5G Technologies
24-Hour Hackathon

CODE • INNOVATE • TRANSFORM

Problem Statements

Vellore Institute of Technology, Chennai
School of Electronics Engineering (SENSE)

Starts:	7th February 2026, 5:00 PM
Ends:	8th February 2026, 5:00 PM
Venue:	Kasturba Hall (MG Auditorium Basement)
Mode:	Physical Only
Team Size:	Max 4 Members

Note to Participants: Each team must choose **one** theme from the ten below and build a **working prototype or demonstrable proof-of-concept** within 24 hours. The problem statements are intentionally broad — teams are encouraged to **define their own specific scope**, choose their own approach, and surprise us with creative solutions. You may bring your own laptops, microcontrollers, sensors, or any other hardware. Internet access will be provided at the venue. Most problems can be solved with **software only** — hardware integration is a bonus.

General Rules & Evaluation Criteria

What We Expect

- A **working demonstration** — not slides, not a report, not an AI-generated write-up.
- Real data flowing through a real system (live inputs, real processing, measurable outputs).
- A clear explanation of the **problem you chose to solve**, why it matters, and how your solution addresses it.
- Teams must explain their **architecture decisions** and show **live metrics/logs** during evaluation.

Evaluation Criteria

Criterion	Weight
Working prototype with live demonstration	25%
Problem definition, novelty & creative approach	20%
System architecture & design quality	15%
Real-time data handling and measured performance metrics	15%
Relevance to 5G ecosystem (network slicing, edge computing, URLLC, mMTC concepts)	15%
Presentation, clarity of explanation & team collaboration	10%

Anti-AI-Generation Policy

- Judges will conduct a **live code walkthrough** — every team member must explain their contribution.
- Teams must show **git commit history** (GitHub/GitLab) demonstrating incremental development over the 24 hours.
- Evaluation is on the **working system under live testing** — judges will provide surprise inputs, modify parameters, and stress-test your solution on the spot.
- Directly copy-pasting AI-generated code without understanding will be caught during the walkthrough and will lead to disqualification.

What You'll Have

- Your own laptops and any hardware you bring.
- Internet access (WiFi at the venue).
- Power supply at your workspace.
- Open-source tools, public APIs, public datasets — all allowed.

Hardware Bonus: For every theme, teams that integrate **self-brought hardware** (ESP32, Arduino, Raspberry Pi, sensors, cameras, actuators) into their solution will receive **bonus points** during evaluation. However, **all problems are fully solvable with software alone**.

Programme Schedule

Venue: Kasturba Hall (MG Auditorium Basement), VIT Chennai

Date: 7th February 2026 (Saturday) 5:00 PM — 8th February 2026 (Sunday) 5:00 PM

Sl.	Date & Time	Activity
1	7 Feb, 5:00 PM – 6:00 PM	**Inauguration**, Team Registration & Guidelines
	7 Feb, 7:00 PM – 8:00 PM	*Dinner* (Flexible — Gymkhana / Gazebo)
2	7 Feb, 6:00 PM – 12:00 AM	Working on Problem Statement
3	8 Feb, 12:00 AM – 1:30 AM	**Round 1 Evaluation**
4	8 Feb, 1:30 AM – 9:30 AM	Working on Problem Statement (based on judges' feedback)
5	8 Feb, 7:00 AM – 8:30 AM	*Tea Break & Breakfast* (Flexible — Gymkhana / Gazebo)
6	8 Feb, 9:30 AM – 11:00 AM	**Round 2 Evaluation**
7	8 Feb, 11:00 AM – 3:00 PM	Working on Problem Statement (based on judges' feedback)
8	8 Feb, 1:00 PM – 2:00 PM	*Lunch Break* (Flexible — Gymkhana / Gazebo)
9	8 Feb, 3:00 PM – 4:30 PM	**Final Round Evaluation**
10	8 Feb, 4:30 PM – 5:00 PM	**Valedictory & Prize Winner Announcement**

Important Notes:

- There are **3 evaluation rounds**. After Round 1 and Round 2, judges will provide feedback — use this to improve your solution before the next round.
- Meals are at flexible timings. Teams can take breaks in rotation — at least one member should remain at the workspace.
- **Git commits will be checked** at each evaluation round to verify incremental progress.

