CrimeLens

Crime Pattern Prediction and Visualization Platform

Guide :

Assit. Prof. Reshma S

Team Members :

70 - Rajeev R
61 - Vaisakh V
69 - Phoenix Lal P T

INTRODUCTION

The fundamental issue across many domains, from urban planning to law enforcement, is the vast chasm between acquiring large volumes of raw data and deriving timely, actionable intelligencefrom it. Traditional, static analytical methods are often overwhelmed by complexity, preventing organizations from making proactive, data-informed decisions. Our project delivers a novel platform designed to resolve this: it automatically ingests, processes, and analyzes raw datasets, immediately generating descriptive and predictive insights.

Objectives

1. Advanced Crime Analysis and Hotspot Detection

Develop an advanced crime analysis platform that identifies multi-density hotspots, spatial clusters, and temporal crime trends using optimized K-means clustering and machine learning techniques to support data-driven decision-making for law enforcement, researchers, and policymakers.

Objectives

2. Interactive Visual Tools and AI-Powered Safety Guidance

Empower communities and users with interactive visual tools, including heatmaps, time-series charts, and dashboards, that highlight localized crime risks and promote proactive safety measures through AI-generated contextual advice.

Objectives

3. Data Preprocessing, Severity Classification, and Predictive Insights

Empower communities and users with interactive visual tools, including heatmaps, time-series charts, and dashboards, that highlight localized crime risks and promote proactive safety measures through AI-generated contextual advice.

Objectives

4. Scalable Web-Based Architecture and Future Extensions

Build a scalable, web-based, containerized system with modular architecture that supports large datasets, multi-region adaptability, and future extensions such as deep learning-based risk forecasting and human mobility integration.

Challenges and Issues

1. Scalability and In-Memory Processing

Issue: Using an in-memory Pandas DataFrame in FastAPI (df_storage) is fast for a single user but highly vulnerable to memory limits when handling large, real-world crime datasets (millions of rows). Repeated processing of large files on API calls is slow. Impact: Risk of HTTPException 500 (out-of-memory errors) or long latency for model training/filtering.

Challenges and Issues

2. Real-Time Geospatial Visualization

Issue: Rendering thousands of geographical points for both the Heatmap (leaflet.heat) and individual Hotspot Markers on the client-side (React/Leaflet) often causes UI freezing or degraded performance, when the full dataset is loaded. Impact: Poor user experience; visual elements render too slowly after filter changes.

Challenges and Issues

3. Contextual AI Grounding in Streaming

Issue: The Gemini API response is streamed for better perceived performance. However, ensuring the generated safety advice is strictly grounded in the dynamic, user-defined crime context (e.g., only "Theft" and "Burglary") requires a highly precise and robust system prompt within a streaming architecture. **Impact:** Risk of AI hallucination or generic, irrelevant safety tips, reducing the value of the assistant.

ML Endpoints and Their Functionalities:

- **trainModel**

Purpose: Crime severity prediction.
Functionality: Trains an XGBoost classifier on the currently filtered dataset. Returns the model's accuracy and feature importance for visualization.

- **getSeverityBreakdown**

Purpose: Severity distribution analysis.

Functionality: Retrieves data for Pie Charts and stacked Bar Charts, showing High, Medium, and Low crime counts across different areas.

- **getTimeSeries**

Purpose: Temporal crime trend analysis.

Functionality: Fetches historical monthly crime counts and a 12-month forecast using Prophet. Supports confidence interval visualization and forecast toggling.

ML Endpoints and Their Functionalities:

- **getHotspots**

Purpose: Spatial hotspot detection.
Functionality: Applies K-Means clustering to identify multi-density crime hotspots, returning coordinates and intensity for heatmaps.

- **uploadFile**

Purpose: Data ingestion and preprocessing.

