# URBAN-PRISM: A Smart Urban Grievance Management and Infrastructure Prioritization System Using MERN Stack

---


## Abstract

Rapid urbanization has increased the complexity of managing civic infrastructure and public grievances. Traditional grievance handling systems are mostly manual or semi-digital, leading to delayed responses, inefficient prioritization, and poor utilization of available data. This project proposes URBAN-PRISM, a smart urban governance platform developed using the MERN stack (MongoDB, Express.js, React.js, Node.js). The system enables citizens to submit infrastructure-related complaints with geolocation and image evidence. These complaints are processed using spatial clustering and rule-based intelligence engines to compute risk levels and prioritize assets for maintenance. The platform also provides analytics dashboards, interactive maps, and automated notifications to improve transparency and decision-making. The proposed system aims to enhance efficiency, accountability, and data-driven planning in urban infrastructure management.


---


## Keywords

Smart Governance, Grievance Management, MERN Stack, Geospatial Analysis, Risk Assessment, Priority Ranking, Urban Infrastructure, Data Analytics


---

## 1. Introduction

Urban infrastructure management is a major challenge in developing and developed cities due to increasing population density and rising public expectations. Citizens frequently report issues related to roads, drainage systems, water pipelines, streetlights, and public facilities. However, existing grievance redressal mechanisms often lack automation, intelligent prioritization, and analytical support.

Most current systems focus only on complaint registration and tracking, without providing insights into infrastructure health or future maintenance needs. This results in inefficient resource allocation and delayed problem resolution.

To overcome these limitations, this project presents URBAN-PRISM, a web-based smart governance system that integrates geospatial data, structured datasets, and automated analysis techniques. The system transforms raw complaint data into actionable intelligence to support authorities in decision-making and long-term planning.

---

## 2. Problem Statement

Current grievance management systems suffer from several limitations such as manual prioritization, lack of data integration, absence of geospatial analysis, poor analytics support, and minimal transparency. Duplicate complaints caused by GPS inaccuracies further complicate data analysis. Authorities find it difficult to identify critical infrastructure assets and predict failures in advance.

There is a need for an integrated digital platform that can automatically process complaints, eliminate duplicates, assess risks, and provide clear priorities for infrastructure maintenance.

---

## 3. Objectives of the Project

The main objectives of URBAN-PRISM are:

- To design a digital platform for registering and managing civic grievances.

- To integrate geospatial intelligence for mapping complaints to infrastructure assets.
- To eliminate duplicate complaints using spatial clustering techniques.
- To calculate risk scores for assets using rule-based models.
- To generate priority rankings for maintenance planning.
- To provide analytical dashboards and interactive maps for administrators.
- To ensure transparency through status tracking and email notifications.
- To develop a scalable and secure system using MERN stack technologies.

---

## 4. Proposed System

The proposed system is a full-stack web application developed using the MERN stack. It consists of citizen and administrator modules supported by a centralized database and background processing engines.

Citizens can submit complaints along with location and image evidence. The system validates the input and stores it securely. A clustering mechanism groups similar complaints based on distance, ward, and category. Each cluster is mapped to a relevant infrastructure asset.

A risk engine analyzes complaint frequency, severity, recency, and maintenance history to compute a normalized risk score. A priority engine ranks assets based on these scores. Administrators access dashboards that display priorities, maps, trends, and reports. Email notifications are automatically sent when complaint statuses are updated.

---

## 5. System Architecture

The architecture of URBAN-PRISM follows a layered approach.

The frontend is developed using React.js and provides user interfaces for citizens and administrators. It handles data visualization, map interaction, and client-side validation.

The backend is developed using Node.js and Express.js. It manages authentication, API services, validation, and business logic.

MongoDB is used as the database for storing assets, grievances, priorities, and user information. Geospatial indexing is applied for efficient location-based queries.

Background jobs handle clustering, risk calculation, priority generation, and analytics processing.

The overall workflow is:

Frontend → REST API → MongoDB → Intelligence Engine → Dashboard

---

# 6. Technology Stack

The system is developed using the MERN stack and supporting tools.

## Backend Technologies

- Node.js (Runtime Environment)
- Express.js (Web Framework)
- MongoDB (Database)
- Mongoose (ODM)
- Multer (File Upload)

- Winston (Logging)
- dotenv (Environment Management)

### Frontend Technologies

- React.js
- Vite
- Tailwind CSS
- Axios
- React Query
- Zustand
- Recharts
- Leaflet
- Framer Motion
- React Hot Toast

### Other Tools

- JWT (Authentication)
- SMTP (Email Notification)

- Git (Version Control)

---

## 7. Dataset Description

The system uses synthetic datasets to simulate real-world urban environments.

The assets dataset contains information about infrastructure elements including asset ID, type, geographical coordinates, maintenance history, repair cost, and service radius.

The grievances dataset contains citizen complaints including grievance ID, description, category, location, severity, and date.

These datasets are validated, cleaned, and imported into MongoDB. They serve as operational data for analysis and intelligence processing.

---

## 8. Methodology

The methodology of the system consists of multiple stages.

Initially, datasets are imported and validated using ETL scripts. Invalid records are rejected and duplicates are removed.

When a complaint is submitted, spatial clustering is applied to group nearby complaints with similar context. This eliminates duplicate entries caused by GPS variations.

Each cluster is mapped to the nearest relevant asset using geospatial queries.

Risk scores are calculated using weighted formulas considering complaint count, severity, recency, maintenance age, and repair cost.

Assets are ranked using stable sorting algorithms to generate priority lists.

Analytical modules aggregate data to generate charts, heatmaps, and reports.

Email notifications are triggered on status updates through background services.

---

# 9. System Features

The major features of the system include:

- Online grievance registration
- Map-based location selection
- Image upload support
- Automatic complaint clustering
- Asset mapping
- Risk score computation
- Priority ranking
- Interactive dashboards
- Heatmap visualization
- Filtering and sorting
- Email notifications
- Secure authentication
- Role-based access control

- Report generation

---

## 10. Security and Reliability

The system implements JWT-based authentication and role-based authorization. Input validation and sanitization protect against injection attacks. Rate limiting prevents abuse.

Centralized logging and global error handling ensure fault detection. Crash recovery and health monitoring enhance system reliability.

Sensitive data is protected using environment variables and secure storage practices.

---

## 11. Expected Results

The proposed system is expected to improve grievance resolution time, enhance transparency, support proactive maintenance, and provide data-driven insights. Administrators can easily identify high-risk assets and allocate resources efficiently. Citizens can track their complaints and receive timely updates.

---

## 12. Future Enhancements

In future versions, machine learning techniques can be integrated for predictive maintenance and risk forecasting. Natural language processing can be used for automatic complaint classification. Real-time updates using WebSockets and IoT-based sensor integration can further enhance system intelligence.

---

# 13. Conclusion

URBAN-PRISM is a comprehensive smart governance platform that integrates grievance management, geospatial intelligence, and automated prioritization using MERN stack technologies. By converting raw complaint data into actionable insights, the system improves urban infrastructure management and supports sustainable development. The project demonstrates practical application of full-stack development, data analytics, and intelligent system design.