AURA

## AURA: Adaptive Urban Risk Analyzer for Crowd Environments
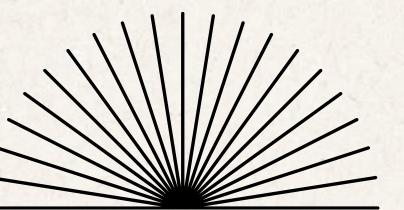
PRESENTED BY:

Divyasree D - 1BG23CS036
Harshitha N C -1BG23CS049
Thrisha R - 1BG23CS170
Vismaya M - 1BG23CS190
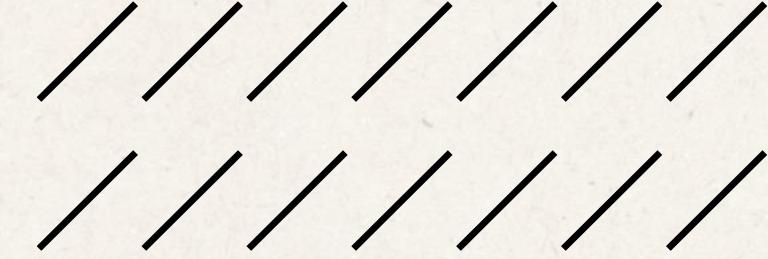
Objectives

- **Real-time Crowd Monitoring:** Continuously analyze crowd density, movement, and risk levels across all zones
- **AI-Driven Risk Detection:** Automatically identify congestion, anomalies, and high-risk conditions using ML-based metrics.
- **Intelligent Alerting System:** Generate actionable alerts and recommended evacuation routes for administrators.
- **User Safety Communication:** Deliver real-time updates and safe-route instructions to mobile users in critical situations.
- **Data-Driven Decision Support:** Provide analytics dashboards, playback mode, and historical insights for event authorities.

System Modules

1. Risk Intelligence Engine (Backend)
- Adaptive Risk Scoring (AURA-Core)
- Crowd Pressure Index (CPI)
- Surge Detection
- Exit Load Balance Score (ELBS)
- Pressure Zone Classification
- Dynamic Safe Route Generation

3. Admin Dashboard (Web)
- Multi-Zone Monitoring
- Live Heatmaps
- Risk Alerts & Broadcast System
- Time-Based Risk Trends
- Real-time analytics
- Scenario Control & Playback Mode

2. Video Processing & Simulation
- Real-time Video → Crowd Density
- Movement Vector Extraction
- Huddle & Congestion Detection
- Simulation/Test Mode
- Data Cleaning & Streaming to Backend
- Safe Route Calculation

4. User Mobile App
- Live Zone Safety Status
- Safe Route Navigation
- Mobile Alerts & Warnings
- SOS Emergency Button
- Simple UI for General Public

Crowd Analytics Engine (Backend Processing)

- Converts CCTV/mobile video into real-time crowd insights
- Detects people, tracks movement, and analyzes crowd behavior frame-by-frame

Core Processing Steps
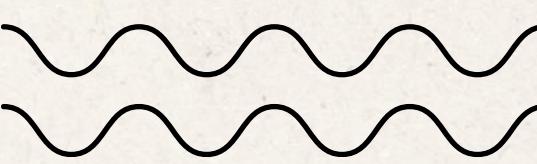
01 Video Input

02 Person Detection (YOLO)

03 Tracking & Motion Analysis
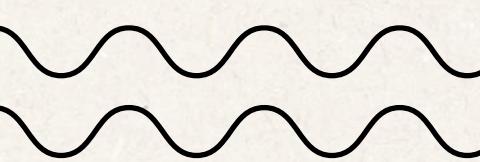
04 Zone-Based Intelligence

05 Exit Load Analysis (ELBS)

Feature Extraction & Scenario Detection

- Computes zone-wise density, crowd count, and occupancy from active tracked IDs to understand real-time crowd load.

- Measures movement speed, direction, and flow stability using centroid displacement and optical-flow based motion tracking.

- Identifies direction conflicts by comparing dominant crowd direction vs. opposite movements.

