SOFTWARE DESIGN DOCUMENT

FOR

NYAY MITRA

Prepared by: Ajina A-11 ,Devakrishna S-70 ,Arshak Muhammed –72,Sayooj V- 76

DATE:24-03-2025

<TKM College of Engineering>

Table of Contents

1. Introduction

1.1 Purpose

Our AI-driven legal app, powered by **Gemini**, is designed to assist legal officers by automating the process of identifying relevant legal sections for complaints. By analysing complaint details, the app quickly provides accurate legal references, reducing the time and effort spent on manual research. This ensures greater efficiency, minimizes errors in section identification, and supports informed decision-making. With its ability to streamline legal research, the app serves as a valuable tool for legal professionals, enhancing their accuracy and productivity in handling cases.

1.2 Scope

The AI-driven legal app is designed to assist legal officers by providing instant legal references based on complaint details. Its scope includes:

- **Legal Section Identification:** Automatically analyzing complaints and suggesting relevant legal sections.
- **AI-Powered Legal Research:** Utilizing Gemini's AI capabilities to enhance accuracy and efficiency.
- **Multi-Jurisdictional Support:** Customizable for different legal systems based on region and jurisdiction.
- **User-Friendly Interface:** Ensuring accessibility for legal officers, law enforcement, and legal researchers.
- **Case Documentation Assistance:** Helping legal professionals organize and reference applicable laws effectively.
- **Scalability & Integration:** Potential to integrate with existing legal databases and case management systems.

1.3 Definitions, Acronyms, and Abbreviations

- **AI (Artificial Intelligence):** Simulation of human intelligence for data analysis and decision-making.
- **NLP (Natural Language Processing):** AI technique for analyzing and understanding human language.
- **API (Application Programming Interface):** A set of functions allowing communication between software components.
- **ML (Machine Learning):** A subset of AI that enables systems to learn from data.
- **DBMS (Database Management System):** A system used to store, retrieve, and manage data.

1.4 References

1. Harvard Law Review – AI in Legal Research (https://harvardlawreview.org/)
2. Google Gemini AI Documentation (https://ai.google.dev/)
3. Indian Kanoon Legal Database (https://www.indiankanoon.org/)

