Functional Requirements

Functional Requirements for Citizen

1. User Authentication & Account Management

- Citizens can easily create an account by providing essential details like name, email, phone number.
- Citizens can log in securely using their Username and password.
- Citizens can reset their password via a registered Email id and OTP.

2. Complaint Submission

- Submit complaints with a photo, description, and category.
- Option to submit complaints anonymously.
- Automatic generation of a unique complaint ID for each submission.
- Automatic Department suggestion by Machine Learning model using Photos Uploaded.

3. Complaint Tracking

- View the status of complaints through stages: Received → In Progress → Resolved.
- Receive notifications/alerts when the complaint status changes.
- Detailed View of any complaint (Address, Current status, to whom complaint is assigned) uploaded by Citizen can be seen in Homepage.

4. Community Engagement

- Upvote existing complaints to indicate priority or urgency.
- Users can report the existing complaint if the user finds it fake.

5. Search and Filtering

- Search complaints by keywords, category, or location.
- Filter & Sort complaints based on Upvotes, department, or Upload date.

6. Feedback & acceptance of resolution

- User can accept or reject resolutions submitted by Fieldworker.
- User can see all the complaints submitted by him/her.

Functional Requirements for Government Authorities

1. **User Authentication & Account Management**
    - Sign-Up (Admin-Approval Based): Government officials can request an account by providing details including name, official email, and crucially their department.
    - Officials can log in securely using their credentials.

2. **Dashboard and Complaint Management**
    - View all complaints on a centralized dashboard.
    - Complaints are automatically categorized and prioritized using AI/ML algorithms.
    - Assign complaints to relevant field workers for resolution.

3. **Search & Filters**
    - Search complaints by ID, category, location.
    - Filter complaints by Upvotes, department, or upload date.

4. **Additional**
    - Track Field Worker leaderboard.

Functional Requirements for Field Workers

1. **User Authentication & Account Management**
    - Sign-Up (Admin-Approval Based): Field workers can be registered by an administrator or request an account with details including name, and assigned department.
    - Field workers can log in securely using their credentials.

2. **Work Reporting**
    - Upload photos of completed work for verification and record keeping.
    - Add remarks or notes regarding work done, challenges faced, or special observations.

3. **Task Management**
    - View assigned complaints/tasks with details such as location, priority, and deadline.
    - Receive alerts or notifications for newly assigned or escalated tasks.

4. **Additional**
    - Access history of completed tasks.
    - Leaderboard for fieldworkers who have solved maximum complaints.

NonFunctional Requirement

1. **Usability**
    - Intuitive and easy-to-use interface suitable for users of all ages and backgrounds.

2. **Performance**
    - AI-powered complaint categorization and prioritization should complete within 10 seconds.
    - The system should maintain low response times for all user interactions, even under high load.

3. **Scalability**
    - Capable of handling thousands of concurrent complaints and user sessions.
    - Flexible architecture to accommodate future growth in users, complaints, and modules.

4. **Security & Privacy**
    - Data encryption for storage and transmission of sensitive information.
    - Implement federated learning or other privacy-preserving techniques for AI training on sensitive data.
    - Role-based access control to prevent unauthorized access.

User Cases

UC1: User Registration & Login

Actor: Citizen / Government Official / Field Worker

Goal: Create an account and log in securely.

Preconditions:

- User must provide required details.
- Officials/workers need admin approval.

Main Flow:

1. User opens Sign Up page.
2. Enters required information.
3. System validates inputs.
4. If Official/Field Worker → account goes for admin approval.
5. Post-approval user logs in with credentials.

Postconditions:

- User is authenticated and can access role-based dashboard.

Exceptions:

- Missing/invalid fields.
- Incorrect login credentials.
- Account not approved yet.

UC2: Submit Complaint

Actor: Citizen

Goal: Submit complaint with photo, category & description.

Main Flow:

1. Citizen opens "Submit Complaint".
2. Uploads photo → enters description → selects category.
3. System generates unique complaint ID.
4. Complaint stored in database.

Exceptions:

- Missing mandatory fields.
- Invalid/unsupported photo format.

UC3: Submit Anonymous Complaint

Actor: Citizen

Goal: Report issues without revealing identity.

Main Flow:

1. Citizen toggles “Submit Anonymously”.
2. Fills complaint details.
3. System hides user identity from officials.

Exceptions:

- Identity accidentally visible in dashboard.
- Anonymous complaints not tracked correctly.

UC4: Track Complaint Status

Actor: Citizen

Goal: Track progress of submitted complaint.

Main Flow:

1. Citizen opens "My Complaints".
2. System displays status: Received → In Progress → Resolved.
3. Citizen receives notifications for updates.

