

# **AL-AMEEN INSTITUTE OF INFORMATION AND SCIENCES**



Project Title

**Crime File Management System**

Submitted by

**RABE HUSSAINI**  
**[P18AS23S038009]**

Submitted to

**Mr. Azharuddin**

**Professor**

**Department of Computer Science**

**Al-Ameen Institute of Information and Sciences**

# CRIME FILE MANAGEMENT SYSTEM

## 1. ABSTRACT

The **Crime File Management System (CFMS)** is a comprehensive web-based application designed to digitize and streamline the management of criminal records. In traditional law enforcement environments, crime data is often stored in paper files or basic spreadsheets, leading to inefficiencies, data loss, and limited accessibility. CFMS addresses these challenges by offering a centralized, secure, and scalable platform for managing FIRs (First Information Reports), suspect profiles, case details, and digital evidence. Developed using Python with Flask or Django frameworks, and backed by SQLite or MySQL databases, the system ensures robust backend support and seamless frontend interaction. It incorporates role-based access control to maintain data confidentiality and integrity, allowing different levels of access for administrators, officers, and clerks. By digitizing the crime record management process, CFMS enhances operational efficiency, supports faster case resolutions, and promotes transparency and accountability in law enforcement agencies.

## 2. SYSTEM ANALYSIS

### 2.1 LITERATURE SURVEY

Digitization in law enforcement has been a growing trend globally. Studies have shown that digital crime management systems significantly reduce administrative overhead and improve data accuracy. Research papers on e-Governance highlight the importance of centralized databases and secure access for effective policing. For instance, the integration of digital FIR systems in metropolitan police departments has led to a 40% reduction in case processing time.

Literature also emphasizes the role of technology in crime prevention. Systems that incorporate real-time data sharing, facial recognition, and predictive analytics have proven effective in identifying repeat offenders and preventing crimes. The CFMS aligns with these technological advancements by offering a scalable and modular architecture that can integrate future enhancements like biometric verification and mobile patrol apps.

## 2.2 EXISTING SYSTEM

The current systems used in many police stations are either paper-based or rely on basic spreadsheet software. These systems suffer from several limitations:

- **Manual Entry Errors:** High chances of data entry mistakes due to lack of validation.
- **Data Redundancy:** Duplicate records are common due to lack of centralized control.
- **Limited Accessibility:** Records are confined to physical locations and cannot be accessed remotely.
- **Security Risks:** Paper files are vulnerable to theft, fire, and unauthorized access.
- **Inefficient Search:** Finding specific records requires manual scanning, which is time-consuming.

## 2.3 PROPOSED SYSTEM

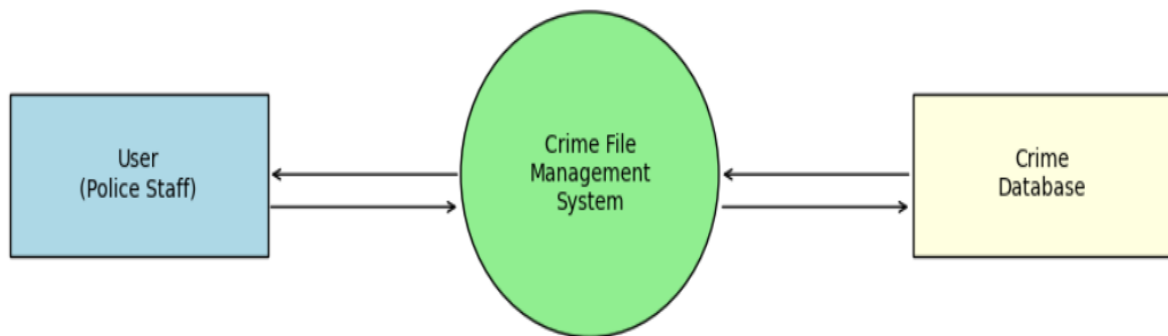
The proposed CFMS offers a robust solution to overcome the limitations of existing systems. Key features include:

- **Web-Based Interface:** Accessible from any device with internet connectivity.
- **Centralized Database:** All records are stored in a secure, centralized repository.
- **Role-Based Access:** Ensures that only authorized personnel can access or modify data.
- **Digital Evidence Management:** Supports uploading and viewing of images, documents, and videos.
- **Advanced Search Filters:** Allows filtering by suspect name, crime type, date, and location.
- **Real-Time Updates:** Officers can update case status instantly, improving coordination.