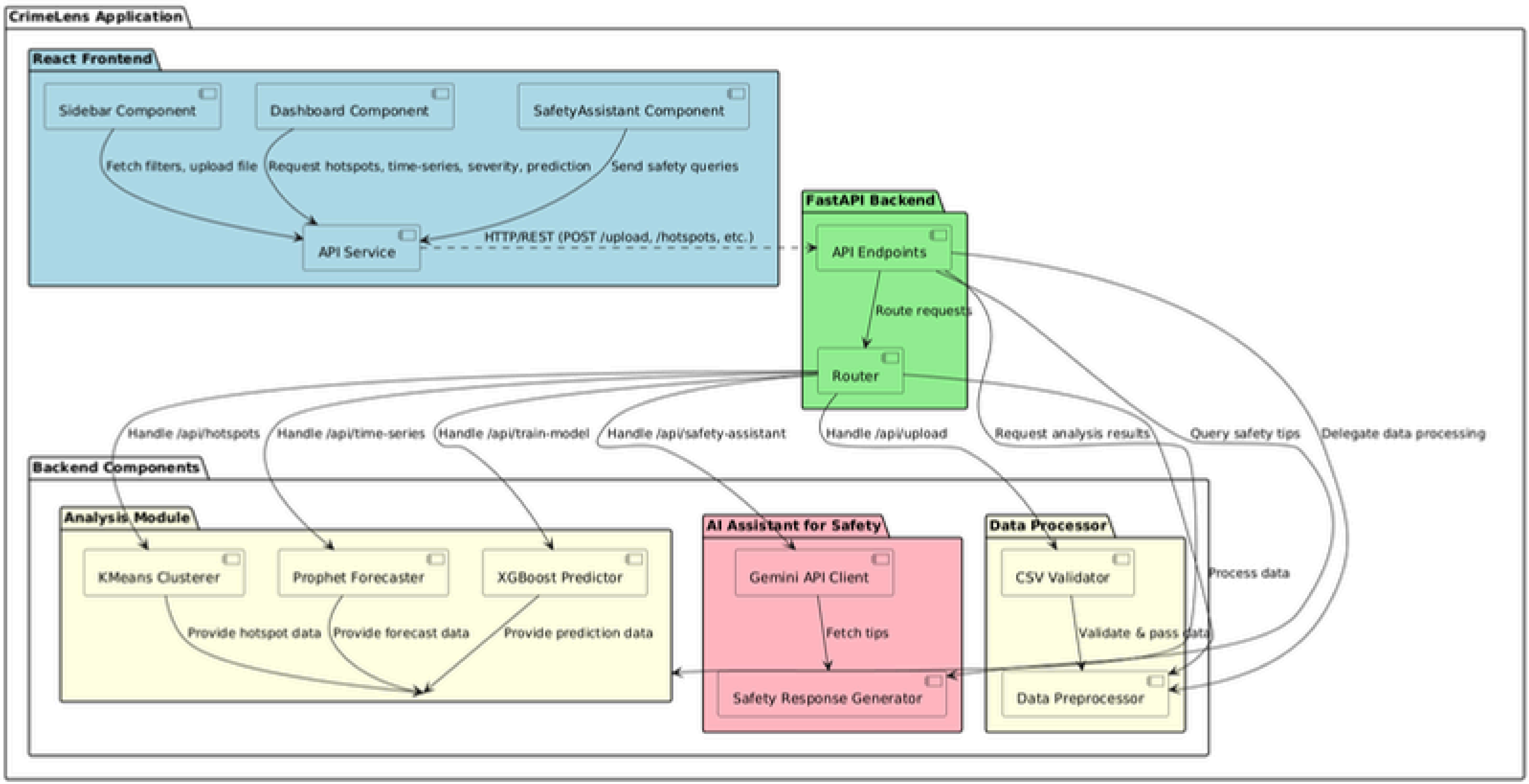
Functionality: Accepts CSV uploads, validates the format, preprocesses the data (cleaning, encoding, scaling, feature selection), and generates initial filters for Areas, Crime Types, and Severities.

- **SafetyAssistant / Gemini API**

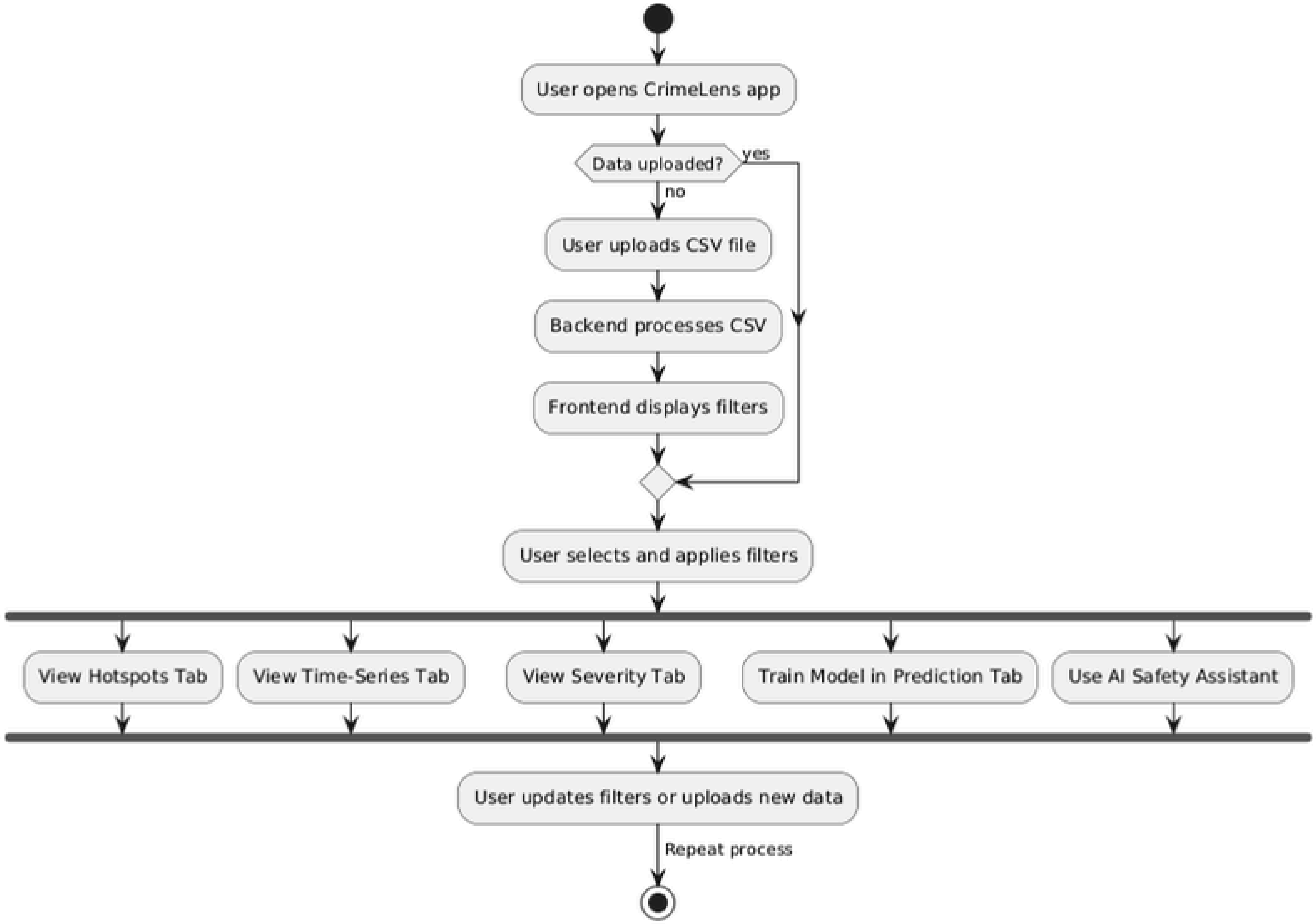
Purpose: AI-driven safety recommendations.

