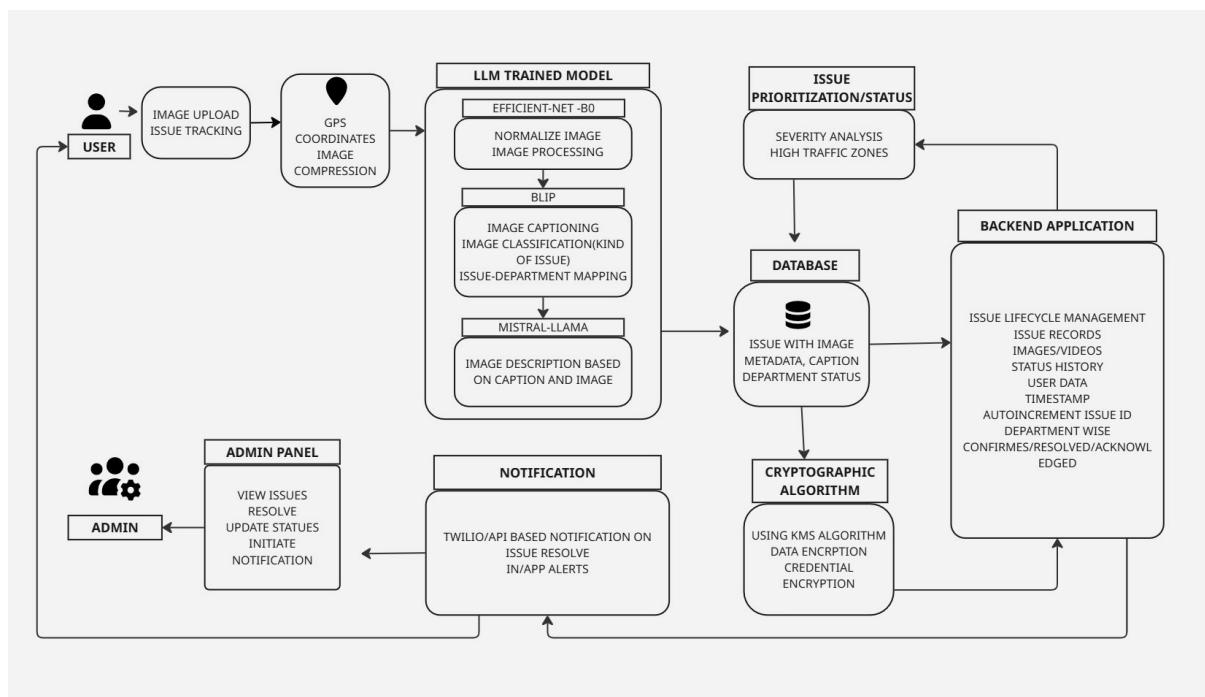


# SOFTWARE ENGINEERING BCSE301P

NAME	SONALI
REG NO	23BCE1572
LAB NO	1

## ARCHITECTURE DIAGRAM

### CROWD-SOURCED CIVIC ISSUE REPORTING AND RESOLUTION SYSTEM



## FUNCTIONAL REQUIREMENTS

### 1. User Management

- The system shall allow citizens to submit civic issues via mobile/web application.
- The system shall store user details securely.
- The system shall associate each issue with a unique user ID.

### 2. Issue Submission

- The system shall allow users to upload an image or video of the civic issue.
- The system shall automatically extract GPS coordinates from the device.
- The system shall compress and normalize uploaded media before processing.

### **3. AI Processing**

- The system shall classify the type of civic issue using a trained image classification model.
- The system shall generate an image caption describing the issue.
- The system shall generate a detailed issue description using an LLM.
- The system shall automatically map the issue to the appropriate department.

### **4. Issue Prioritization**

- The system shall analyze issue severity based on visual cues.
- The system shall prioritize issues occurring in high-traffic or critical zones.
- The system shall update issue priority dynamically if conditions change.

### **5. Issue Tracking**

- The system shall allow users to track the status of their submitted issues.
- The system shall maintain a status history (Confirmed → Acknowledged → Resolved).
- The system shall timestamp all status updates.

### **6. Admin Operations**

- The system shall allow administrators to view all reported issues.
- The system shall allow administrators to update issue status.
- The system shall allow administrators to initiate notifications to users.
- The system shall log all admin actions for accountability.

### **7. Notification System**

- The system shall send notifications when issue status changes.
- The system shall support API-based notifications (SMS / In-app).
- The system shall notify both citizens and administrators.

### **8. Data Security**

- The system shall encrypt stored data using cryptographic algorithms.
- The system shall encrypt user credentials and sensitive information.
- The system shall ensure secure data transmission between modules.

## **NON-FUNCTIONAL REQUIREMENTS**

### **1. Performance**

- The system shall process image uploads with minimal latency.

- The system shall generate AI predictions within acceptable response time.
- The system shall support concurrent submissions from multiple users.

## 2. Scalability

- The system shall scale horizontally to handle increased user traffic.
- The system shall support addition of new issue categories without redesign.
- The system shall support multiple city or region deployments.

## 3. Reliability

- The system shall ensure high availability of services.
- The system shall prevent data loss during system failures.
- The system shall maintain consistency of issue records.

## 4. Security

- The system shall follow secure authentication and authorization mechanisms.
- The system shall protect against unauthorized access.
- The system shall comply with data privacy regulations.

## 5. Usability

- The system shall provide a simple and intuitive user interface.
- The system shall minimize user effort during issue reporting.
- The system shall be accessible on both mobile and web platforms.

## 6. Maintainability

- The system shall be modular and easy to maintain.
- The system shall allow updates to AI models without system downtime.
- The system shall support logging and monitoring for debugging.

## 7. Compatibility

- The system shall support major mobile operating systems and browsers.
- The system shall be compatible with different device cameras and resolutions.