- Generates risk indicators like surge index, bottleneck index, and swirl index using patterns in speed, spacing, and motion clustering.

- Outputs all metrics as structured JSON data for seamless integration with the main backend and dashboard.

Crowd Risk Engine – Risk, Pressure & Zone Status

1. RISK SCORE (0–100)

$$\text{base\_risk} = 0.45P + 0.30F + 0.25S$$

- Converts crowd conditions into a single danger value.
- P = Pressure from density,
- F= Flow instability from speed drop + conflict,
- S = Structural risk from surge + bottleneck.
- Higher weights on density → because overcrowding is the main cause of disasters.
- Smoothed over time:

$$R_t = 0.6\,BR_t + 0.4\,R_{t-1}$$

- Spike detection when sudden danger appears.

2. CPI (Crowd Pressure Index)
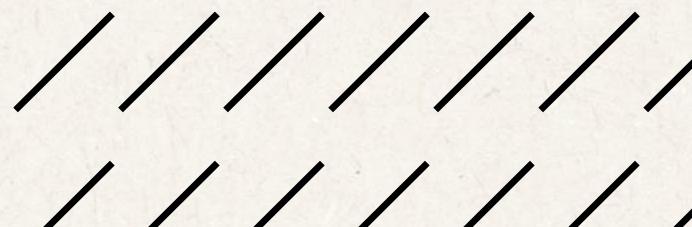
$$\text{CPI} = D \times F \times C$$

- Measures physical pressure that causes crush incidents.
- D= density, F= movement breakdown, C= compression from surge/bottleneck.
- CPI rises only when all three combine → early crush detection.
- Outputs LOW / MED / HIGH / CRITICAL pressure levels.

3. ZONE CLASSIFICATION (Colors)

$$\text{combined} = 0.6 \cdot \text{Risk} + 0.4 \cdot \text{CPI}$$

- Blends overall danger (risk) + crush pressure (CPI).
- Maps zones to GREEN, YELLOW, ORANGE, RED.
- CPI Critical → instantly RED (crush risk override).
- Risk spike → auto-upgrade to ORANGE for early warning.

Dynamic Crowd Behavior & Exit Stress Detection

4. SURGE & PANIC DETECTION

$$\text{panic\_level} = f(\text{anomaly\_count},\ \text{risk})$$

- Detects abnormal movement patterns that indicate panic.
- Checks 4 signals: forward surge, speed spike, direction chaos, risk spike.
- Classifies panic as NORMAL / MILD / ELEVATED / SEVERE.
- Requires multiple anomalies → avoids false alarms.

5. HUDDLE DETECTION

$$\text{huddle\_level} = f(\text{signals: density, speed, swirl, risk})$$
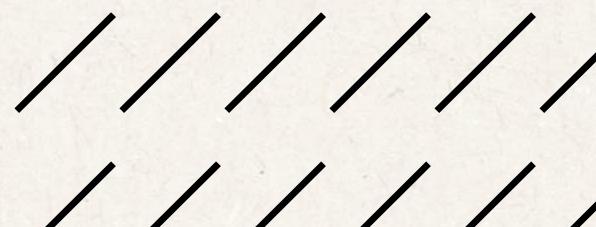
- Detects dangerous clustering / compression in the crowd.
- Uses 6 signals: high density, fast density rise, low speed, swirl motion, etc.
- Levels: NONE / MILD / ELEVATED / SEVERE.
- Identifies the early phase of crush formation before panic.

6. ELBS (Exit Load Balance Score)

Where:

$$\text{ELBS} = 0.5P + 0.3Q + 0.2B$$

- P= pressure (approaching · capacity), Q= queue build-up, B= bottleneck factor
- Measures how overloaded or free each exit is.
- Identifies NORMAL / BUSY / OVERLOADED exits.
- Also computes imbalance across all exits.
- Crucial for preventing exit crushes and bottleneck failures.