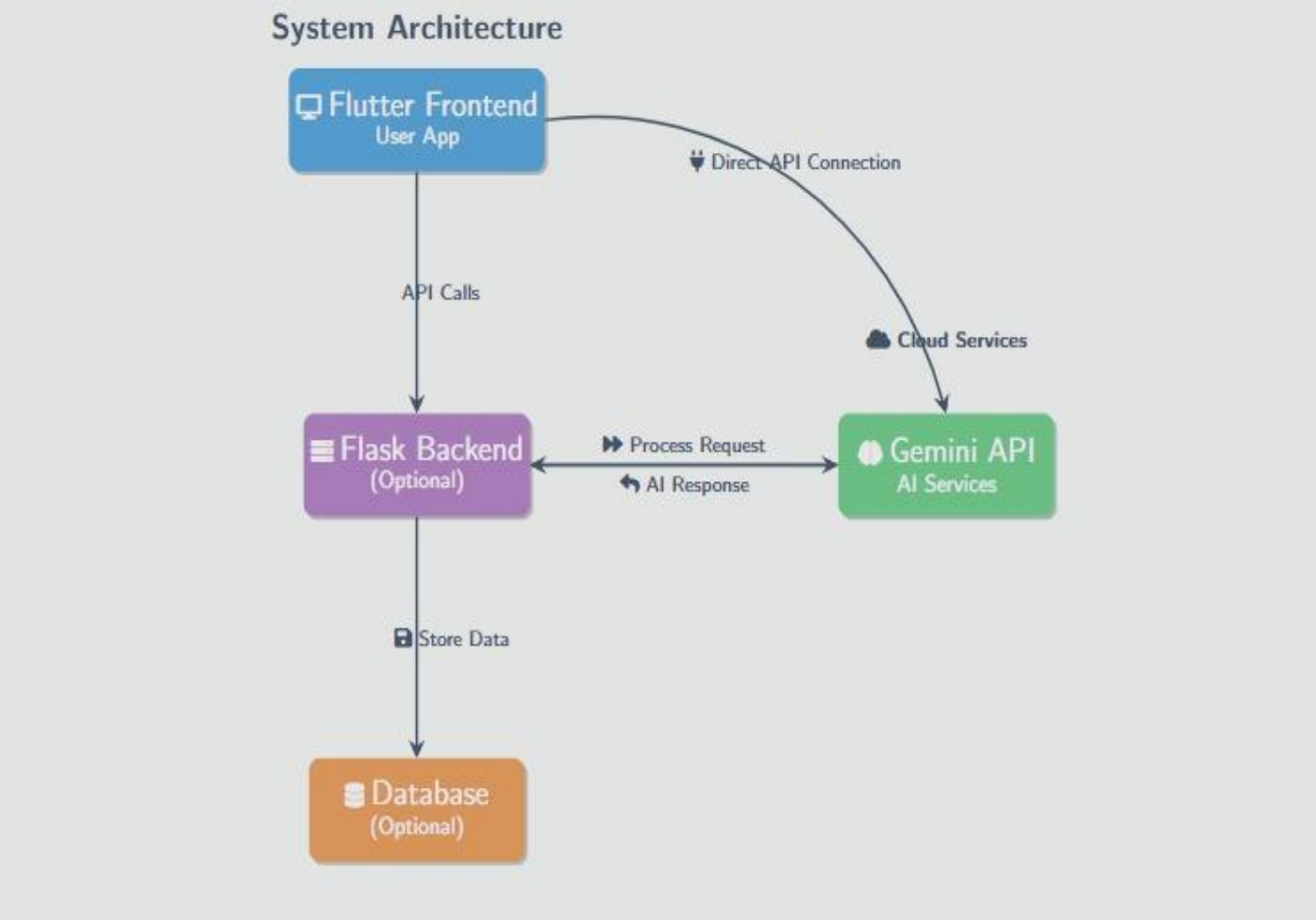
2. System Overview

2.1 System Description

The app takes complaint details as input and uses NLP and AI techniques to extract relevant legal sections. The results are presented to the legal officer in a structured and easy-to-understand format.

2.2 System Architecture

Nyay Mitra is a legal assistance application that leverages Google Gemini API for legal text processing. The system follows a modular client-server architecture, integrating a Flutter-based frontend with a Python-Flask backend, and directly interacting with Gemini for natural language understanding and legal query resolution.



2.3 System Components

1. Frontend (User Interface)

Technology: Flutter

Platform: Android, iOS, and Web

Features:

Text Input – Users can manually enter legal queries.

Voice Input – Converts speech to text and processes it.

Image Input – Extracts text from images and analyzes legal content.

News Section – Fetches legal news using the News API.

User Interaction – Displays responses received from Gemini.

2. Backend (Processing & API Handling)

Technology: Python (Flask)

Role: Acts as a middleware between the frontend and Gemini API

Functions:

Query Processing – Sends user queries (text/voice/image) to Gemini.

Text Extraction – Uses OCR for extracting text from image inputs.

Legal Analysis – Interprets Gemini's response and formats results.

Logging & Monitoring – Tracks queries for debugging and analytics.

3. External APIs & Services

Google Gemini API – Handles NLP-based legal query understanding and text/image analysis.

Google Speech-to-Text API – Converts voice input into text for processing.

News API – Fetches real-time legal news and updates.

4.Data Flow Overview

User Input (Text, Voice, or Image)

Frontend Sends Request

Backend Processes Request

Gemini API Analyzes & Returns Response

Backend Formats Response

Frontend Displays Results

5.Advantages of the Architecture

Lightweight & Scalable – Uses Google's APIs instead of maintaining a local legal database.

Cross-Platform – Works on mobile and web using Flutter.

Minimal Backend Complexity – Only processes and forwards requests, reducing overhead.

