Requirements Specification Document

1. Functional Requirements

1.1 User Authentication & Authorization

- The system must provide **secure registration and login** for:
  - **Citizens**
  - **Administrators**
  - **Super Administrators**
- Enforce **role-based access control (RBAC)**:
  - **Citizen:** Can report and view crimes.
  - **Admin:** Can verify/reject reported crimes and view analytics.
  - **Super Admin:** Can manage admins (create, update, remove), view all audit logs, and access platform-wide settings.

1.2 Crime Reporting

- Citizens must be able to **report a crime** with the following details:
  - **Crime Type**
  - **Description**
  - **Location** (via GPS or manual input)
  - **Optional Media** (images/videos)
- All reports will initially have a **status of "Pending"**.

1.3 Crime Verification & Management

- **Admins** can:
  - View all **pending** reports.
  - **Verify** or **Reject** reports.
  - Add remarks or reasons during verification.
- **Super Admins** can monitor admin performance and activity logs.

1.4 Hotspot Detection & Visualization

- The system must:
  - Automatically **detect hotspots** — defined as ≥10 verified crimes within a **1 km radius**.
  - Display hotspots on an **interactive map (Leaflet.js)** using **heatmap visualization**.
- Filters:
  - By **crime type**
  - By **date range**
  - By **severity/frequency**

1.5 Admin Management (Super Admin Feature)

- Super Admin can:
  - Create new **Admin** accounts.
  - Update or deactivate existing **Admins**.
  - Access and filter **audit logs** for transparency and accountability.

1.6 Audit Logging

- Every critical system event must be logged, including:
  - **Report Creation** (Citizen)
  - **Verification or Rejection** (Admin)
  - **Admin Management Actions** (Super Admin)
- Log fields include:
  - Actor (User ID)
  - Action
  - Target Entity
  - Timestamp
  - Status/Remarks

---

2. Non-Functional Requirements

2.1 Performance & Scalability

- The system must support **1000+ concurrent users** without performance degradation.
- Geospatial clustering queries should complete within **3 seconds**.
- Backend and database must be **horizontally scalable** on cloud infrastructure (AWS/GCP).

2.2 Security

- Use **bcrypt** for password hashing.
- Use **JWT tokens** for session management.
- **Role-based authorization** for API protection.
- **HTTPS** enforced for all endpoints.
- Audit logs must be **immutable (append-only)**.

---

3. User Stories

3.1 Citizen

- As a **Citizen**, I want to **register/login securely**, so that I can access my account safely.
- As a **Citizen**, I want to **report a crime** with a description, type, and location.
- As a **Citizen**, I want to **upload images or videos** to strengthen my report.
- As a **Citizen**, I want to **track my report statuses** (Pending, Verified, Rejected).
- As a **Citizen**, I want to **view verified crimes and hotspots** near my area.

3.2 Admin

- As an **Admin**, I want to **view pending reports**, so that I can verify them quickly.
- As an **Admin**, I want to **approve/reject reports** with proper remarks.
- As an **Admin**, I want to **see real-time heatmaps**, so that I can identify critical areas.
- As an **Admin**, I want to **access analytics dashboards** for decision-making.

3.3 Super Admin

- As a **Super Admin**, I want to **create/manage admins** for different zones.
- As a **Super Admin**, I want to **view all audit logs** to ensure accountability.
- As a **Super Admin**, I want to **deactivate inactive or malicious admins**.
- As a **Super Admin**, I want to **oversee all system operations** from one dashboard.

---

4. Interface Requirements

4.1 User Interface (UI) Requirements

Citizen Interface

- **Report Crime Form** with fields for type, description, location, and media upload.
- **Report Tracker** showing status and timestamps.
- **Interactive Map** with cluster and heatmap layers for visualization.

Admin Dashboard

- **Table View** of reports with filters (status/date/type).
- **Action Buttons**: Verify / Reject / View Details.
- **Heatmap Visualization** for hotspots.
- **Color Legend:**
  - Pending → Yellow
  - Verified → Green
  - Rejected → Red

Super Admin Dashboard

- **Admin Management Panel**: Add / Edit / Delete Admins.
- **Audit Logs Viewer** with filters by user, date, or action.
- **Global Overview Dashboard** with system statistics.

4.2 Application Programming Interface (API) Requirements

- RESTful APIs using JSON format
- API Endpoints:
  - User Authentication (register, login, JWT)
  - Crime Reporting (submit, update, delete, fetch)
  - Crime Verification (admin-only actions)
  - Hotspot Detection (geo-clustering)
  - Analytics (crime trends, types, count)
- HTTPS enforced on all endpoints
- Role-based access control (Citizen vs Admin)

4.3 Hardware Interface Requirements

- Must support:
  - GPS modules (mobile/tablet)
  - Camera and file access for evidence upload
- Server compatibility:
  - AWS, GCP, Azure (cloud-hosted environments)

4.4 Software Interface Requirements

- Integrations:
  - MongoDB (geospatial queries)
  - Leaflet.js (map rendering)
  - Express.js (Node.js server)
  - JWT + bcrypt (authentication/security)
- Compatibility:
  - Chrome, Firefox, Edge, Safari
  - Node.js v18+

---

5. Acceptance Criteria

Authentication & Authorization

- ☐ Citizens, Admins, and Super Admins can register/login securely.
- ☐ Passwords hashed using bcrypt.
- ☐ JWT tokens generated and verified on every protected route.
- ☐ Role-based access enforced for every endpoint.
- ☐ Unauthorized access denied.

Crime Reporting

- ☐ Citizens can submit crime reports with media and location.
- ☐ Crime reports saved in MongoDB with geospatial data.
- ☐ Status updates reflected in user dashboard.

Verification Workflow

- ☐ Admins can verify/reject reports.
- ☐ Every verification/rejection triggers an **audit log**.
- ☐ Verified reports appear publicly on the map.

Hotspot Detection

- ☐ Hotspots identified for ≥10 crimes within 1 km.
- ☐ Clusters auto-refresh when new crimes are verified.
- ☐ Map interaction time < 3 seconds.

Super Admin Features

- ☐ Super Admin can manage Admin accounts.
- ☐ Audit logs record all create/update/delete actions.
- ☐ Logs cannot be edited or deleted manually.

---

6. Wireframes (Initial Draft)

Wireframes