Theme 1: Intelligent Video Analytics

PS 1 — Vision-Based Monitoring & Automated Event Intelligence

The Big Picture: Cameras are everywhere — on campuses, in traffic, at warehouses, in retail stores — but raw video is useless without intelligence. Build a system that takes live or recorded video and turns it into **actionable, real-time insights** for any domain you choose.

Core Objectives (Must Demonstrate):

1. **Live video processing** — your system must take real-time input (webcam, phone camera via IP stream, or video feed) and process it with visible, measurable throughput (show FPS or latency).
2. **Intelligent event detection** — go beyond just detecting objects. Your system should understand *what is happening* (events, behaviors, patterns, anomalies) and raise meaningful alerts or take actions.
3. **User-facing interface** — a dashboard, alert system, or interactive UI that makes the analysis accessible to a non-technical user. The end-user must be able to configure, interact with, or respond to what the system shows.

Exploration Directions (Pick any or propose your own)

- **Campus/workplace safety:** loitering detection, unattended objects, crowd density monitoring, restricted zone intrusion, after-hours activity alerts.
- **Retail analytics:** customer flow heatmaps, queue length estimation, shelf interaction tracking, dwell-time analysis for store layout optimization.
- **Traffic & parking:** vehicle counting and classification, wrong-way detection, parking occupancy monitoring, license plate recognition.
- **Industrial safety:** PPE (helmet/vest) compliance checking, forklift–pedestrian proximity alerts, hazardous zone monitoring.
- **Wildlife/environment:** animal detection and counting in camera-trap footage, illegal dumping detection in public areas.
- **Custom rule engine:** let non-technical users define their own detection rules by drawing regions on the video frame and setting conditions via a visual interface.
- **Multi-camera federation:** correlate events across multiple camera feeds (even simulated ones) for richer situational awareness.
- **Edge–cloud split:** demonstrate how processing could be split between an edge device and a cloud server, relevant to 5G edge computing.

Live Evaluation — Judges Will:

- Provide **surprise scenarios** (walk in front of camera, place objects, change conditions) and expect the system to respond.
- Test configurability — can the system be adapted on the fly without code changes?
- Ask team members to explain their detection logic and architecture.

Hardware Bonus: Use a Raspberry Pi, ESP32-CAM, or a separate device as a remote camera node streaming to your processing server.

Theme 2: Connected Healthcare

PS 2 — Intelligent Health Monitoring & Clinical Decision Support

The Big Picture: Healthcare in resource-constrained settings needs smarter tools — systems that can monitor patients, assist clinical decisions, handle unreliable connectivity, and help overworked medical staff prioritize what matters most. Build any health-tech system that processes physiological or clinical data in real time and delivers actionable intelligence.

Core Objectives (Must Demonstrate):

1. **Real-time health data processing** — your system must handle continuous physiological or clinical data (simulated or from real sensors) and process it live, not in batch mode.
2. **Intelligent analysis beyond simple thresholds** — the system must demonstrate reasoning over data: cross-parameter correlation, trend detection, anomaly classification, or clinical decision logic.
3. **Resilience to real-world conditions** — demonstrate how your system handles at least one of: network disruption, sensor failure, noisy/missing data, or resource constraints.

Exploration Directions (Pick any or propose your own)

- **Multi-patient monitoring:** simulate a ward with multiple patients, each with realistic vitals, and build an automated triage system that classifies severity using cross-parameter analysis.
- **Remote diagnostics:** build a telemedicine-ready system where a remote doctor can view patient data, annotate findings, and receive AI-assisted preliminary assessments.
- **Maternal/neonatal care:** focus on monitoring for high-risk pregnancies or newborn ICU — different vital ranges, different alarm thresholds, different clinical priorities.
- **Mental health & wellness:** use wearable-like data (heart rate variability, sleep patterns, activity levels) to detect stress, anxiety, or mood patterns and provide actionable nudges.
- **Epidemic surveillance:** process symptom reports from multiple locations and detect disease clustering or outbreak patterns on a map in real time.
- **Medication & treatment tracking:** build a system that tracks drug interactions, dosage schedules, and flags potential conflicts across multiple medications for a patient.
- **Sensor fault vs. patient emergency:** distinguish between equipment malfunctions and real clinical events using cross-validation and data quality metrics.
- **Offline-first design:** simulate 5G/4G handover and network dropouts — queue data locally, sync intelligently when reconnected, and indicate data freshness.