This architecture ensures efficient, real-time legal query processing while keeping the system lightweight and easy to maintain

3. Functional Requirements

3.1 User Roles & Permissions

- **Legal Officer:** Can input complaints and receive legal section suggestions.
- **Administrator:** Manages user access and system configurations.
- **System:** Automates complaint analysis and section retrieval.

3.2 Use Case Scenarios

1. **Legal Officer Inputs Complaint Details** → System extracts and displays relevant legal sections.
2. **User Searches for a Specific Legal Section** → System retrieves and displays information.
3. **Administrator Manages Database & Configurations** → Updates laws and policies in the system.

4. Non-Functional Requirements

4.1 Performance Requirements

- The system should process complaints within 5 seconds.
- AI model should have an accuracy rate of at least 90%.

4.2 Security Requirements

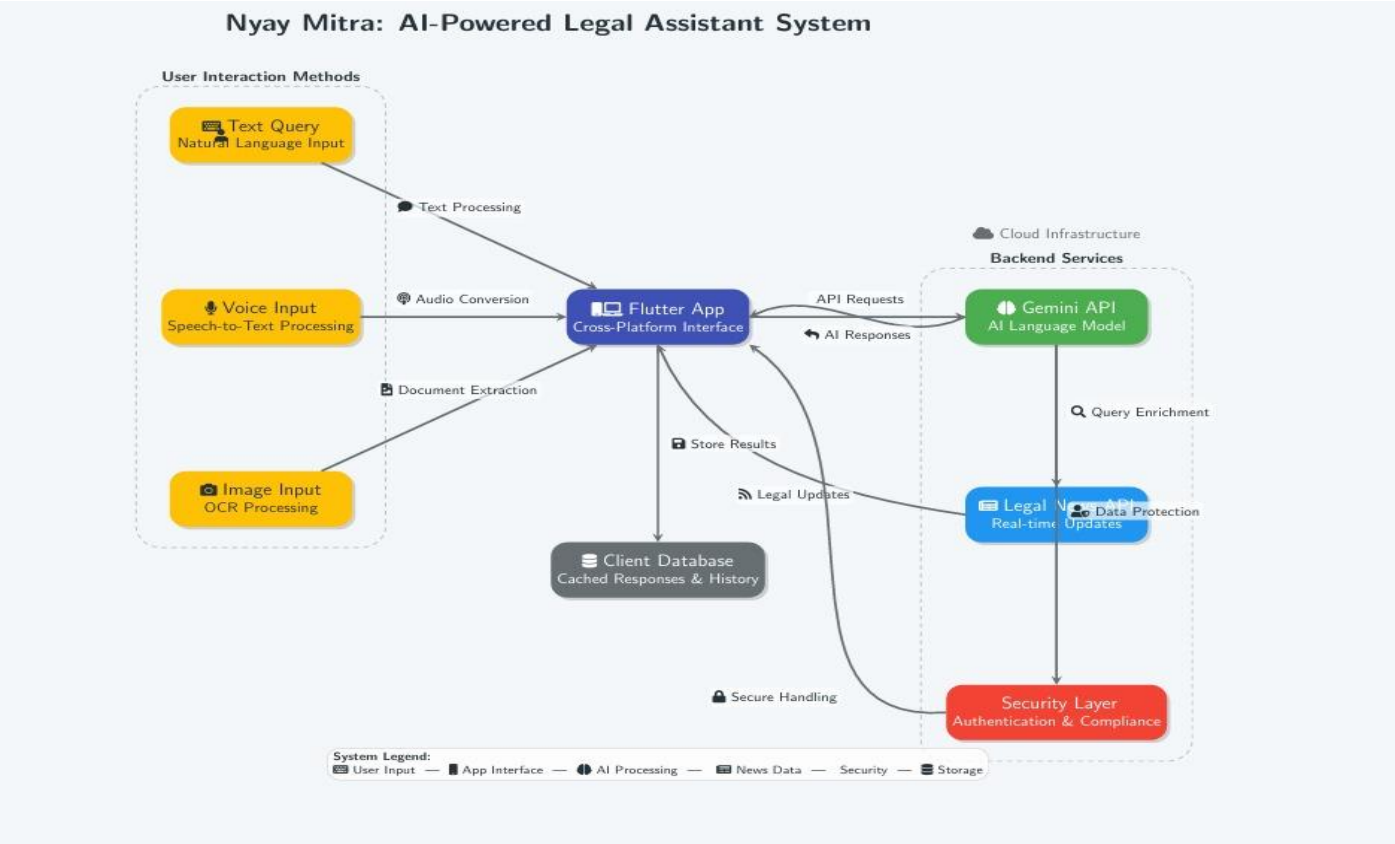
- Data encryption for all user inputs and retrieved legal information.
- Role-based access control (RBAC) for user management.

