

# Comprehensive Master Specification: Citizen Road Infrastructure Reporting System

## 1. Project Overview & Strategic Objective

The Citizen Road Infrastructure Reporting System is an end-to-end civic technology platform designed to transform road maintenance into a transparent, accountable, and data-driven process. By connecting citizens, municipal authorities, and field maintenance teams, the system ensures that infrastructure failures (potholes, drainage issues, etc.) are captured with high precision and resolved with verifiable proof.

---

## 2. The "Rules of the Road": System-Wide Logic

These foundational rules govern the interaction between all five project modules.

### 2.1 The 5-Player Ecosystem

- **Citizen:** Primary reporter of issues; provider of "Before" evidence.
- **Govt Admin (Zone Manager):** Operational controller; manages triage, assignment, and quality review for a specific ward/zone.
- **Field Worker:** On-the-ground execution staff; provider of "After" resolution evidence.
- **System Admin:** Platform owner; manages the onboarding of authorities and global configuration.
- **Analytics Viewer (NGOs/Public):** External monitors; track performance and city health via read-only metrics.

### 2.2 The Unified Issue Lifecycle

Every issue must transition through these states. Skipping steps is forbidden.

1. **REPORTED (Start):** Citizen uploads photo + location. System performs a silent 5m radius duplicate check.
2. **ASSIGNED:** Govt Admin selects a verified worker for the task.
3. **IN\_PROGRESS:** Worker clicks "Start Work" upon arrival. Ticket is locked to that worker.
4. **RESOLVED:** Worker uploads mandatory "After" photo proof.
5. **CLOSED (End):** Terminal state after Admin approval. Data is archived for analytics.
6. **DISMISSED:** Admin flags the report as spam, duplicate, or out-of-jurisdiction.

### 2.3 Shared Data Properties (The Data Dictionary)

- **Issue ID:** Unique UUID tracking number.
  - **Location:** GPS Lat/Long, Zone ID (GeoJSON polygon), and human-readable Address.
  - **Status:** {REPORTED, ASSIGNED, IN\_PROGRESS, RESOLVED, CLOSED, DISMISSED}.
  - **Evidence:** photo\_report (Before) and photo\_resolve (After).
  - **Scoring Logic:**
    - **report\_count:** Number of citizens reporting the same issue (duplicate aggregation).
    - **Admin Severity:** P1 (Critical) to P4 (Low).
    - **Priority Score:** Determined by admin priority and report\_count.
- 

## 3. Module 1: Platform Administration (System Admin)

**Objective:** Centralized control layer for authority onboarding and system-wide governance.

### 3.1 Functional Requirements

- **Authority Management:** Add/manage government departments and municipal bodies. Assign jurisdictional scopes (City, Zone, Ward).
- **Role-Based Access Control (RBAC):** Create and manage roles for Govt Admins, Field Workers, and NGO Viewers.
- **Issue Configuration:** Manage the global category list (Add/Edit/Deactivate Potholes, Streetlights, etc.). Configure category metadata (Priority, expected resolution time).
- **Workflow Engine:** Define valid ticket states and toggle mandatory Admin approval before resolution.
- **System Monitoring:** View high-level stats (Total reports, active users, storage usage). Monitor duplicate detection performance.
- **Data Integrity:** Flag abnormal reporting patterns or system misuse.
- **Manual Creation:** Admin-only interface for adding internal inspection issues.
- **Audit Trail:** Maintain logs of all platform-level changes (Role updates, category changes, onboarding events).

### 3.2 Technical Requirements

- **Configuration Store:** Centralized service for permissions and workflows; changes propagate without downtime.
- **Storage Management:** Admin-only access to MinIO metadata (PII redacted). Monitor storage growth trends.

### 3.3 User Flow: Administration Lifecycle

1. **Authority Onboarding:** Admin adds authority -> defines jurisdiction boundary -> authority becomes available.
  2. **System Configuration:** Admin updates categories and resolution rules -> changes apply to future reports.
  3. **User Setup:** Admin creates/approves accounts -> users gain access to dashboards.
  4. **Monitoring:** Admin tracks system usage and platform health.
- 