Live Evaluation — Judges Will:

- Inject clinical scenarios (deterioration, sensor failure, sudden events) and verify intelligent response.
- Test the system's behavior under simulated network disruption.
- Ask team to explain their clinical logic and data models.

Hardware Bonus: Connect any real health sensor (pulse oximeter, temperature sensor, heart rate module like MAX30102) to feed live data into your system.

Theme 3: Autonomous Vehicle & Drone Systems

PS 3 — Networked Autonomous Navigation & Mission Intelligence

The Big Picture: Drones, delivery bots, autonomous vehicles, and robotic systems increasingly depend on reliable network connectivity for coordination, telemetry, and remote control. Build a system that plans, simulates, and manages autonomous missions while accounting for real-world network variability and environmental constraints.

Core Objectives (Must Demonstrate):

1. **Mission planning with constraints** — your system must allow users to define missions (start, end, waypoints, objectives) while respecting constraints (no-fly zones, battery limits, payload, terrain, or network coverage).
2. **Dynamic simulation** — the planned mission must execute as an animated real-time simulation showing the vehicle's state, telemetry, and response to changing conditions.
3. **Adaptability** — the system must react intelligently when conditions change mid-mission (new obstacles, network degradation, weather changes, or competing objectives).

Exploration Directions (Pick any or propose your own)

- **Network-aware path planning:** define signal coverage zones on a map and find optimal routes that balance distance, coverage, and battery — show what happens when the vehicle enters a dead zone.
- **Multi-vehicle coordination:** manage fleets of drones/vehicles with collision avoidance, task distribution, and dynamic rebalancing when one vehicle fails.
- **Delivery logistics:** build a last-mile delivery planner that optimizes multi-drop routes considering time windows, vehicle capacity, traffic, and customer priority.
- **Search & rescue:** plan systematic area-coverage missions for disaster zones, optimizing for coverage completeness, battery swaps, and data relay back to base.
- **Surveillance & mapping:** plan photogrammetry or inspection missions with overlap constraints, altitude optimization, and real-time image stitching.
- **V2X (Vehicle-to-Everything):** simulate vehicles communicating with each other and with infrastructure (traffic signals, road sensors) to make cooperative driving decisions.
- **3D terrain navigation:** model elevation, wind conditions, or indoor floor plans and plan paths in 3D space.
- **Post-mission analytics:** generate comprehensive reports with distance covered, time in each network zone, energy consumed, events encountered, and optimization recommendations.

Live Evaluation — Judges Will:

- Define a custom mission and **add obstacles mid-flight** to test dynamic re-routing.
- Stress-test with multiple simultaneous vehicles or tight constraints.
- Ask team to explain their path-planning algorithm and decision logic.

Hardware Bonus: Integrate a physical drone, rover, or small robot that mirrors simulated movements, or use a phone as a “field unit” transmitting real GPS data.

Theme 4: IoT & Sensor Intelligence

PS 4 — Large-Scale IoT Data Platform with Real-Time Sense-Making

The Big Picture: The 5G–IoT era promises billions of connected sensors, but raw sensor data is meaningless without smart pipelines that collect, process, and extract insights from it. Build a platform that simulates or processes IoT sensor data at scale and delivers intelligent, actionable analytics.

Core Objectives (Must Demonstrate):

1. **Multi-sensor data handling at scale** — your system must work with data from multiple sensor types or nodes simultaneously (at least 10+), not just a single data stream.
2. **Intelligent analysis** — go beyond plotting raw data. Demonstrate anomaly detection, pattern recognition, cross-sensor correlation, predictive insights, or automated decision-making.
3. **Scenario injection** — provide a way to inject events, faults, or environmental changes and show how the system detects and responds to them.

Exploration Directions (Pick any or propose your own)

- **Smart campus/building:** simulate sensors (temperature, humidity, light, occupancy, power) across rooms and floors — detect anomalies, optimize energy usage, or predict maintenance needs.
- **Environmental monitoring:** air quality, water quality, noise pollution, or weather stations across a city — detect pollution events, source localization, or forecast alerts.
- **Industrial IoT:** monitor machine health, vibration, power consumption in a factory setting — predictive maintenance, efficiency optimization, or failure cascade detection.
- **Agriculture IoT:** soil moisture, temperature, light across farm zones — irrigation scheduling, frost alerts, or pest risk prediction.
- **Smart grid / energy:** simulate solar panels, batteries, and loads across a microgrid — optimize generation, detect faults, balance supply and demand.
- **Sensor fault detection:** build a system that reliably distinguishes between a sensor malfunction (drift, stuck, noise) and a real-world event using cross-sensor evidence.
- **Data pipeline visualization:** show the full journey from sensor to insight — edge aggregation, network transport (with simulated bandwidth/latency), cloud processing, and dashboard.
- **Digital twin:** create a visual spatial representation (map, floor plan, 3D model) where sensor data is overlaid in real time.