Exceptions:

- Status not updated on backend.
- Notification not sent.

UC5: Upvote Complaint

Actor: Citizen

Goal: Support complaints so they get higher priority.

Main Flow:

1. Citizen opens a complaint.
2. Clicks "Upvote".
3. System increments count and updates dashboard.

Exceptions:

- Multiple votes by same user.
- Upvote not reflected.

UC6: Search Complaints (Citizen)

Actor: Citizen

Goal: Find complaints using keywords/category/location.

Main Flow:

1. Citizen opens search bar.
2. Enters keyword OR selects category/location.
3. System displays matching complaints.

Exceptions:

- No results displayed even if complaint exists.
- Wrong complaints shown.

UC7: Filter Complaints (Citizen)

Actor: Citizen

Goal: Narrow down results by status, category, timeframe.

Main Flow:

1. Citizen applies selected filters.
2. System filters complaints accordingly.

Exceptions:

- Filters not applied correctly.
- Combined filters give wrong results.

UC8: Provide Feedback & Ratings

Actor: Citizen

Goal: Rate resolved complaints.

Main Flow:

1. Citizen opens resolved complaint.
2. Gives star rating & optional comment.
3. System stores feedback with complaint.

Exceptions:

- Rating allowed before resolution.
- Feedback not saved.

UC9: Official Dashboard View

Actor: Government Official

Goal: View and manage all complaints.

Main Flow:

1. Official logs in.
2. Dashboard shows list of complaints.
3. Official sorts complaints by urgency, date, etc.

Exceptions:

- Dashboard fails to load.
- Sorting not working

UC10: AI-based Complaint Prioritization

Actor: Government Official / System

Goal: Auto-categorize & prioritize complaints.

Main Flow:

1. System receives new complaint.
2. AI categorizes & assigns priority score.
3. High-priority complaints appear on top.

Exceptions:

- Wrong classification.
- Misrouted complaints.

UC11: Review Feedback (Officials)

Actor: Government Official

Goal: See citizen ratings.

Main Flow:

1. Official opens any resolved complaint.
2. System displays all ratings & comments.

Exceptions:

- Feedback missing or mismatched.
- Feedback shown for unresolved tasks.

UC12: Search Complaints (Officials)

Actor: Government Official

Goal: Search complaints by ID/category/location/status.

Main Flow:

1. Official enters search criteria.
2. System displays matching results.

Exceptions:

- Wrong or incomplete search results.

UC13: Field Worker Task Assignment

Actor: Field Worker

Goal: Receive details about assigned tasks.

Main Flow:

1. Field worker gets new task notification.
2. Task contains description, photo, location map.

Exceptions:

- Missing photo or location.
- Worker doesn’t receive assignment.

UC14: Upload Proof of Resolution

Actor: Field Worker

Goal: Provide completion proof.

Main Flow:

1. Worker uploads completion images.
2. System saves proof.
3. Status updates to “Resolved”.

Exceptions:

- Upload fails.
- Complaint closed without proof.

UC15: System Security / Role-Based Access

Actor: System Admin

Goal: Enforce role permissions.

Main Flow:

1. Admin defines roles.
2. System restricts access based on roles.
3. Logs unauthorized attempts.

Exceptions:

- Citizens accessing admin dashboard.
- Permissions applied inconsistently.

UC16: Multi-language Support

Actor: Citizen

Goal: Use app in preferred local language.

Main Flow:

1. User selects language in settings.
2. System updates UI text accordingly.

Exceptions:

- Missing translations.
- Incorrect or partial translations.

User Stories

User Story 1 – Sign Up / Login (All Users)

Front:
As a user (Citizen, Government Official, or Field Worker), I want to create an account and log in securely so that I can access the complaint management system according to my role.

Back (Acceptance Criteria):

- **Citizen:**
  - Can sign up using name, email, phone number, address/location.
  - Login using email/phone and password.
- **Government Official:**
  - Can request an account by providing name, official email, department/jurisdiction.
  - Account requires admin approval.
  - Login using approved credentials.
- **Field Worker:**
  - Registered by admin OR request account using name, employee ID, department/crew.
  - Login using approved credentials.

Failure:

- Missing fields → account creation fails.
- Incorrect credentials → login fails.
- Account not approved → login denied.

User Story 2 – Complaint Submission

Front:
As a citizen, I want to submit a complaint with a photo and description so that authorities can understand and resolve it.

Back (Acceptance Criteria):

- Complaint ID auto-generated.
- Photo and description required.
- Category selection available.

Failure:

- Missing mandatory fields.
- Unsupported photo format/size.