4.3 Scalability & Availability

- Cloud-based infrastructure for high availability and fault tolerance.
- Support for thousands of concurrent users.

5. System Design

5.1 Architecture Diagram



5.2 Database Schema

Tables:

1. **Users:** Stores user details and roles.
2. **Complaints:** Stores complaint texts and metadata.
3. **Legal_Sections:** Stores legal provisions and their descriptions.
4. **Logs:** Maintains logs for AI model interactions.

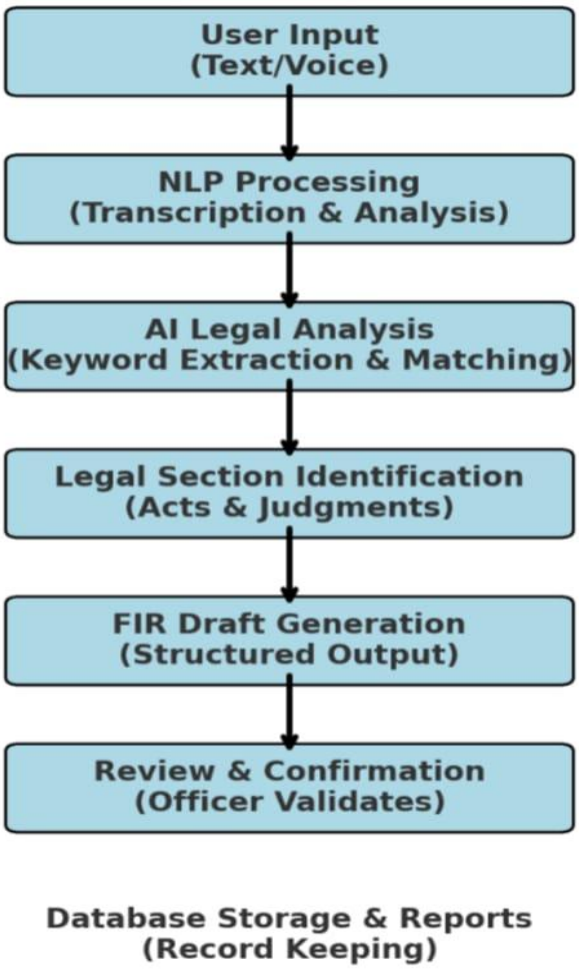
6. Technologies Used

- **Frontend:** React, Angular
- **Backend:** Python (Django/FastAPI)
- **Database:** PostgreSQL/MySQL
- **AI/ML:** Google Gemini API
- **Hosting:** AWS/Azure/GCP