Live Evaluation — Judges Will:

- Inject scenarios (fire, equipment failure, sensor malfunction) and verify correct detection and classification.
- Add or reconfigure sensors and see the system adapt.
- Ask team to explain their anomaly detection algorithm and data pipeline architecture.

Hardware Bonus: Connect 2–3 real sensors (DHT11, LDR, MQ135 on ESP32/Arduino) whose live readings blend seamlessly with simulated data.

Theme 5: AI at the Edge

PS 5 — Adaptive AI Systems for Variable Network & Resource Conditions

The Big Picture: In the real world, AI models don't run in perfect lab conditions. Network bandwidth fluctuates, devices have limited compute, and users expect fast responses regardless. Build a system that demonstrates **intelligent adaptation** of AI inference based on changing conditions — network quality, device capability, latency requirements, or resource availability.

Core Objectives (Must Demonstrate):

1. **A working AI task** — your system must perform real inference on real inputs (images, text, audio, video, or sensor data), not just display pre-computed results.
2. **Measurable adaptation** — when conditions change (network, compute, latency budget), the system must visibly and measurably adapt its strategy (model selection, input quality, batching, offloading, etc.).
3. **Performance transparency** — show live metrics: latency, accuracy/confidence, throughput, resource usage. The user should clearly see the trade-offs being made.

Exploration Directions (Pick any or propose your own)

- **Model cascading:** use lightweight models for easy inputs and escalate to heavier models only when confidence is low — measure the efficiency gains.
- **Edge–cloud split inference:** run early layers on-device and later layers on a server, dynamically adjusting the split point based on network quality.
- **Multi-modal fusion:** combine inputs from different sources (camera + audio, text + image) and gracefully degrade when one modality is unavailable or delayed.
- **Federated or distributed inference:** split workload across multiple devices and demonstrate resilience when one device drops out.
- **Real-time translation/transcription:** speech-to-text or translation that adapts quality vs. speed based on network conditions and user-defined SLA.
- **Adaptive video analytics:** object detection that adjusts frame rate, resolution, or model complexity based on scene complexity and available bandwidth.
- **AI-powered network optimization:** use ML to predict network conditions and proactively adjust the inference strategy before degradation hits.
- **Benchmark suite:** systematically test your adaptation strategy across multiple network profiles and generate a comparative report showing the value of adaptation vs. a static approach.

Live Evaluation — Judges Will:

- Change network/resource conditions using a control panel and observe real-time adaptation.
- Provide **new input data** to verify the system isn't using pre-cached results.
- Set strict performance SLAs and verify compliance.
- Ask team to explain the adaptation policy and switching logic.

Hardware Bonus: Use two separate machines as "device" and "server" communicating over real network, with actual traffic shaping using `tc` or similar tools.

Theme 6: Smart Industry & Manufacturing

PS 6 — Digital Twin & Intelligent Process Control for Industry

The Big Picture: Industry 4.0 promises intelligent, self-optimizing factories and supply chains powered by real-time data and digital twins. Build a system that simulates, monitors, and intelligently controls an industrial process — demonstrating how 5G-enabled edge computing can transform manufacturing, logistics, or operations.

Core Objectives (Must Demonstrate):

1. **A realistic process simulation** — model an industrial process with interconnected stages where upstream changes propagate downstream. Parameters must evolve realistically over time, not randomly.
2. **Predictive or intelligent control** — demonstrate at least one form of intelligence: predictive maintenance, automated optimization, anomaly detection, or adaptive scheduling.
3. **Interactive visualization** — provide a digital twin or dashboard where the system state is visible in real time, and operators can intervene (adjust parameters, trigger maintenance, reroute work).