Functionality: Provides real-time, context-aware safety tips based on the top crime types from the current dataset view, displayed via a chat widget.

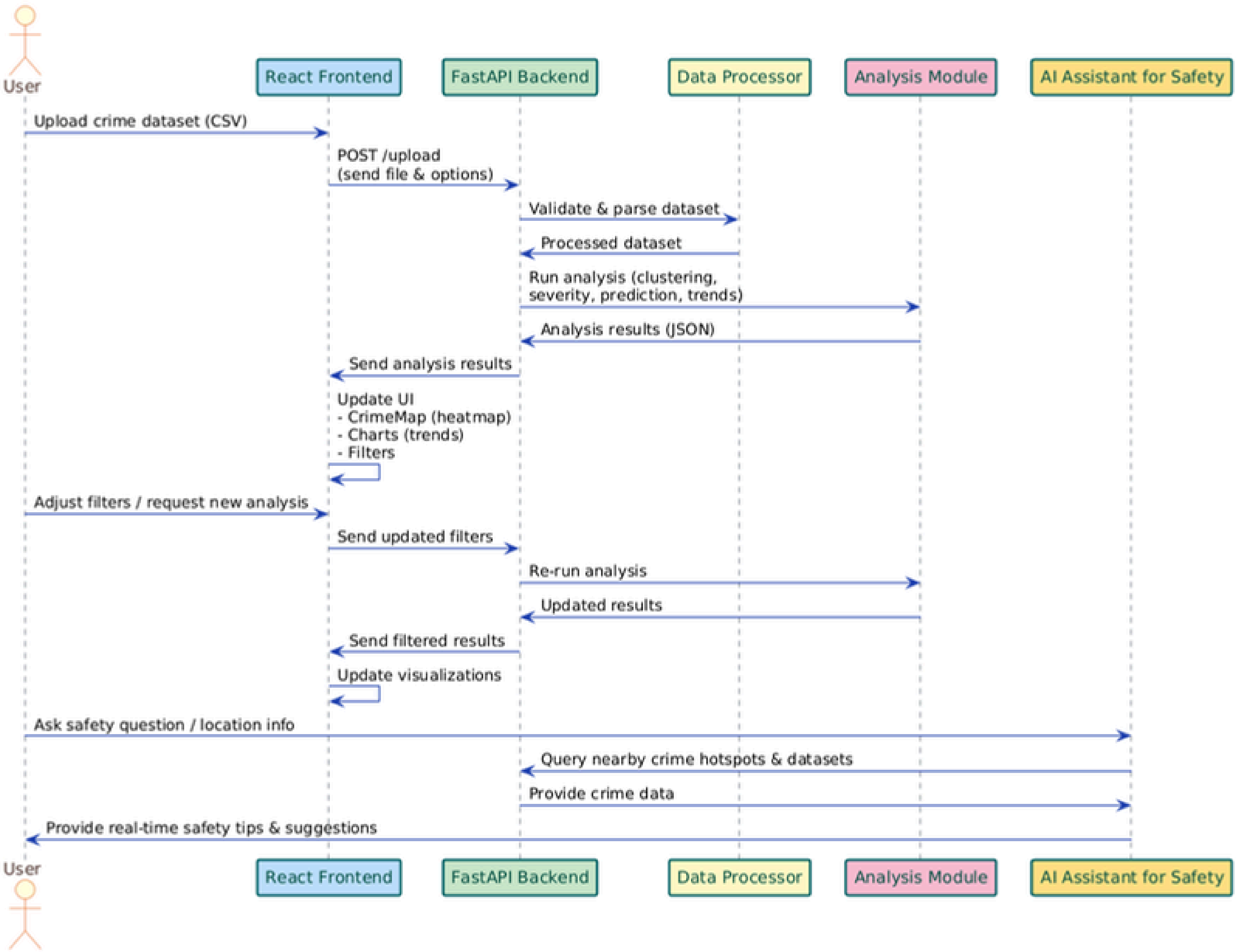
ARCHITECTURE DIAGRAM



ACTIVITY DIAGRAM



SEQUENCE DIAGRAM



Current Progress

Component	Achievement	Description
Data Pipeline & ML Ready	Functional Three-Tier Architecture	The full FastAPI (Python) backend is connected to the React frontend. Data ingestion, filtering, and in-memory storage are operational.
Analytical Engine Online	Hotspots & Forecasting Live	K-Means Clustering for hotspot identification is implemented. Prophet time-series forecasting is calculating 12-month trends. Results are successfully rendered on Leaflet and Chart.js.

Current Progress

Component	Achievement	Description
Core Prediction Implemented	XGBoost Classifier Deployed	The ability to train an XGBoost model on the filtered data is complete, returning accuracy and feature importance metrics to the dashboard for user review.
Gemini AI Integration	Contextual Streaming Chat Fully Functional	The Gemini API is integrated via a streaming FastAPI endpoint and a dedicated React chat component. The AI's response is dynamically grounded by the current crime context from the user's data.
Visualization Layer Complete	All Tabs Functional	The four main dashboard tabs (Hotspots, Time-Series, Severity, Prediction) are built using React, Headless UI, and Chart.js/Leaflet, displaying results from the backend in real-time based on filter changes.