7. Deployment Strategy

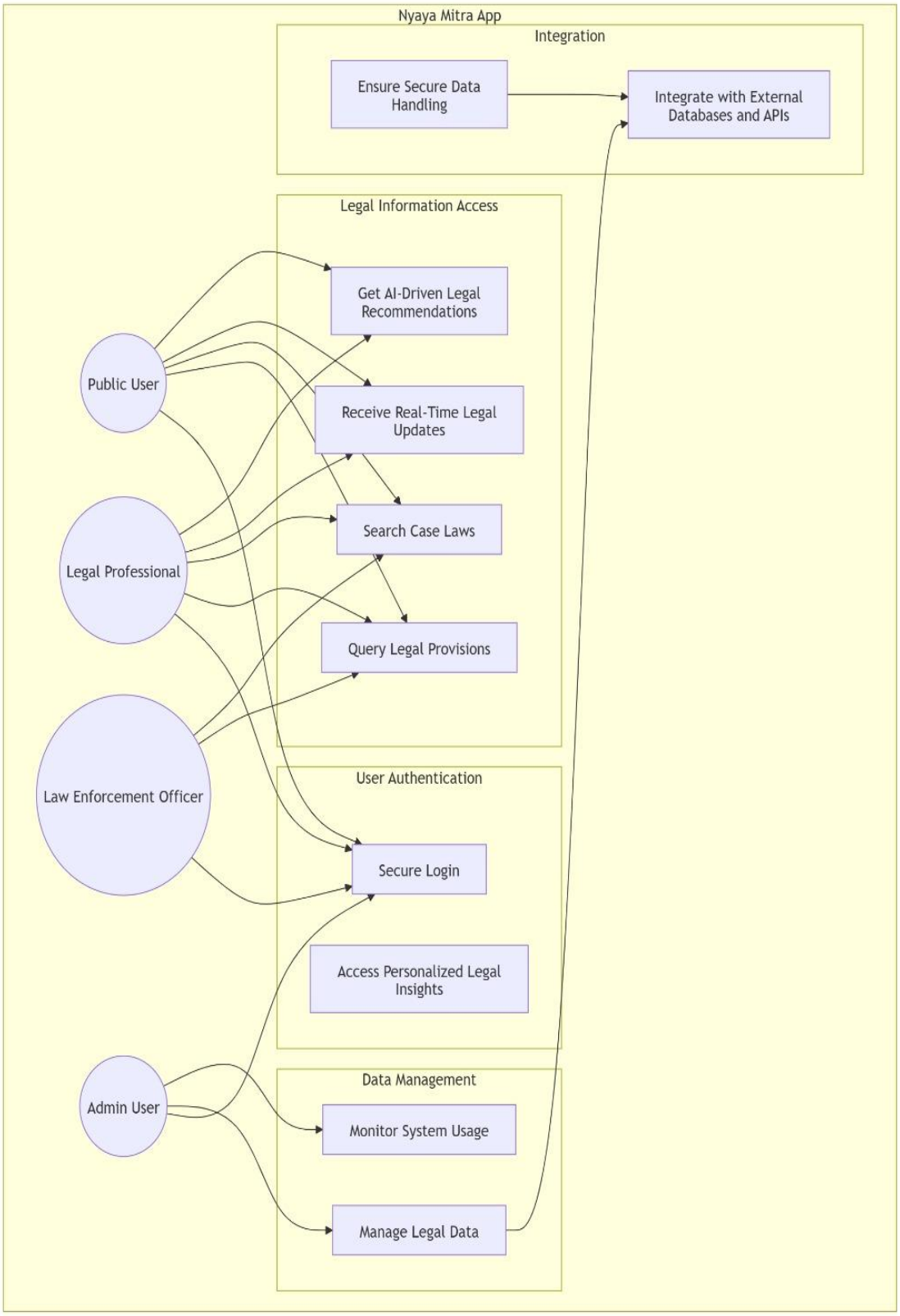
- **Development Phase:** Local and test environment setup.
- **Staging Phase:** Testing on cloud infrastructure.
- **Production Phase:** Deployment with monitoring and security controls.