Exploration Directions (Pick any or propose your own)

- **Factory floor digital twin:** model assembly line stages with machines that heat up, wear out, and fail — predict failures before they happen and show cascading effects.
- **Supply chain simulator:** model suppliers, warehouses, and delivery routes — optimize inventory, respond to demand spikes, and handle supply disruptions.
- **Quality control system:** use vision or data analytics to detect defective products in a simulated production line and trace root causes back to specific machines or parameters.
- **Energy optimization:** model machines with different power profiles and optimize scheduling to minimize energy cost while meeting production targets.
- **Warehouse robotics:** simulate autonomous mobile robots in a warehouse — optimize pick paths, handle order surges, and avoid collisions.
- **Process mining:** analyze event logs from a simulated process to discover bottlenecks, deviations, and optimization opportunities automatically.
- **Multi-site coordination:** model two or more production sites connected via 5G, with coordinated scheduling and failover when one site goes down.
- **Human-in-the-loop:** design the system so that AI recommendations are presented to an operator who can accept, modify, or override — track the impact of human decisions.

Live Evaluation — Judges Will:

- ▶ Inject disruptions (machine failure, demand spike, supply shortage) and verify intelligent system response.
- ▶ Check that the digital twin accurately reflects the underlying simulation state.
- ▶ Ask team to explain their process models and prediction/optimization algorithms.

Hardware Bonus: Use a servo motor, LED array, or small actuator to physically mirror one aspect of the digital twin's state.

Theme 7: AgriTech & Environment

PS 7 — Intelligent Systems for Agriculture, Food, or Environmental Sustainability

The Big Picture: From precision farming to food supply chain optimization to environmental monitoring, technology can make agriculture more productive and sustainability more measurable. Build a system that uses real-time data, simulation, and intelligence to address a challenge in agriculture, food systems, or environmental management.

Core Objectives (Must Demonstrate):

1. **Domain-specific data modeling** — your system must work with realistic data for the domain (soil conditions, weather, crop growth, water levels, pollution, supply chain metrics, etc.) that evolves meaningfully over time.
2. **Actionable intelligence** — the system must generate specific, justified recommendations or automated actions — not just visualizations.
3. **External data integration** — incorporate at least one external real-time data source (weather API, satellite imagery, market prices, public environmental data, etc.).

Exploration Directions (Pick any or propose your own)

- **Precision farming:** simulate a farm with multiple zones, integrate live weather data, model crop growth stages, and recommend zone-specific irrigation, fertilization, or pest control actions.
- **Water resource management:** model a reservoir, canal, or groundwater system — optimize distribution, predict shortages, and allocate fairly among users.
- **Food supply chain:** track produce from farm to retail — predict spoilage, optimize cold-chain logistics, minimize waste, or verify freshness.
- **Forest fire / wildfire risk:** integrate weather, terrain, and vegetation data to create a dynamic fire-risk map with early warnings and resource deployment recommendations.
- **Carbon footprint tracker:** model emissions from agriculture, transport, or industrial operations and suggest optimization strategies with measurable impact.
- **Fisheries management:** model fish stocks, weather patterns, and fishing effort to recommend sustainable catch limits and detect illegal fishing patterns.
- **Urban farming / vertical farming:** simulate an indoor farm environment and optimize lighting, nutrients, and climate control for maximum yield per unit energy.
- **Pest/disease prediction:** use weather, soil, and historical data to predict pest outbreaks or crop disease and issue early warnings with suggested interventions.

Live Evaluation: Judges will inject scenarios (drought, pest outbreak, supply disruption, sensor failure) and verify that the system responds with intelligent, prioritized recommendations. They will also change the location/context and check if external data integration adapts.

Hardware Bonus: Connect a soil moisture sensor, DHT11/22, light sensor, or any environmental sensor to feed real data into your system.

Theme 8: Smart City & Urban Systems

PS 8 — Intelligent Urban Infrastructure Simulation & Optimization

The Big Picture: Smart cities need systems that can simulate, monitor, and optimize urban infrastructure — from traffic to energy grids to public safety to waste management. Build a platform that models any aspect of urban life and demonstrates how 5G-connected intelligence can make cities more efficient, safer, or more livable.

Core Objectives (Must Demonstrate):

1. **A city-scale simulation** — model a meaningful urban system with multiple interacting components (intersections, buildings, routes, zones, etc.) that behave realistically.
2. **At least two competing strategies** — implement different approaches to managing the system and show measurable, side-by-side comparison of their performance.
3. **Dynamic event handling** — the system must respond intelligently to real-time events (accidents, emergencies, demand surges, infrastructure failures) injected during the demo.