## 2.4 DATA FLOW DIAGRAMS

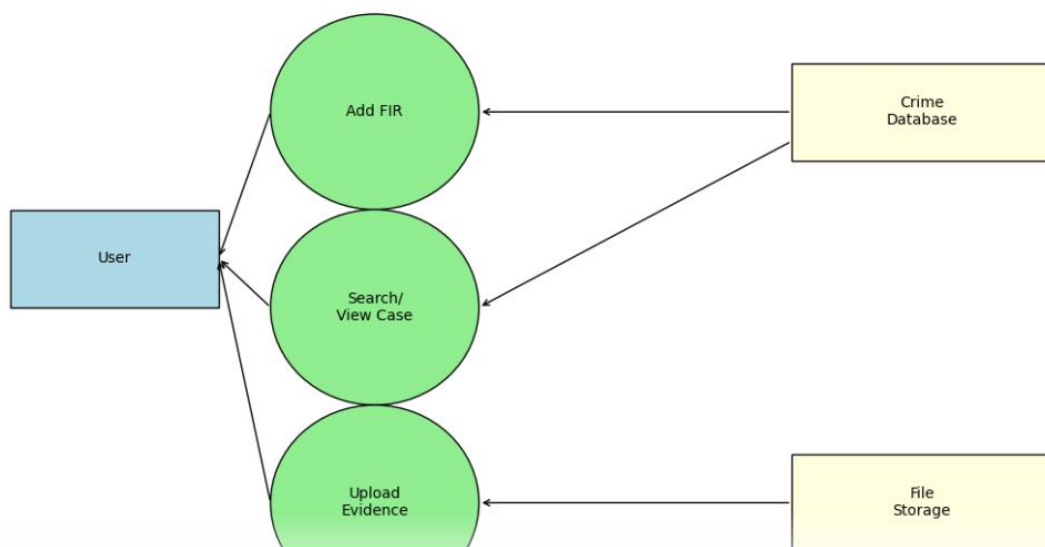
### Level 0 DFD

Depicts the interaction between the user and the system at a high level.