User Story 3 – Anonymous Complaint

Front:
As a citizen, I want the option to submit complaints anonymously so that I can report issues without fear.

Back (Acceptance Criteria):

- Toggle/checkbox for anonymity.
- Identity hidden from officials.

Failure:

- Identity accidentally revealed.
- Anonymous complaints not properly tracked.

User Story 4 – Complaint Tracking

Front:
As a citizen, I want to track my complaint status so that I know when it is being resolved.

Back (Acceptance Criteria):

- Status stages: **Received** → **In Progress** → **Resolved**.
- Real-time updates visible.

Failure:

- Status not updated.
- Notifications not received.

User Story 5 – Upvote Complaint

Front:
As a citizen, I want to upvote complaints so that high-impact issues get priority.

Back (Acceptance Criteria):

- Single upvote per user.
- Upvote count visible to officials.

Failure:

- Multiple upvotes by same user.
- Dashboard does not reflect votes.

User Story 6 – Search Complaints (Citizen)

Front:
As a citizen, I want to search complaints by keywords, category, or location so that I can quickly find relevant issues.

Back (Acceptance Criteria):

- Search by keyword.
- Search by category.

- Search by location.

Failure:

- Complaints not shown even if existing.
- Wrong complaints displayed.

User Story 7 – Field Worker Leaderboard

Front:
As a field worker, I want a leaderboard showing workers ranked by problems solved so that performance is appreciated.

Back (Acceptance Criteria):

- Leaderboard ranked by problems solved.
- Filtering by day/week/month.
- Shows worker name, ID, solved count.
- Auto-updated.

Failure:

- Leaderboard not updating.
- Incorrect ranking.
- Missing worker data.

User Story 8 – Filter Complaints (Citizen)

Front:
As a citizen, I want to filter complaints based on status, category, or timeframe so that I can narrow down results.

Back (Acceptance Criteria):

- Filter by status (Received, In Progress, Resolved).
- Filter by category & date.

Failure:

- Filters applied incorrectly.

- Combined filters wrong.

User Story 9 – Feedback & Ratings

Front:
As a citizen, I want to provide feedback or ratings for resolved complaints so that I can evaluate service quality.

Back (Acceptance Criteria):

- Rating scale (1–5).
- Optional text feedback.
- Feedback stored.

Failure:

- Ratings allowed before resolution.
- Feedback not saved.

User Story 10 – Dashboard for Officials

Front:
As a government official, I want a centralized dashboard so that I can monitor and resolve complaints efficiently.

Back (Acceptance Criteria):

- Dashboard lists all complaints.
- Sorting by urgency, date, location.

Failure:

- Dashboard fails to load many complaints.
- Sorting not working.

User Story 11 – AI Prioritization

Front:
As a government official, I want AI to auto-categorize and prioritize complaints so that urgent cases are addressed first.

Back (Acceptance Criteria):

- High-priority complaints appear first.

Failure:

- Wrong categorization.
- Misrouted complaints.

User Story 12 – Review Feedback (Officials)

Front:
As a government official, I want to review feedback on resolved complaints so that I can assess satisfaction.

Back (Acceptance Criteria):

- Officials can view ratings/comments.
- Shown in complaint details.

Failure:

- Feedback not displayed.
- Feedback shown for unresolved complaints.

User Story 13 – Search Complaints (Officials)

Front:
As a government official, I want to search complaints by ID, category, location, or status so that I can find cases quickly.

Back (Acceptance Criteria):

- Search works for all parameters.

Failure:

- Wrong or incomplete results.

User Story 14 – Field Worker Assignment

Front:
As a field worker, I want to receive assigned tasks with photos and maps so that I can resolve issues quickly.

Back (Acceptance Criteria):

- Task includes description, photo, map.
- Push notification received.

Failure:

- Missing details.
- Assignment not received.

User Story 15 – Proof of Resolution

Front:
As a field worker, I want to upload photos after work so I can provide proof of completion.

Back (Acceptance Criteria):

- At least one photo required.
- Status becomes “Resolved” after upload.

Failure:

- Upload fails.
- Complaint closed without proof.

User Story 16 – System Security

Front:
As a system admin, I want to enforce role-based access so that unauthorized users cannot misuse the system.

Back (Acceptance Criteria):

- Role permissions clearly defined.
- Unauthorized attempts logged.

Failure:

- Citizens see admin dashboard.
- Permissions inconsistent.

User Story 17 – Multi-language Support

Front:
As a citizen, I want the app in my local language so that I can use it comfortably.

Back (Acceptance Criteria):

- At least 2 languages supported.
- Easy switching.

Failure:

- Incomplete translations.