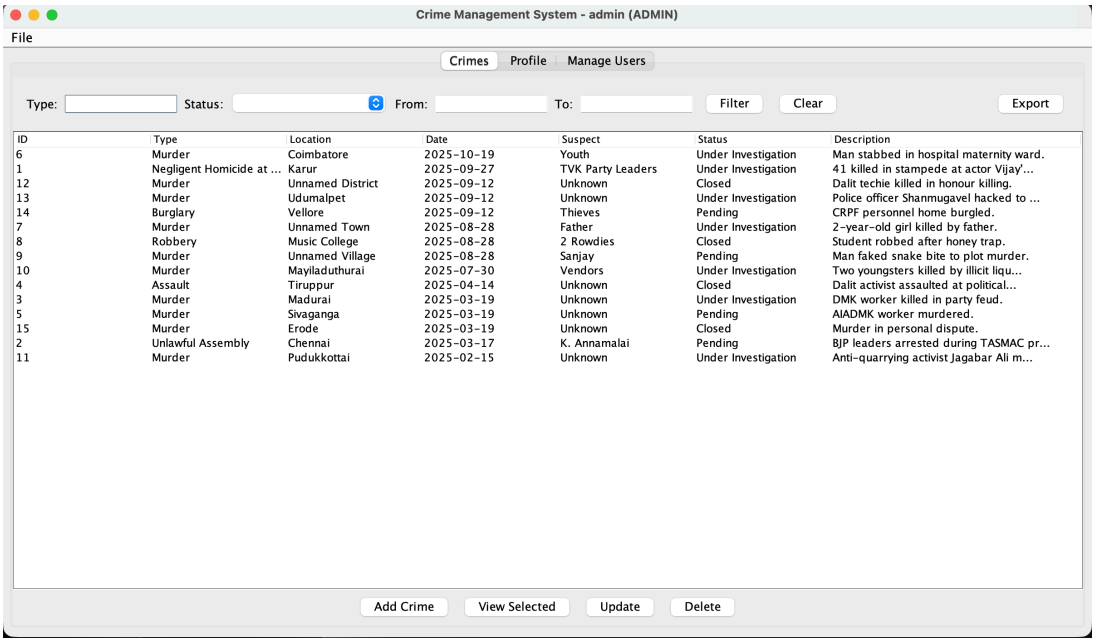


**Fig 5.3 Admin crime Log**

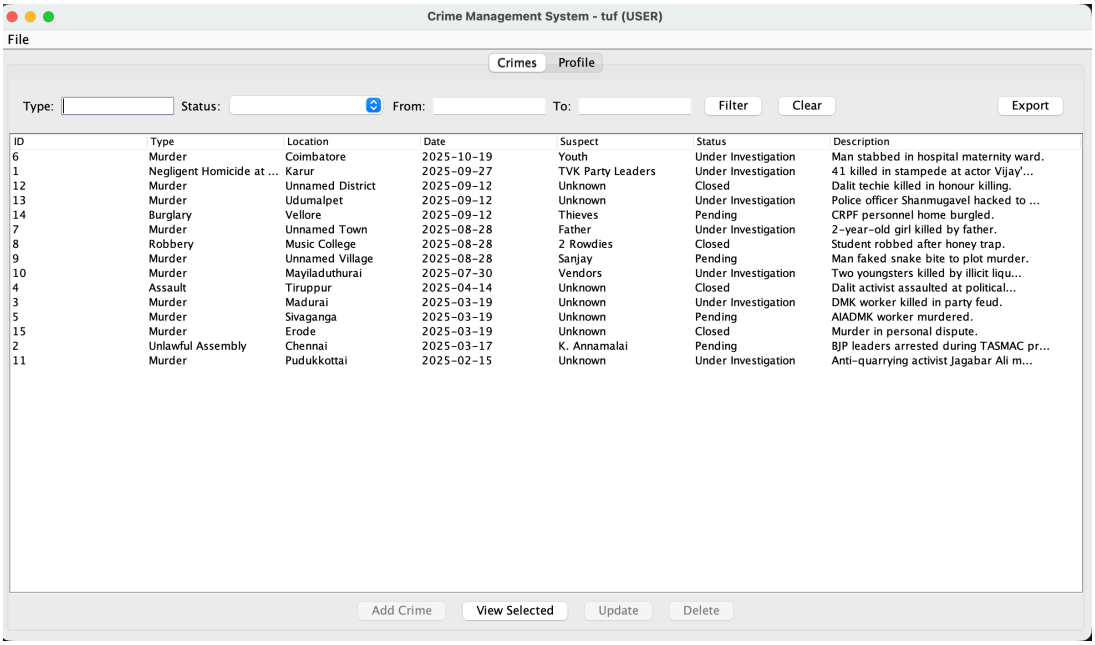


**Fig 5.4 User crime Log**

**Fig 5.5 Addition of Crime**

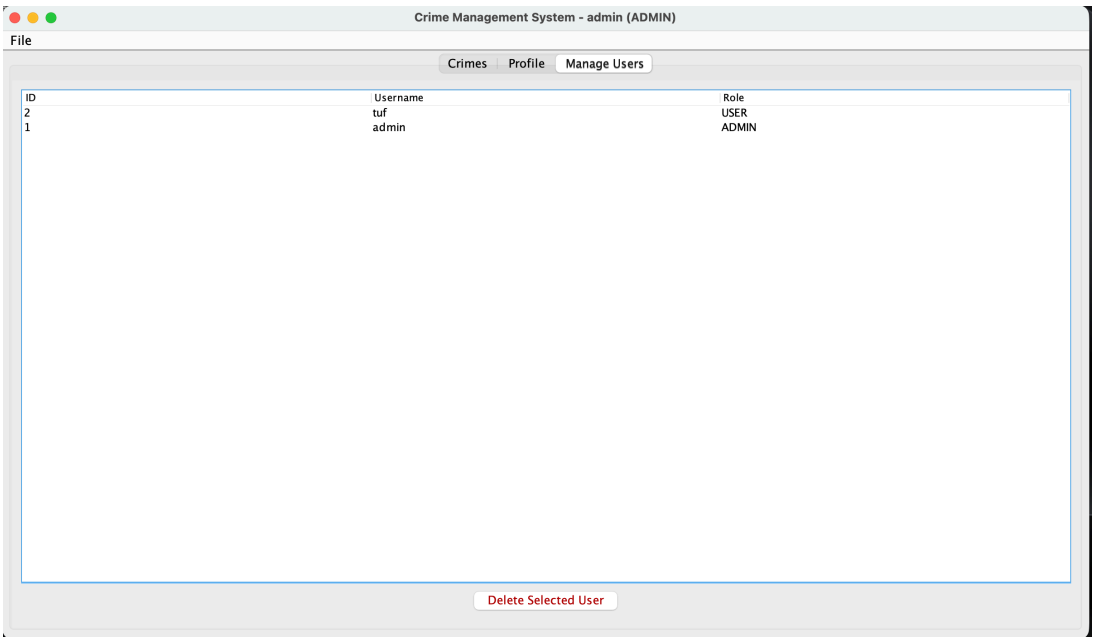


**Fig 5.6 User Details**

# CHAPTER 6

# CONCLUSION AND FUTURE ENHANCEMENT

In such a way, with the help of our project, authorities will be able to efficiently check the list of reported crimes, register new cases, and maintain accurate records of investigations. The system organizes essential data such as details of crimes, suspects, victims, and officers through a structured and secure database, making crime tracking and management more reliable and efficient.

In the future, the system can be further enhanced with advanced features such as **forensic data integration**, allowing digital and physical evidence to be recorded, tracked, and analyzed within the same platform. Additionally, the implementation of a **user activity log secured by blockchain technology** will ensure data transparency, prevent tampering, and enhance accountability among users.

New **database tables** can also be introduced to store forensic reports, digital evidence, and blockchain transaction logs. These additions will improve system security, support better case analysis, and enable inter-departmental collaboration. Hence, the project provides a strong foundation for building a technologically advanced and transparent crime management system that can evolve with modern investigative needs.

# REFERENCES


1. https://www.w3schools.com/sql/
2. https://www.wikipedia.org/
3. https://grok.com/
4. https://chatgpt.com/