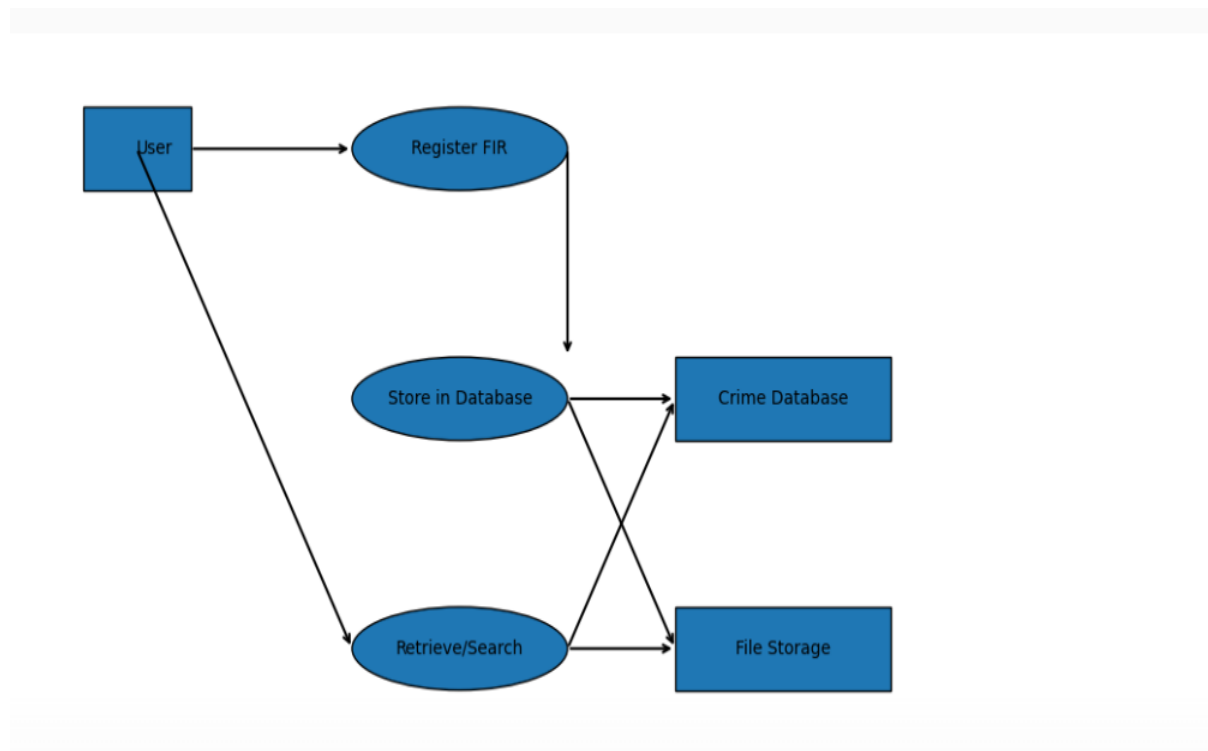
### Level 1 DFD

Breaks down the system into major processes like FIR registration, case search, and evidence upload.



## Level 2 DFD

Provides a detailed view of the FIR registration process, database storage, and retrieval/search functionalities.



## 3. REQUIREMENT ANALYSIS

### 3.1 HARDWARE REQUIREMENTS

To deploy and run the CFMS efficiently, the following hardware specifications are recommended:

- **Processor:** Dual Core or higher (Intel i3/i5 or AMD equivalent)
- **RAM:** Minimum 4 GB (8 GB recommended for better performance)
- **Storage:** At least 1 GB for logs and evidence files; scalable based on usage
- **Webcam:** Optional for capturing suspect images or evidence
- **Internet Connectivity:** Required for accessing the system remotely and syncing data

### 3.2 SOFTWARE REQUIREMENTS

The software stack includes:

- **Operating System:** Windows 10+, Linux (Ubuntu), or macOS
- **Programming Language:** Python 3.8+
- **Framework:** Flask or Django for backend development
- **Database:** SQLite for lightweight deployments; MySQL for scalable environments
- **Frontend Technologies:** HTML5, CSS3, JavaScript (Bootstrap for UI)
- **Browser:** Chrome, Firefox, or Edge
- **IDE:** Visual Studio Code, PyCharm, or any Python-compatible editor

## 4. SYSTEM DESIGN

### 4.1 ARCHITECTURE OVERVIEW

The system follows a three-tier architecture:

- **Presentation Layer:** Built using HTML, CSS, and JavaScript. It handles user interactions and displays data.
- **Application Layer:** Developed in Python using Flask/Django. It processes user requests, performs validations, and interacts with the database.
- **Data Layer:** SQLite/MySQL stores all structured data including FIRs, suspects, and evidence metadata.

### 4.2 FUNCTIONAL MODULES

#### 1. Login and Authentication

- Role-based login (Admin, Officer, Clerk)
- Secure password storage using hashing
- Session management

#### 2. FIR Registration

- Input fields for victim/suspect details
- Crime type, description, location, and time
- Attach digital evidence

### 3. Suspect Management

- Add/edit suspect profiles
- Link suspects to FIRs
- View criminal history

### 4. Case Tracking

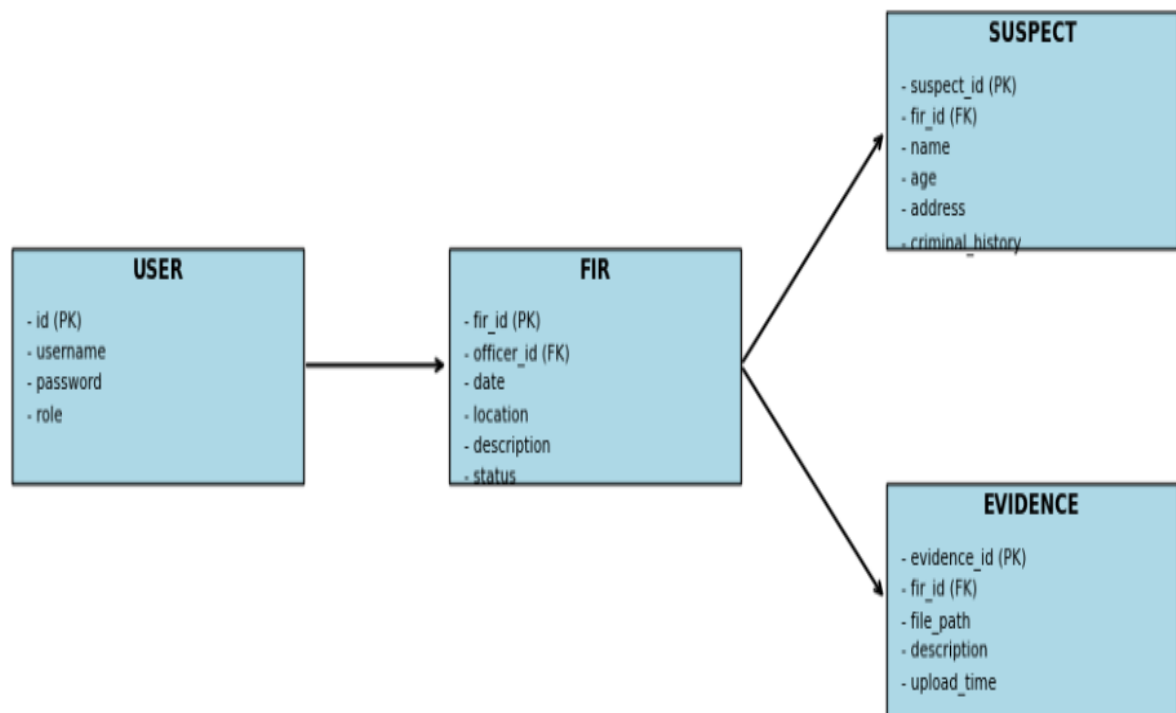
- Monitor case status (Open, Under Investigation, Closed)
- Update progress notes
- Assign officers