Decision Layer: Optimal Pathing & Exit Guidance

7 AURA-PULSE: Predictive Urban Load & Safety Engine

$$\text{cost} = 0.35R + 0.25CPI + 0.15H + 0.15P + 0.10Z$$

Where:
- R= Risk Score, CPI= Pressure, H= Huddle penalty, P= Panic penalty, Z= Zone color penalty
- Computes a single danger cost (0–1) per zone using risk, pressure, panic & huddle.
- Uses greedy safety-first navigation: at each step, choose safest neighboring zone.
- Avoids zones above 0.85 cost (too dangerous to pass).
- Recalculates every frame → adapts instantly to live crowd changes.

WHY NOT A* OR DIJKSTRA?
- A*/Dijkstra assume static costs → crowd danger changes every second.
- They optimize shortest path, not safest path → unsafe in real crowds.
- They explore the entire graph, but AURA-PULSE is local, real-time & adaptive.
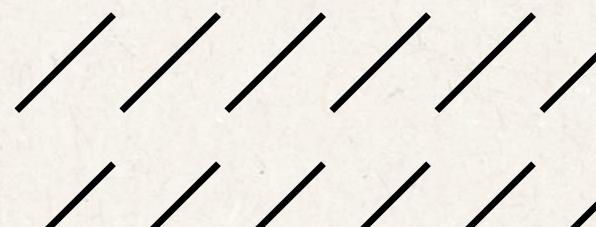
8. Redistribution Recommendations
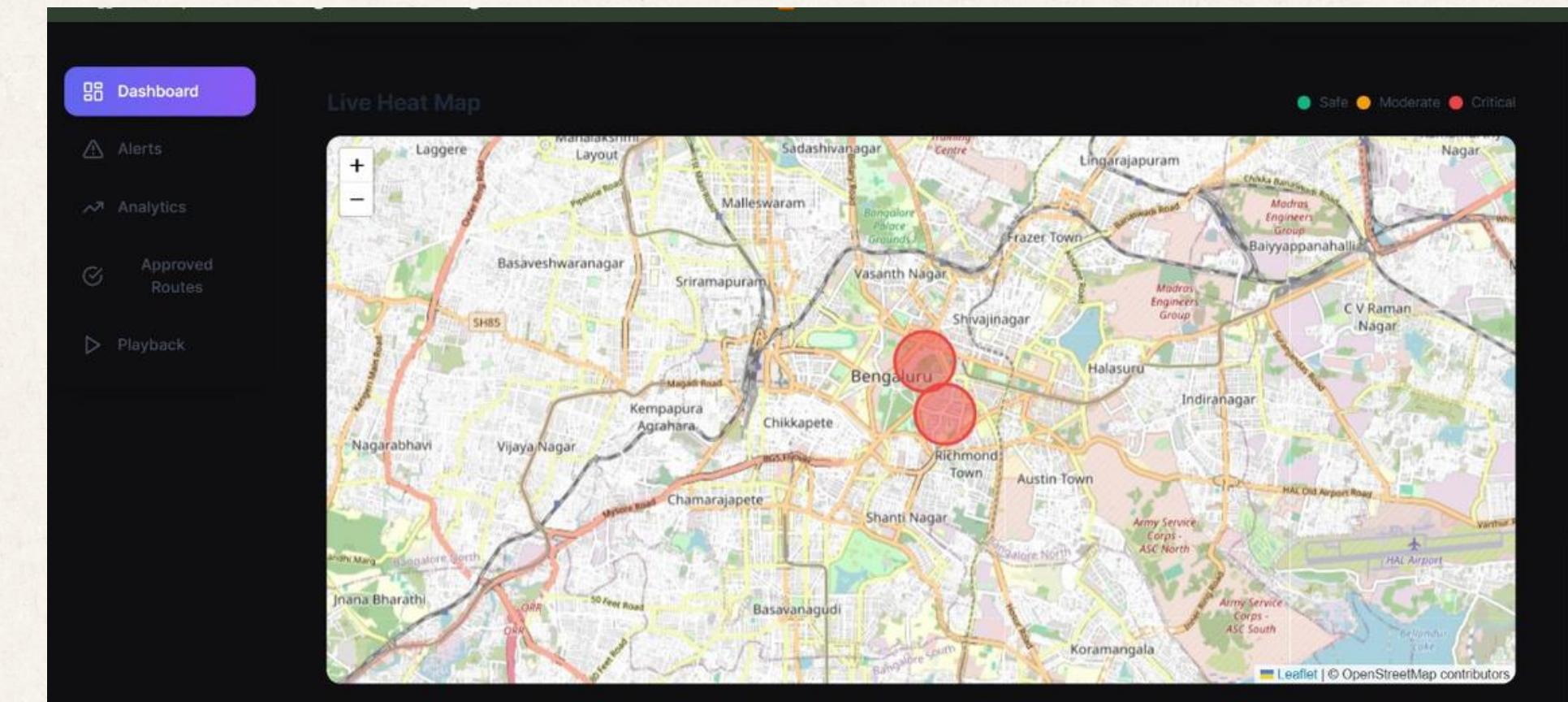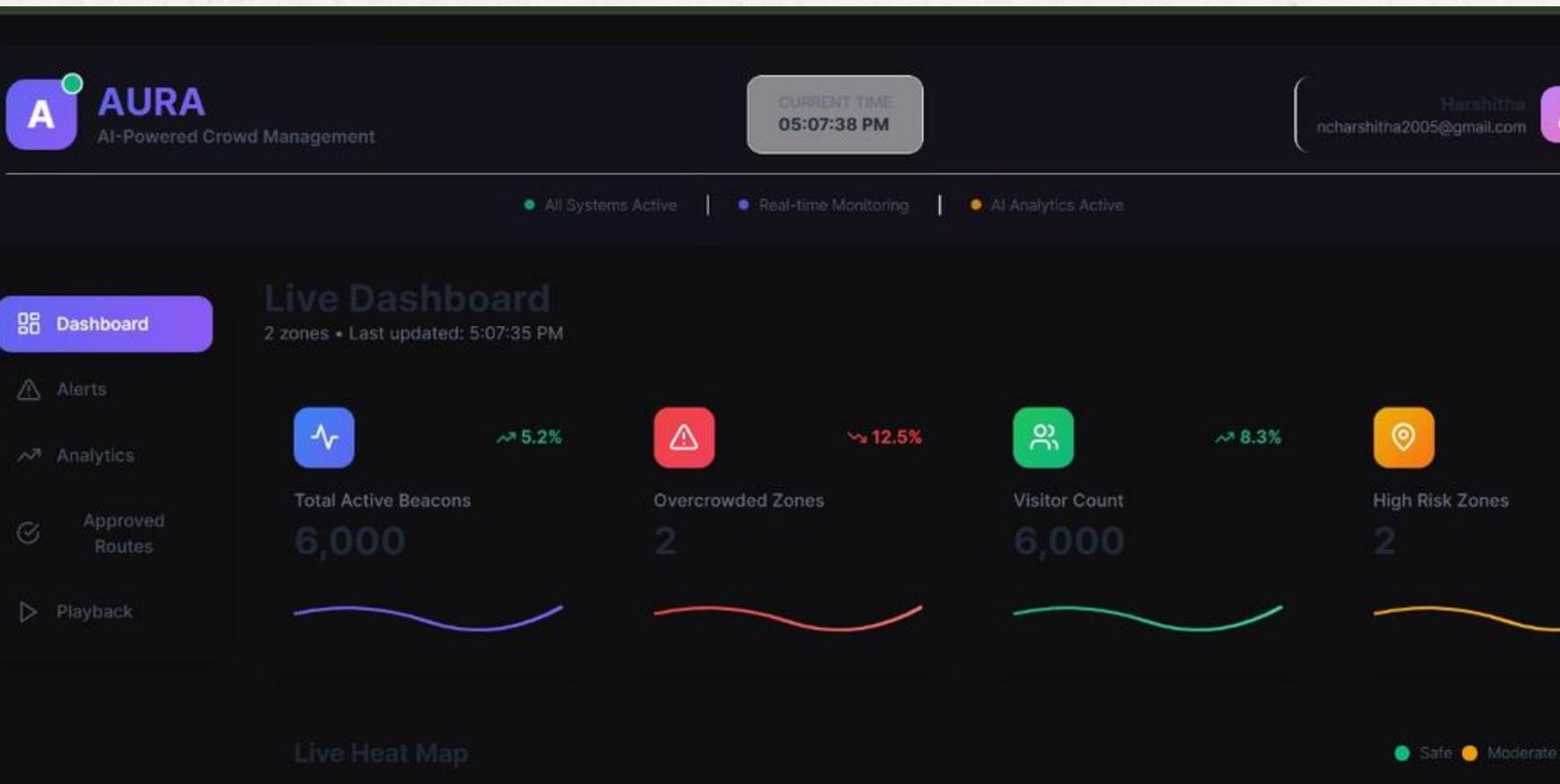
$$\text{redirect} = 0.4 \times \text{population}$$