## 4. Module 2: Citizen Issue Reporting (Mobile App)

**Objective:** High-fidelity data ingestion with duplicate prevention and offline resilience.

### 4.1 Functional Requirements

- **Silent Duplicate Check:** System checks within 5m and merges into an existing issue without showing nearby markers.
- **Duplicate Aggregation:** If a match is found, append evidence and increment report\_count.
- **Location Services:** GPS-locked reporting with Accuracy Indicators ( $\pm 5m$ ). Fine location (GNSS) required; cached data older than X seconds ignored.
- **Evidence Collection:** Mandatory photo capture (Camera or Gallery). App strips EXIF metadata for privacy but enforces device GPS location.
- **Category Fetching:** Dynamic list retrieval ( `GET /categories` ) with local caching and "General" fallback.
- **Offline Support:** Reports queued in local DB (SQLite/AsyncStorage) when signal is lost.
- **Sync Manager:** Connectivity listener triggers background retries; clearing local copies upon success.
- **Citizen Dashboard:** Status tracking via timeline/stepper; side-by-side view of Before/After photos for resolved tasks.
- **Resolution Feedback:** Citizens can like/dislike RESOLVED/CLOSED issues only (no upvoting on report creation).

## 4.2 User Flows

- **Report New Issue:** Fab/Report button -> Map loads focused on GPS -> No match found -> Select Category -> Capture Photo -> Submit -> Success redirect to Dashboard.
- **Duplicate Merge:** Report submitted -> Backend detects nearby issue -> Evidence appended -> report\_count increments.
- **Offline Sync:** Submit report -> Network fails -> "Saved to Outbox" toast -> User moves to WiFi -> System detects sync event -> Status updates to "Reported".

## 4.3 Edge Cases

- **GPS Drift:** If drift >5m, users may miss existing tickets. Issues plotted slightly outside polygons are flagged visually.
  - **Photo Mismatch:** Gallery photos from different locations are overridden by device GPS source of truth.
  - **Permission Denial:** GPS refusal blocks the reporting flow with a "Permission Required" modal.
  - **Indefinite Offline:** Reports remain pending until connectivity returns. Storage caps (FIFO not permitted) prevent overfilling.
  - **Hidden duplicate:** User unknowingly reports an existing issue; backend merges it into the same ticket.
- 

# 5. Module 3: Authority Operations (Govt Dashboard)

**Objective:** Command center for triage, assignment, and quality control.

## 5.1 Functional Requirements

- **Geospatial Operations Map:** Zone boundary overlay (GeoJSON) with color-coded pins (Red: Reported, Yellow: Assigned/In-Progress, Green: Resolved).
- **Map Interaction:** Click pin for "Quick View" card (Photo, ID, Status) with "Open Ticket" modal trigger.
- **Kanban Dashboard:** Status columns (Reported, Assigned, In-Progress, Resolved). Sort by Date (Oldest First) or report\_count (Urgency).
- **Bulk Assignment:** Select multiple "Reported" cards -> Assign to Worker -> Modal shows Worker Load (active task count) -> Transnational (Atomic) backend update.
- **Team Management:** Invite workers by email; auto-activate on first login. Invites expire after a configured window (default 7 days), can be re-sent, and can enforce allowed domains. Remove/Deactivate active staff; triggers immediate session revocation and task reset (Assigned -> Reported).
- **Resolution Review:** Mandatory photo comparison. "Reject Fix" requires reason text (Pushed to Citizen). "Close Ticket" finalizes analytics.
- **Spam Dismissal:** Standard reason codes (Duplicate, Fake, Out-of-Zone) + custom notes.
- **Profile:** Edit name, phone (contact only), and password (email users only).

## 5.2 Technical Requirements

- **Performance:** Map interaction maintains {TARGET\_FPS}; Zone rendering < {MAX\_MS}.
- **Scalability:** Kanban DOM virtualization for large backlogs.
- **Integrity:** Bulk assignments are atomic; rollback on subset failure.

## 5.3 User Flows

- **Bulk Triage:** Dashboard load -> Select cards in Reported -> Click Bulk Assign -> Select worker from dropdown (seeing load) -> Confirm -> Cards move to Assigned.

- **Resolution Review:** Click "Resolved" card -> Inspect Photos side-by-side -> Click Reject -> Enter "Debris left" -> Submit -> Ticket returns to In-Progress.
- **Team Setup:** Notification on "Team" tab -> Review pending request -> Click Approve -> Worker added to dropdown.