LITERATURE SURVEY

SI NO	REFERENCE	FEATURES
1	A Comparative Analysis of Multiple Methods for Predicting a Specific Type of Crime in the City of Chicago - M. F. Buland Iqbal et al. (2023)	• **Issues Addressed:** The challenge of predicting specific crime types accurately using traditional models and the issue of class imbalance in crime datasets. • **Solution Prescribed:** Examines the effectiveness of ensemble models using imbalanced techniques hyperparameters, concluding that XGBoost achieved the best resultswith an F1-score of 0.86, validating its selection for your classification module.

LITERATURE SURVEY

SI NO	REFERENCE	FEATURES
2	Crime Hotspot Detection using Optimized K-means Clustering and Machine Learning Techniques - Bonam J. et al. (2023)	• **Issues Addressed:** Difficulty in identifying the optimal cluster number for K-means and low predictive accuracy from unoptimized clustering. **Solution Prescribed:** Uses the • Elbow Method to find the optimal 'K' for K-means clustering and leverages the clustered output to train a superior classifier (Random Forest, 88.08% accuracy). This provides a robust solution for the core hotspot module.

LITERATURE SURVEY

SI NO	REFERENCE	FEATURES
3	Geospatial analysis of crime patterns and hotspots for crime prevention using deep learning algorithms - Tekin et al. (2025)	• **Issues Addressed:** Traditional GIS lacking predictive capacity, and crime prediction limitations due to overfitting/imbalanced data. • **Solution Prescribed:** Introduces a six-stage framework utilizing DBSCAN for clustering and deep learning models (ConvLSTM) for enhanced spatio-temporal forecasting. It integrates Web-GIS visualization(dynamic heatmaps and charts) to bridge the gap between GIS and forecasting.

LITERATURE SURVEY

SI NO	REFERENCE	FEATURES
4	Multi-Tenant Architecture with FastAPI: Design Patterns and Pitfalls - Koushik Sathish (2025)	• **Issues Addressed:** The challenge of securely scaling a data platform for multi-user access (the project's future goal), and preventing data leakage between tenants. • **Solution Prescribed:** Uses FastAPI dependencies for Tenant Context Injection on every request to enforce isolated data access. It advocates for PostgreSQL with Row-Level Security (RLS) as a solution to prevent accidental data filtering errors, directly informing the project's future migration from in-memory Pandas to a robust database.