- Identifies high-risk zones (RED/ORANGE, high CPI, panic, huddle).
- Finds safe relief zones (GREEN/low-YELLOW).
- Uses ELBS to choose the best exit (underused > busy > overloaded).
- Produces actionable commands like:

"Redirect 200 people from Zone A3 → Zone B1 via Exit E2."

Admin Console

Dashboard and Live Heatmap

# Admin Console

Analytics

Alerts
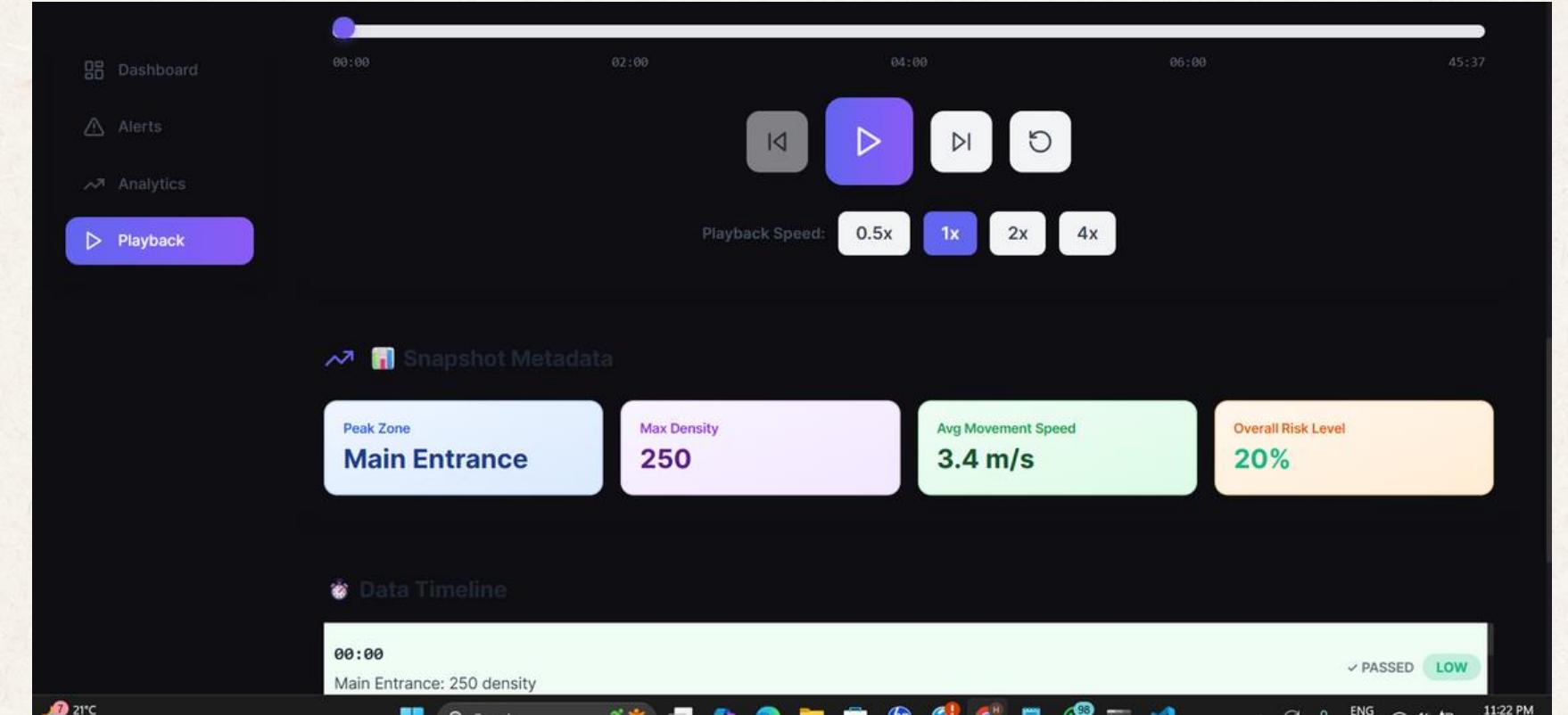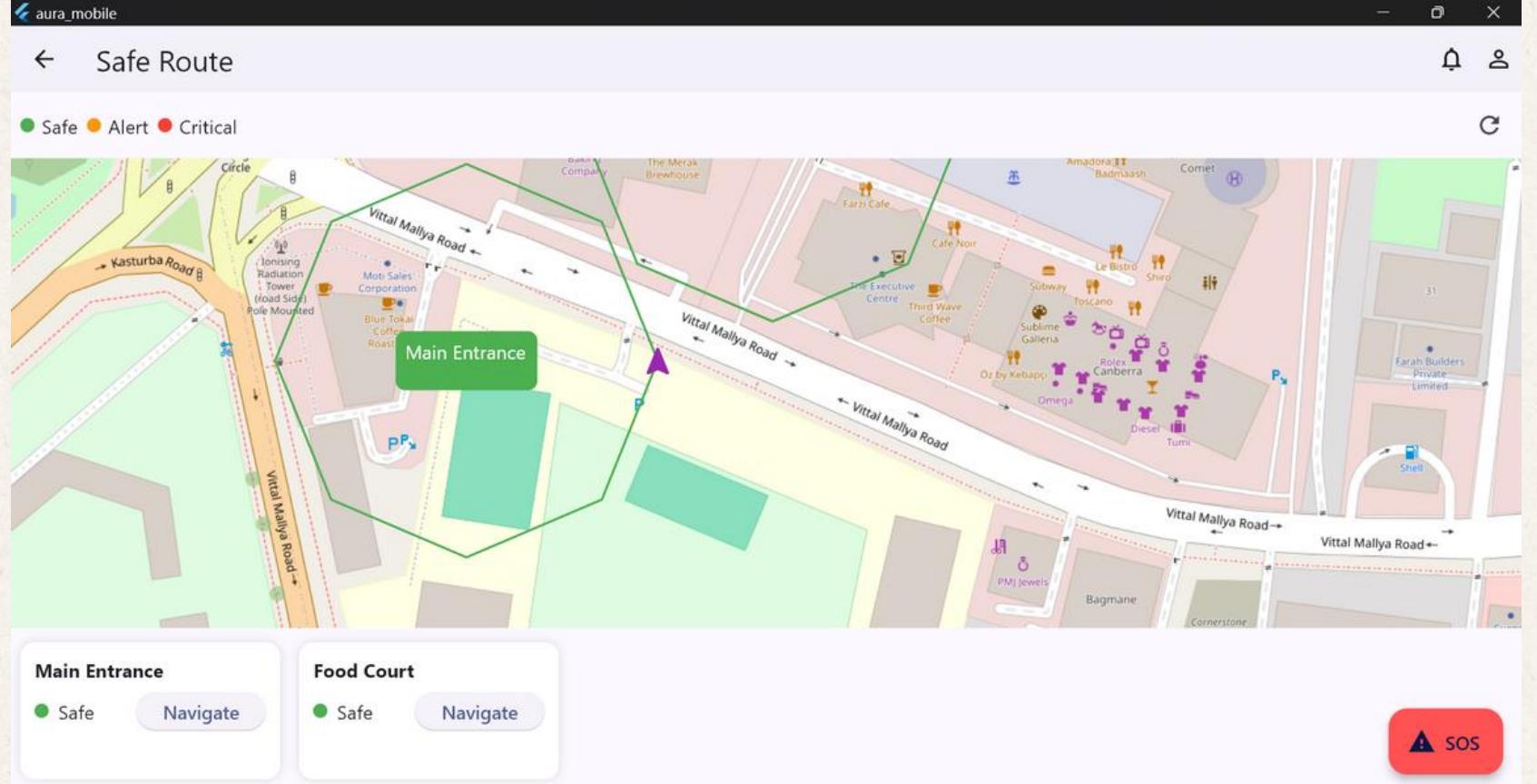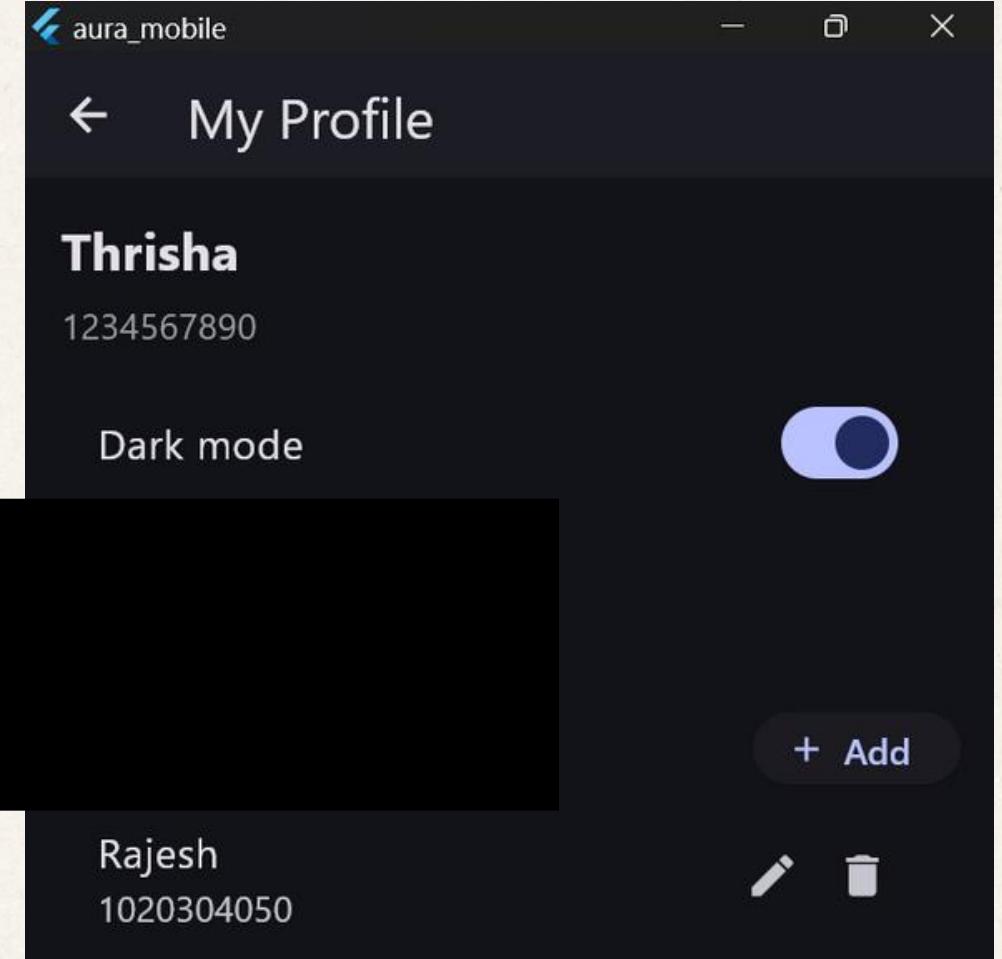