8.Work Flow Diagram

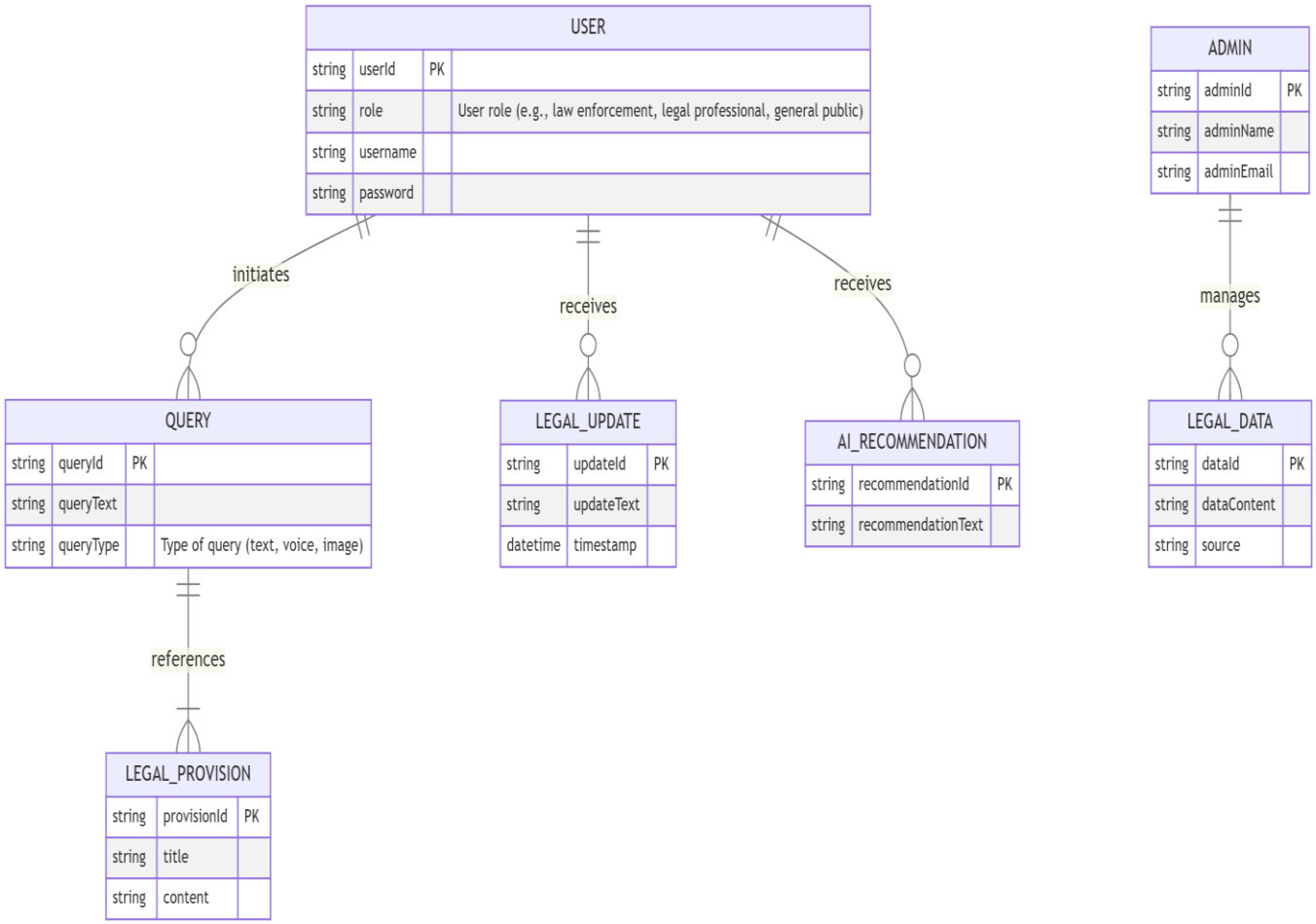


9.System Models

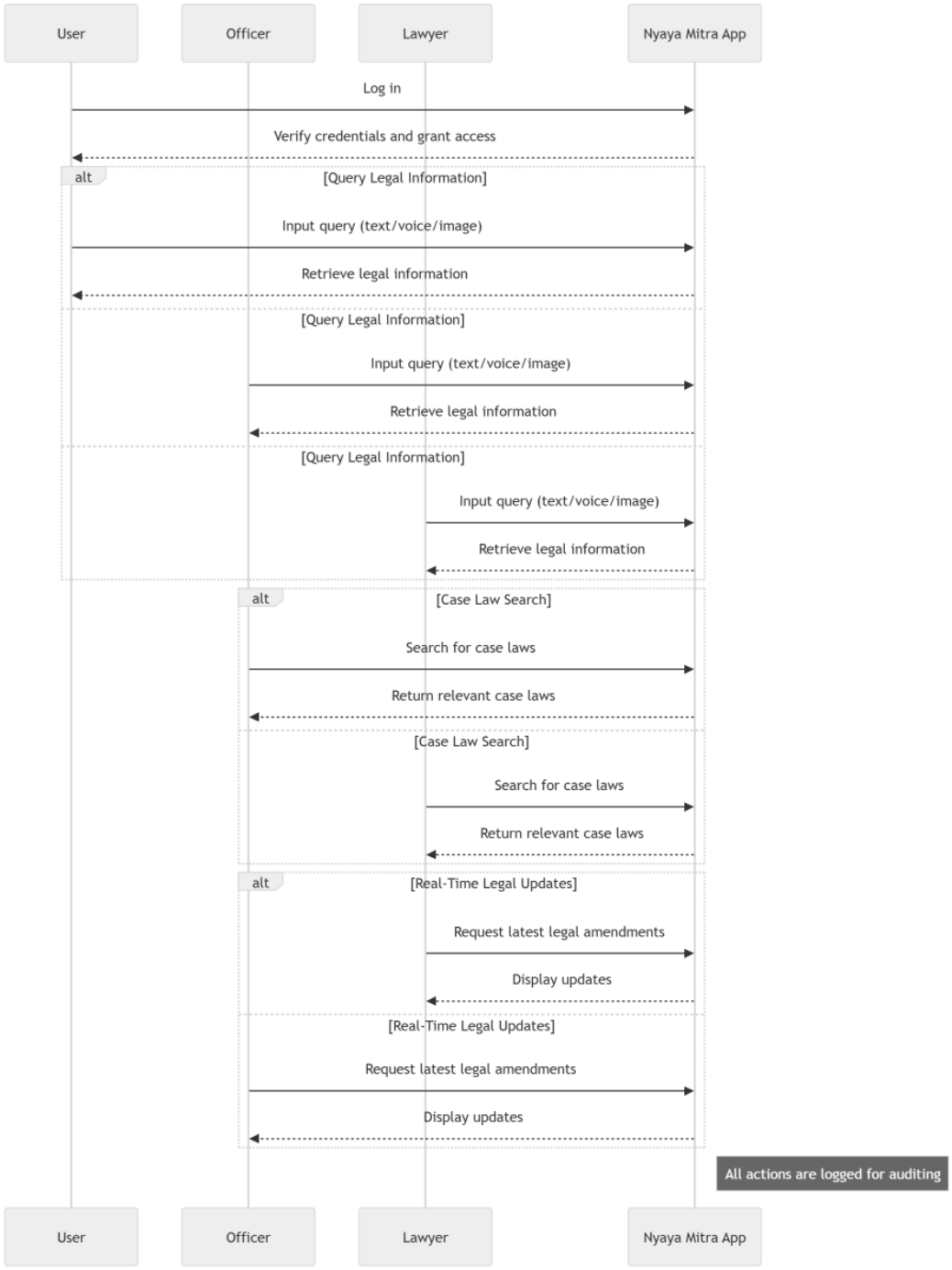
9.1 Use Case Diagram


Nyaya Mitra App
Integration
Ensure Secure Data Handling
Integrate with External Databases and APIs
Legal Information Access
Get AI-Driven Legal Recommendations
Public User
Receive Real-Time Legal Updates
Search Case Laws
Legal Professional
Query Legal Provisions
Law Enforcement Officer
User Authentication
Secure Login
Access Personalized Legal Insights
Admin User
Data Management
Monitor System Usage
Manage Legal Data

9.2 ER Diagram



9.3 Sequence Diagram



10. Future Enhancements

- Multi-language support for different legal systems.
- Advanced AI-based legal document summarization.
- Chatbot integration for real-time legal query resolution.

11. Conclusion

The AI-driven legal app aims to revolutionize legal research by automating section identification, enhancing efficiency, and improving accuracy for legal professionals. Its scalable and modular design ensures adaptability for future legal advancements.