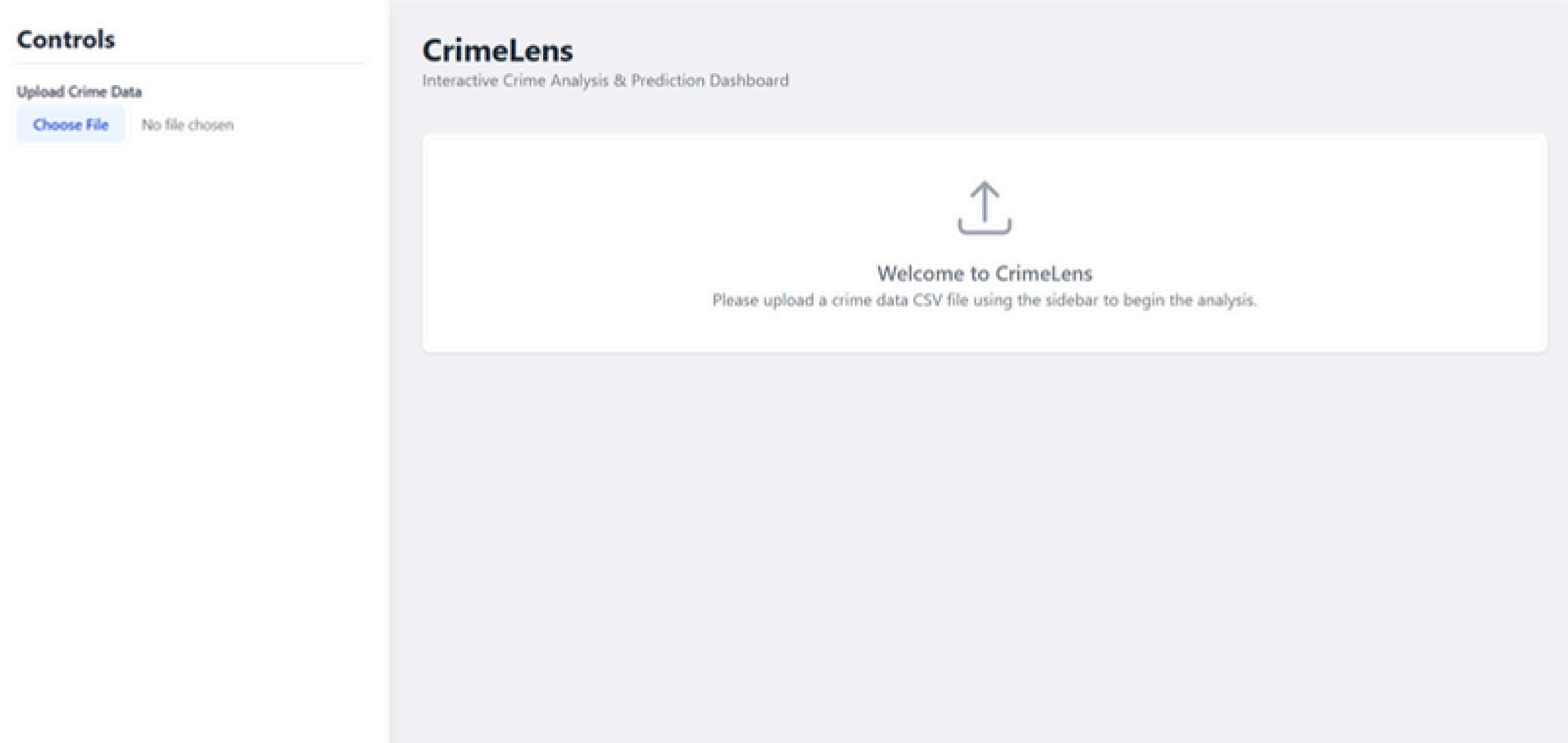
LITERATURE SURVEY

SI NO	REFERENCE	FEATURES
5	Crime Analysis using K-means Clustering - Agarwal J. et al. (2013)	• **Issues Addressed:** The difficulty of identifying crime patterns and clusters using manual or traditional statistical methods. • **Solution Prescribed:** Recommends a practical methodology for Crime Analysis using the K-means clustering algorithm. It shows that K-means can effectively identify crime patterns and clusters to help law enforcement agencies allocate resources and design effective prevention strategies.

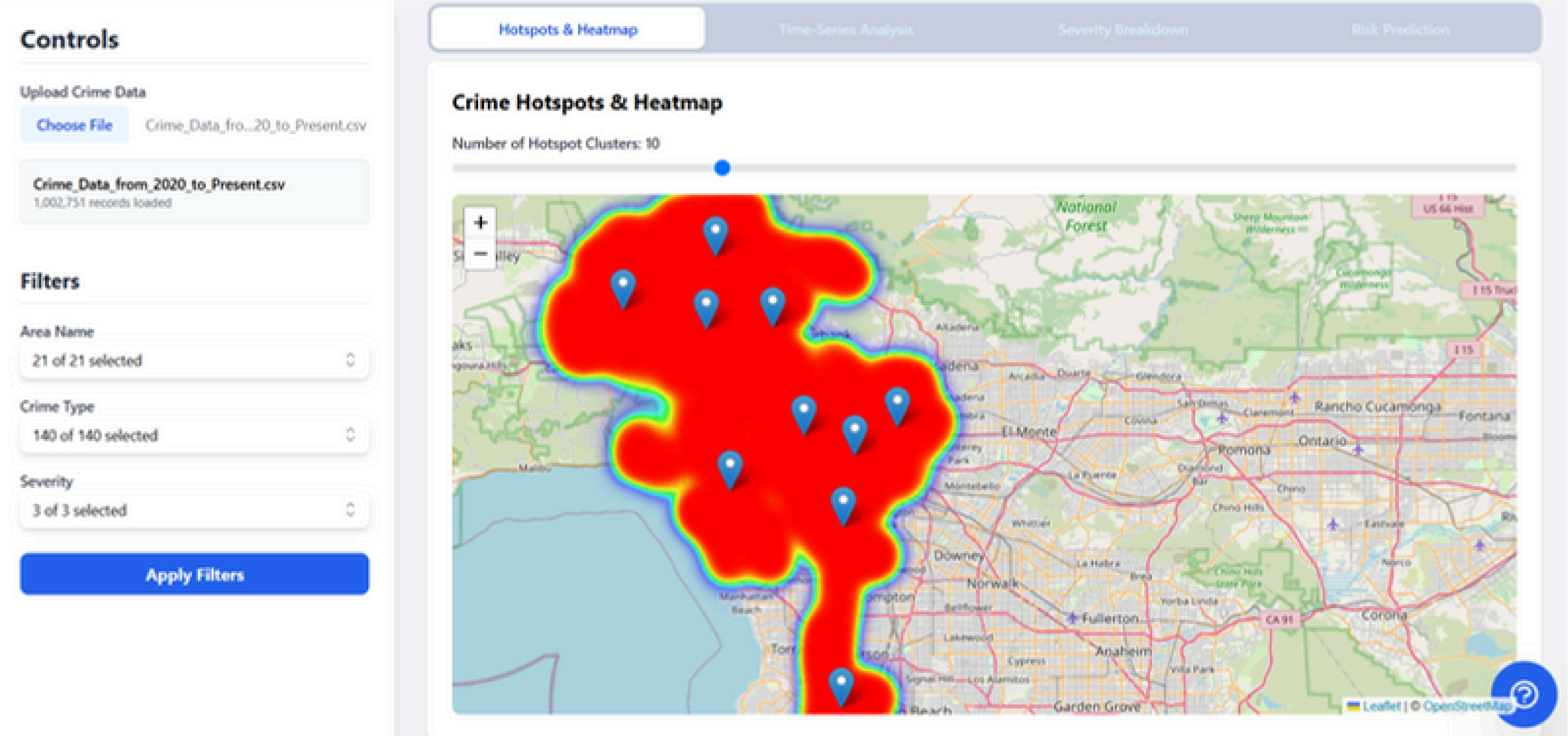
LITERATURE SURVEY

SI NO	REFERENCE	FEATURES
6	Layered Architecture & Dependency Injection: A Recipe for Clean and Testable FastAPI Code - Markoulis (2025)	• **Issues Addressed:** Lack of long-term maintainability, testability, and scalability in traditional monolithic backend designs. • **Solution Prescribed:** Structures the FastAPI application using a Layered Architecture (Presentation, Service, DAO). It uses FastAPI's Dependency Injection system to ensure components are decoupled, making the system more flexible and easier to test and maintain as it grows.