Playback Mode
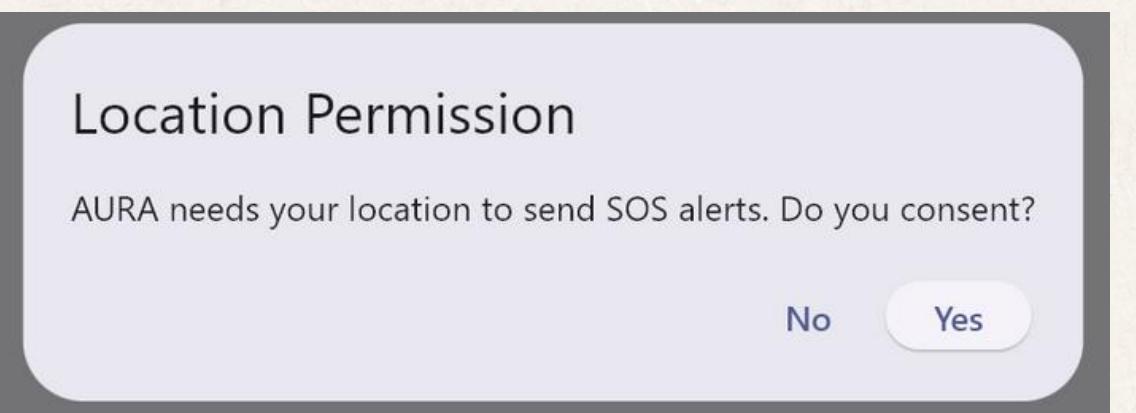
User Interface

Safe Route Guidance

Profile – Emergency Contact

Location Access - SOS

Revenue Model & Implementation Cost

Revenue Model

- SaaS Subscription Licensing – organisations (stadiums, malls, metro, event organizers) pay a monthly/annual fee for AURA's dashboard, risk engine, and safe-route analytics. (Industry-standard SaaS model used by all cloud-based security and analytics platforms.)
- Pay-Per-Event Pricing – for concerts, festivals, marathons, rallies; clients pay only for the event duration, reducing entry barriers. (Used by event-security analytics firms and temporary deployment companies.)
- Integration & Deployment Services – one-time charges for connecting AURA to existing CCTV systems, cloud setup, and training teams. (Standard in enterprise security deployments.)

Implementation Costs

These costs come from AWS/Azure cloud calculators, market rates for video-analytics, and standard CCTV infrastructure pricing ranges.

1. Camera / Input Infrastructure
   - Existing CCTV (most venues already have): ₹0 cost to AURA
   - New cameras (if needed): ₹8,000 – ₹25,000 per camera → standard India CCTV pricing range
2. Cloud Compute for Video Analytics
   - Running real-time analytics on AWS/Azure for a medium venue:
   - ₹18,000 – ₹45,000/month
3. Backend + Dashboard Hosting
   - Cloud hosting (API, database, dashboard):
   - ₹5,000 – ₹12,000/month
4. Mobile App Notifications + Traffic
   - Minimal cost due to lightweight JSON updates:
   - ₹1,000 – ₹2,000/month
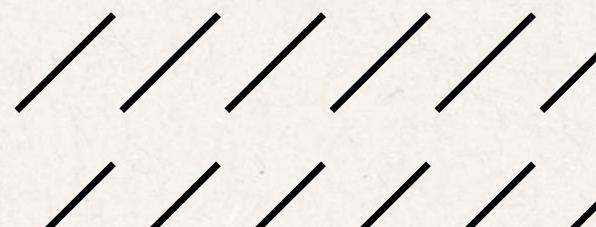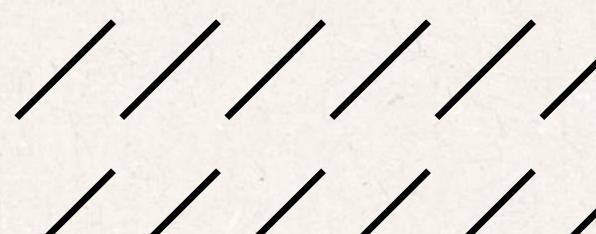
Future Scope

- Integration with Drones for aerial crowd analysis

- Thermal Camera Support to detect stress, overheating & medical emergencies

- Audio-Based Risk Detection (screams, abnormal noise spikes)

- Integration with Police & Emergency Networks for real-time coordination

- Wearable Device Alerts (wristbands, staff badges)

- Scalability to Smart Cities with multi-event, multi-location control

- Voice Alerts & Multi-language Support for inclusivity

Thank you!