Crowd-Powered Civic Reporting App (Hackathon MVP) PRD

1) Summary
A community app where people tap a map, report issues with a photo, and upvote problems so they rise to t

2) Problem
Cities and neighborhoods struggle to track small but high-impact issues like potholes, overflowing trash, brol

3) Goal
Make reporting fast, public, and crowd-ranked.

4) Target Users
- Residents: report and upvote issues near them
- Moderators (optional): update status for demo realism

5) MVP Features (Hackathon Scope)
A) Map → Drop Pin → Report
Flow: Open map → Tap to drop pin → Choose category → Add description → Upload photo → Submit.

B) Browse Nearby Reports
Pins shown on map. Click pin to view details.

C) Voting (Upvote)
Users can upvote a report once.

D) Visibility Ranking (Simple Trending)
score = upvotes. Sort descending.

E) Status Tracking
Statuses: OPEN, IN_PROGRESS, RESOLVED.

F) AI Chatbot (Demo Version)
FAQ + tool to answer "What's trending near me?" via getTrendingReports().

G) Leaderboard
Top users by # reports created or upvotes received.

6) Non-Goals (Not in Hackathon)
No notifications, ward routing, escalation emails, duplicate detection, AI photo verification, or comments.

7) Pages
- Home Map Page
- Report Modal/Page
- Report Detail Drawer/Page
- Leaderboard Page
- Admin (Optional)

8) Data Model
users: id, name, email(optional), createdAt
reports: id, userId, category, description, lat, lng, photoUrl, status, createdAt
votes: id, reportId, userId, createdAt (unique userId+reportId)

9) API Endpoints
Reports: POST /reports, GET /reports, GET /reports/:id, PATCH /reports/:id/status
Votes: POST /reports/:id/upvote, DELETE /reports/:id/upvote
Leaderboard: GET /leaderboard
Upload: POST /upload
Chat: POST /chat

10) Hackathon Build Plan
Phase 1: Map + create report + pins
Phase 2: Upvotes + trending
Phase 3: Leaderboard
Phase 4: Chatbot

11) Demo Script
Report issue → Upvote → Trending → Leaderboard → Ask chatbot.

12) Future Hook
AI photo verification to prevent spam + estimate severity.