OUTPUT SCREENSHOTS

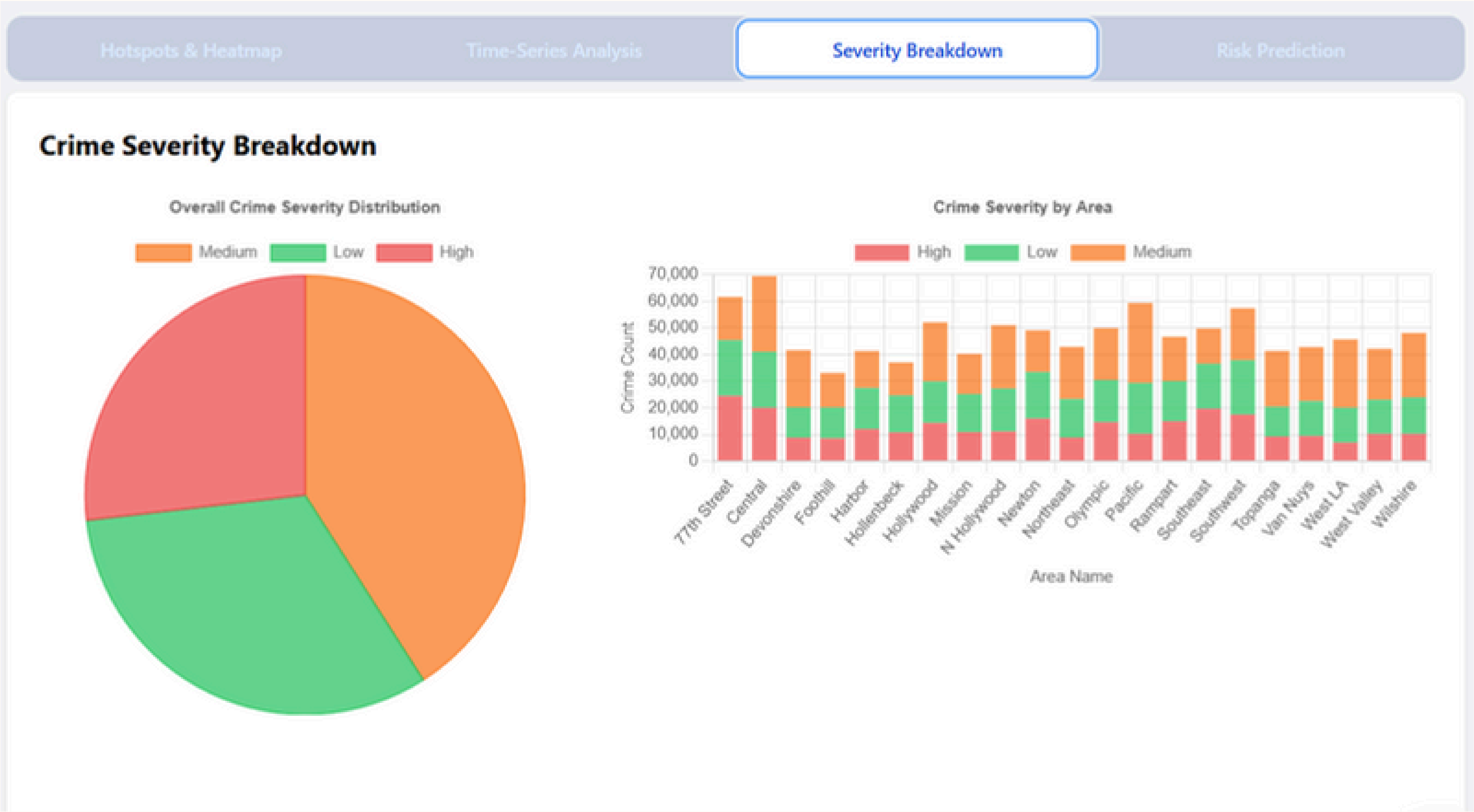


OUTPUT SCREENSHOTS

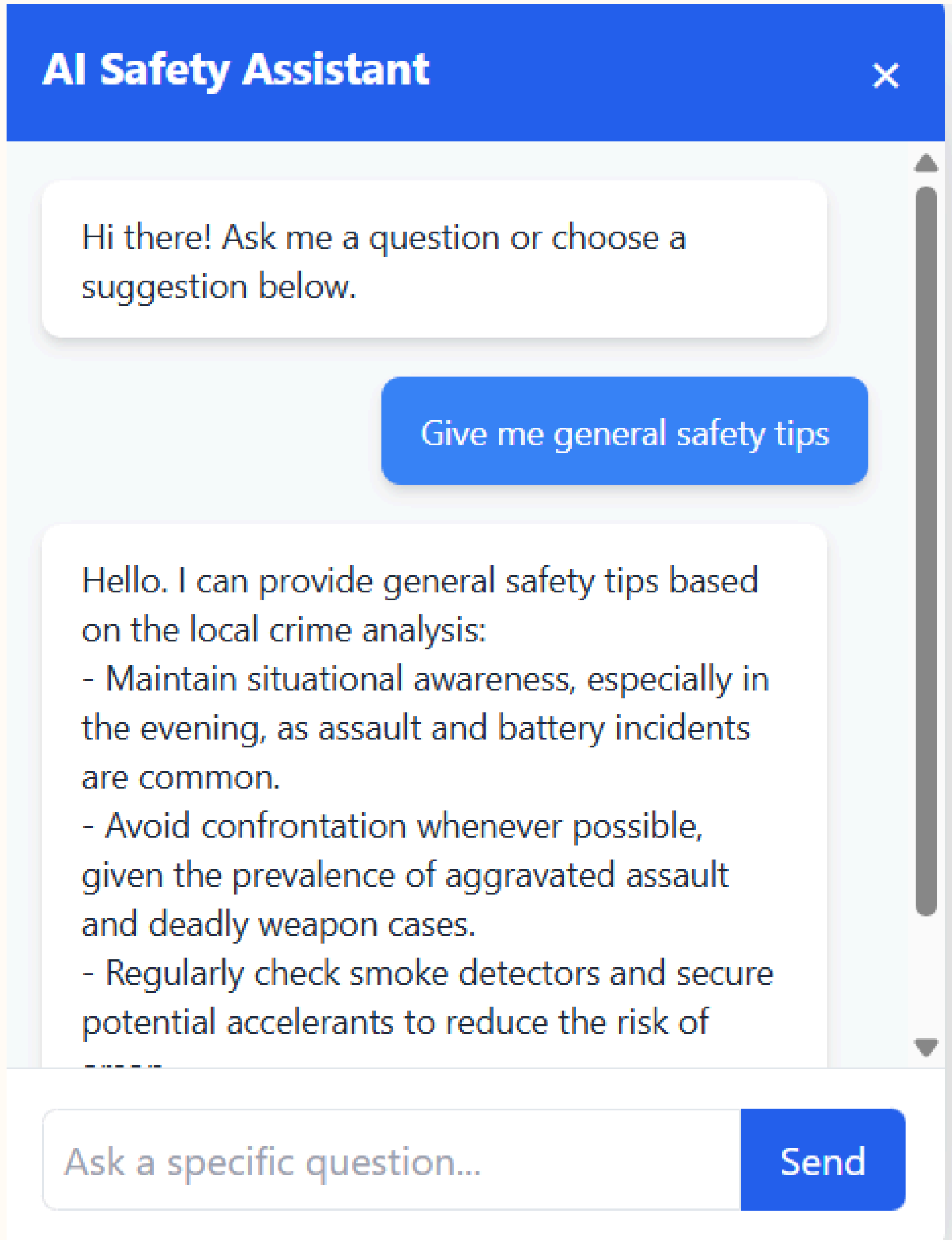


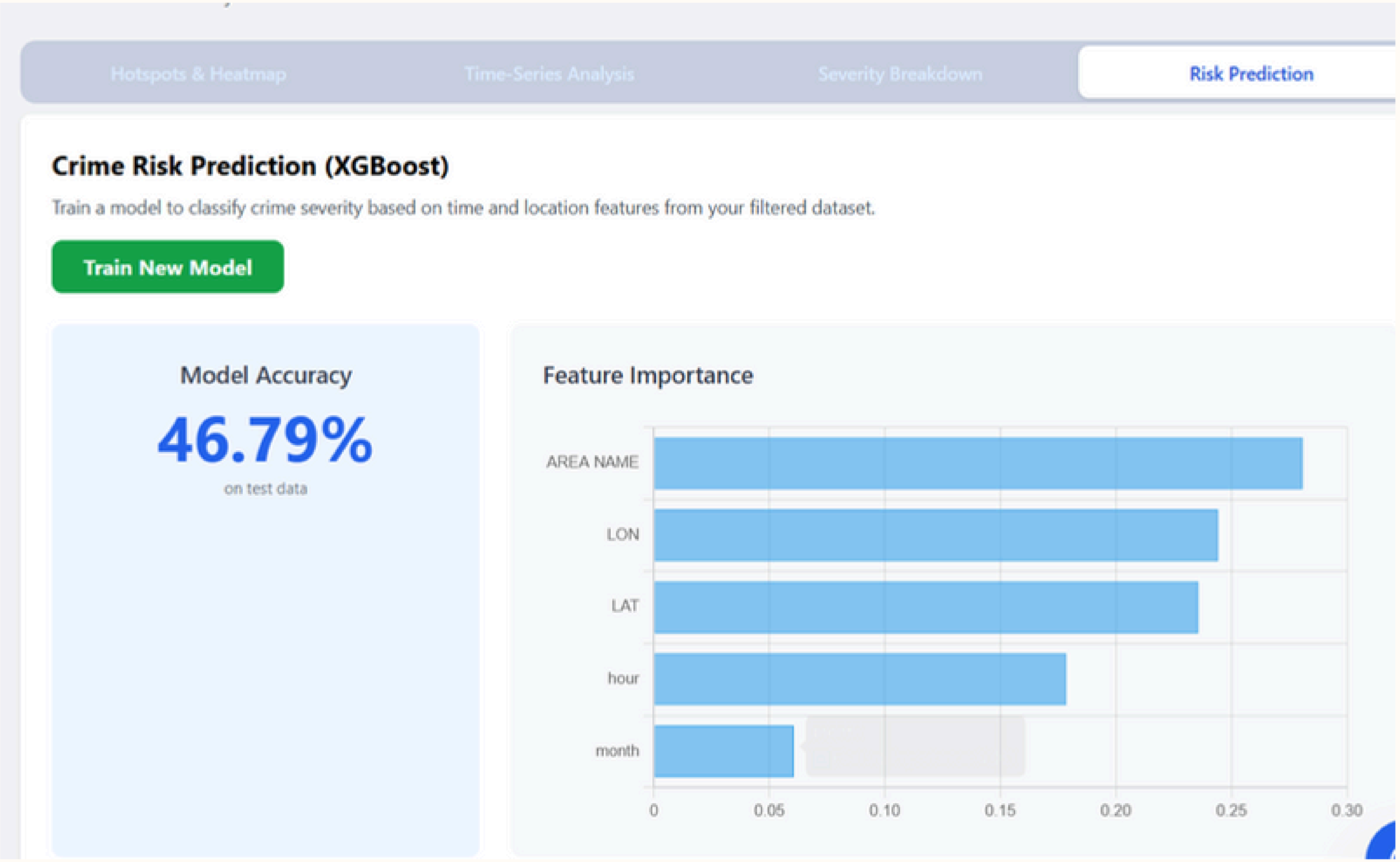
OUTPUT SCREENSHOTS

OUTPUT SCREENSHOTS

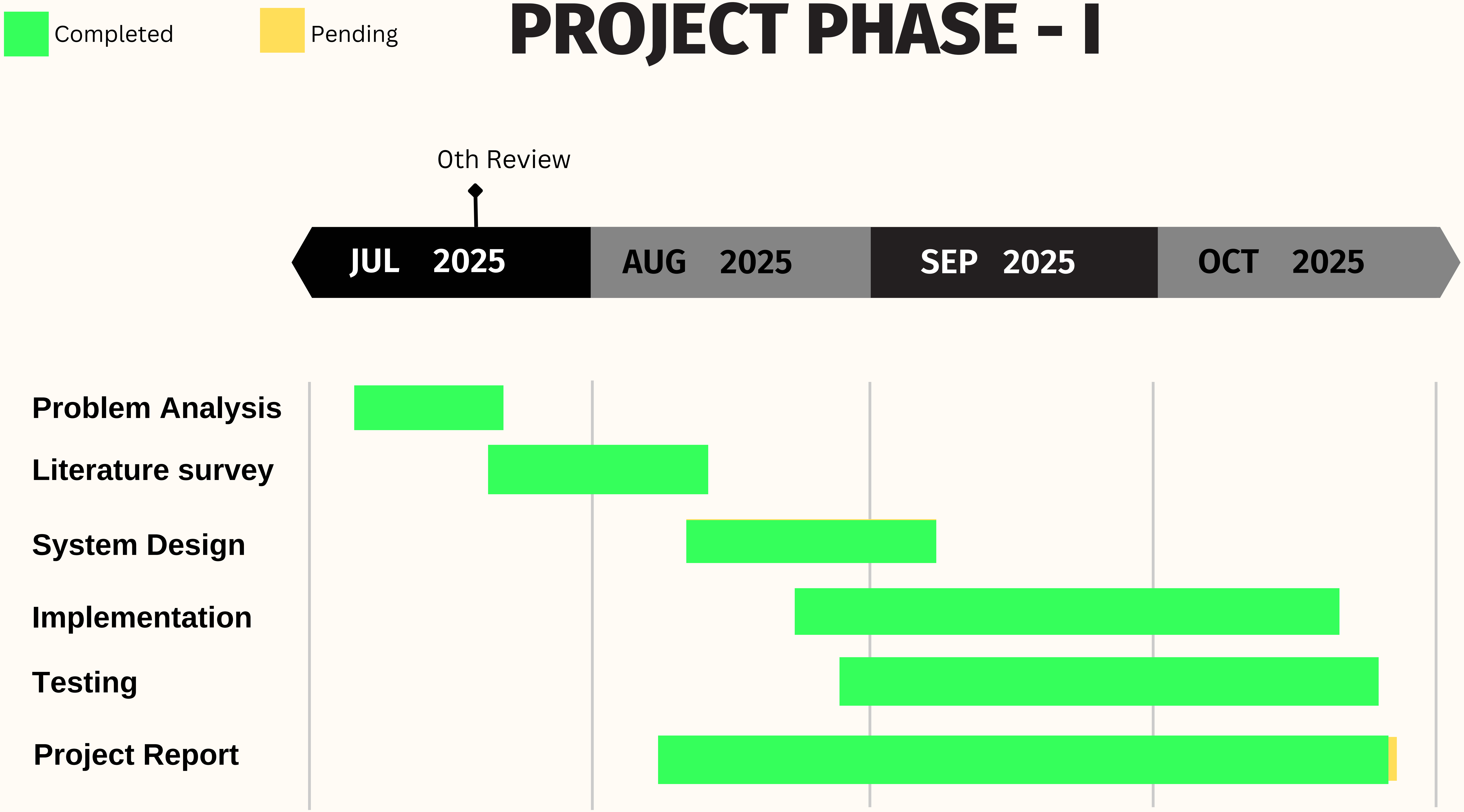


OUTPUT SCREENSHOTS





PROJECT PHASE - I



Future Scope

- Real-time crime reporting & live data integration
- Enhanced AI Safety Assistant for smarter guidance
- Instant crime alerts for emerging hotspots
- Deep learning forecasting (LSTM/BLSTM/Transformers)
- Advanced multi-density hotspot detection using HDBSCAN

Conclusion

- Successfully developed CrimeLens, an interactive crime analysis platform integrating hotspot detection, time-series forecasting, severity classification, and predictive modeling.
- Provides real-time visual insights through dashboards, heatmaps, and charts, enabling data-driven decision-making for law enforcement and policymakers.
- Incorporates an AI Safety Assistant for contextual safety guidance, enhancing public awareness and proactive measures.
- Demonstrates a scalable, modular architecture suitable for multi-region datasets and future extensions in predictive crime analytics.

REFERENCES

1. A Comparative Analysis of Multiple Methods for Predicting a Specific Type of Crime in the City of Chicago - M. F. Buland Iqbal et al. (2023)
2. Crime Hotspot Detection using Optimized K-means Clustering and Machine Learning Techniques - Bonam J. et al. (2023)
3. Geospatial analysis of crime patterns and hotspots for crime prevention using deep learning algorithms - Tekin et al. (2025)
4. Multi-Tenant Architecture with FastAPI: Design Patterns and Pitfalls - Koushik Sathish (2025)
5. Crime Analysis using K- means Clustering - Agarwal J. et al. (2013)
6. Layered Architecture & Dependency Injection: A Recipe for Clean and Testable FastAPI Code - Markoulis (2025)

THANK YOU