## 5.4 Edge Cases

- **Concurrent Updates:** card updated by another Admin during bulk action; requires optimistic UI and error handling.
  - **No Zone Configured:** API 404 for Zone GeoJSON shows error overlay.
  - **Broken Evidence:** Corruption in resolution photos forces "Reject Fix" with "Corrupted Photo" reason.
  - **Worker Deactivation:** If worker removed mid-upload, action fails with "Unauthorized".
- 

# 6. Module 4: Field Force Mobility (Worker App)

**Objective:** Digital task queue and verifiable resolution submission.

## 6.1 Functional Requirements

- **Onboarding:** Invite-only registration via pre-approved email. Verified status required for task access (HTTP 403 fallback).
- **Task Queue:** List/Map views with distance sorting. Card displays Issue ID, Category Icon, Photo Thumbnail, and "Days Open".
- **Navigation:** "Get Directions" triggers external map apps to the specific GPS pin.
- **Status Lifecycle:** Manual "Start Work" button locks ticket and updates status/timestamp.
- **Evidence Submission:** Resolution requires in-app camera capture (Gallery forbidden). Resolution notes optional.
- **Performance History:** Profile counts of total resolved tasks; historical list of Resolved/Closed items.

## 6.2 User Flow: Task Execution

1. **Acceptance:** Worker opens "My Tasks" -> Selects card -> Navigates to site.
2. **Start:** Worker arrives -> Clicks "Start Work" -> Status Badge updates to In-Progress.
3. **Resolve:** Repairs finished -> Taps "Resolve" -> Capture photo -> Submit -> Ticket moves to Resolved; Citizen notified.

## 6.3 Edge Cases

- **Misleading Coordinates:** Citizen reported GPS may be inaccurate; worker relies on "Before" photo for visual search.
  - **Data Limitations:** High-res photo loading may fail on limited data plans.
  - **Forgetfulness:** Worker forgets to click "Start Work", impacting performance metrics.
  - **Accidental Tap:** "Start Work" clicked by mistake; requires resolution or Admin help (no undo).
  - **Upload Failure:** Resolve fails mid-upload; state remains In-Progress with "Retry" option.
- 

# 7. Module 5: Aggregated Public Analytics (Visualization Dashboard)

**Objective:** End-to-end transparency and performance visualization.

## 7.1 Functional Requirements

- **Issue Pipeline:** Funnel showing live volume transition (Reported -> In-Progress -> Resolved).
- **Transparency Logs:** Public access to side-by-side Before/After photos for all closed tickets.

- **Advanced Scoping:** Filters for Issue Type, Geography (Ward/Zone/GPS Radius), and Temporal Range (Historical/Seasonal).
- **NGO Features:**
  - **Heatmap of Neglect:** Geospatial visualization of unresolved vs resolved concentrations.
  - **Average Resolution Time (ART):** Neighborhood-level performance metrics.
  - **Data Export:** CSV/JSON downloads for advocacy research.

7.2 Technical Requirements

- **Aggregation Engine:** Background jobs calculate ART and volume trends for instant loading.
- **Separation:** Read-only architecture physically or logically separate from the Write DB.
- **Privacy:** Strict data masking (PII redaction) for citizens and worker names.
- **Accessibility:** Multilingual support (Bhashini integration).

7.3 User Flow: Citizen Reporting & Aggregation

1. **Ingestion:** Citizen identifies issue -> uploads photo/GPS -> ticket created.
2. **Operations:** Admin reviews -> system runs de-duplication -> Admin assigns to worker.
3. **Resolution:** Worker arrives -> updates status -> submits resolution photo.
4. **Analytics:** Background engine pulls resolved data -> Heatmap and Funnel update instantly.

8. Integrated Failure Recovery & Edge Case Master List

Category	Issue	System Response / Mitigation
GPS	Accuracy/Drift	Accuracy indicators (±15m); ignore cached data; flag out-of-bounds pins.
Connectivity	Signal Loss	Store-and-forward (Local DB); Background retry; "Cloud with Slash" status icon.
Operations	Concurrent Edit	Optimistic UI; Server-side status validation before state change.
Evidence	Poor Quality	Admin "Reject Fix" loop; side-by-side photo audit; mandatory reason input.
Personnel	Invite Delays	Dashboard indicators for outstanding invites.
Security	Access Revocation	Immediate session token invalidation; automatic task reset to unassigned.
Data	Duplicates	Silent 5m dedup; append evidence and increment report_count.
Infrastructure	Missing Config	Error overlay if Zone GeoJSON or Categories are not defined by System Admin.