### 5. Search & Reports

- Filter by name, date, crime type, location
- Generate printable reports
- Export data to CSV/PDF

## 4.3 ENTITY RELATIONSHIP DIAGRAM (ERD)

The ERD illustrates the relationships between key entities in the system.



## 5. IMPLEMENTATION

The implementation of the Crime File Management System involves setting up the development environment, designing the database schema, building the backend logic, and integrating the frontend interface.

### 5.1 Development Environment Setup

- **Python Installation:** Python 3.8+ is installed along with pip for package management.
- **Framework Selection:** Flask is chosen for its lightweight nature and ease of use. Django can be used for larger deployments.
- **Database Configuration:** SQLite is used for local testing; MySQL is configured for production.
- **Frontend Tools:** Bootstrap is used for responsive design, along with HTML5, CSS3, and JavaScript.

### 5.2 Backend Implementation

- **Routing:** Flask routes handle HTTP requests for login, FIR registration, suspect management, and evidence upload.
- **Database Models:** SQLAlchemy is used to define models for User, FIR, Suspect, and Evidence.
- **Authentication:** Flask-Login manages user sessions and role-based access.

### 5.3 Frontend Implementation

- **Templates:** Jinja2 templates render dynamic content.
- **Forms:** HTML forms collect user input for FIRs and suspects.
- **Validation:** JavaScript and Flask-WTF ensure input validation.

### 5.4 Evidence Handling

- **File Uploads:** Evidence files are uploaded and stored in a designated directory.
- **Metadata Storage:** File paths and descriptions are stored in the database.



## 6. BENEFITS

The Crime File Management System offers numerous advantages:

- **Centralized Data Storage:** All records are stored in a unified database.
- **Improved Accessibility:** Officers can access data from any location.
- **Enhanced Security:** Role-based access and encrypted credentials.
- **Digital Evidence Support:** Upload and view images, documents, and videos.
- **Faster Case Resolution:** Quick access to suspect history and case details.
- **Reduced Paperwork:** Eliminates manual filing and physical storage.
- **Data Backup:** Prevents data loss through automated backups.

## 7. FUTURE ENHANCEMENTS

To further improve the system, the following enhancements are proposed:

- **Mobile App Integration:** Allow officers to register FIRs and upload evidence from mobile devices.
- **Aadhaar Integration:** Link suspect profiles with national ID databases for verification.
- **Facial Recognition:** Use OpenCV to identify suspects from images.
- **Real-Time Alerts:** Notify officers of repeat offenders or high-risk individuals.
- **Analytics Dashboard:** Visualize crime trends and generate insights.

## 8. CONCLUSION

The Crime File Management System is a transformative solution for modern law enforcement. By digitizing the process of crime record management, it enhances efficiency, transparency, and security. The system supports real-time updates, digital evidence handling, and advanced search capabilities, making it a valuable tool for police departments.

Its modular design allows for future scalability, including mobile integration and biometric verification. With proper deployment and training, CFMS can significantly improve the speed and accuracy of criminal investigations, ultimately contributing to a safer society.