Exploration Directions (Pick any or propose your own)

- **Traffic management:** simulate intersections with adaptive signal control, emergency vehicle green corridors, and congestion prediction.
- **Public transit optimization:** model bus/metro routes with real-time demand — dynamically adjust frequency, routing, or capacity allocation.
- **Smart parking:** simulate a city district with parking spaces, guide drivers to optimal spots, and predict availability.
- **Waste management:** model bin fill levels across a city and optimize collection routes dynamically based on real-time data.
- **Energy distribution:** simulate a city power grid with renewables, storage, and demand — optimize distribution and handle outages gracefully.
- **Emergency response:** model ambulance/fire-truck deployment, optimize response times, and handle multi-incident scenarios.
- **Noise/pollution mapping:** model noise or air quality across a city and suggest mitigation strategies (rerouting traffic, timing construction, green corridors).
- **Citizen reporting platform:** build a system where citizens can report issues (potholes, broken lights, flooding) and the city can prioritize and dispatch crews optimally.
- **Multi-system integration:** show how two urban systems interact — e.g., how a traffic reroute affects air quality, or how a power outage impacts traffic signals.

Live Evaluation: Judges will inject events (accidents, emergencies, infrastructure failures, demand surges) and verify intelligent response. They will compare metrics across different strategies and stress-test the system's limits.

Hardware Bonus: Use LEDs to physically mirror signal/status states, or use a phone as a "field unit" reporting real GPS data into the system.

Theme 9: Smart Education & Human–Computer Interaction

PS 9 — Adaptive & Intelligent Learning or Interaction Systems

The Big Picture: Technology can make learning more personalized, accessible, and engaging — or make human–computer interaction more natural and responsive. Build a system that **adapts to its user in real time** based on behavioral signals, performance data, or contextual cues. This can be in education, training, accessibility, or any interactive domain.

Core Objectives (Must Demonstrate):

1. **Real-time user sensing** — your system must observe something about the user in real time (facial expressions, gaze, voice, interaction patterns, quiz performance, typing behavior, etc.) and use it meaningfully.
2. **Adaptive response** — based on what the system senses, it must change its behavior (content difficulty, pacing, interface layout, recommendations, feedback style, etc.) within seconds, not minutes.
3. **Measurable personalization** — show evidence that the adaptation improves outcomes: engagement scores, learning metrics, task completion rates, or user satisfaction.

Exploration Directions (Pick any or propose your own)

- **Adaptive learning platform:** present educational content that adjusts difficulty, adds explanations, or skips ahead based on student engagement and quiz performance.
- **Engagement detection:** use webcam to detect attention, confusion, or drowsiness and trigger appropriate interventions (simpler content, break suggestions, interactive elements).
- **Language learning:** adaptive pronunciation practice, vocabulary drills, or conversation practice that adjusts to the learner's level and weak areas.
- **Skill training simulator:** build a training system for any skill (typing, coding, medical procedures, equipment operation) that adapts challenge level to the trainee's performance.
- **Accessibility tool:** build a system that adapts interfaces for users with different abilities — text size, contrast, voice control, simplified navigation — based on detected user behavior.
- **Teacher/instructor dashboard:** aggregate data from multiple students/users and provide real-time insights — who needs help, where the content is confusing, and recommended intervention points.
- **Collaborative learning:** build a platform where groups work together, and the system detects imbalanced participation and nudges quieter members to contribute.
- **Gamified assessment:** design an assessment that feels like a game, with adaptive difficulty and real-time feedback, generating a detailed competency report at the end.
- **Interview/presentation coach:** analyze speech patterns, filler words, pacing, and body language during a practice session and provide real-time and post-session feedback.

Live Evaluation: A judge will use the system directly — deliberately performing well, poorly, or acting disengaged — and verify that the system detects and adapts within seconds. Session analytics and reports will be reviewed.

Hardware Bonus: Add a secondary input modality — ambient noise via microphone, heart rate via smartwatch API, or gaze tracking via external camera — to enrich the sensing.

Theme 10: Crisis & Disaster Management

PS 10 — Intelligent Crisis Response & Coordination Platform

The Big Picture: During crises — natural disasters, pandemics, industrial accidents, mass-casualty events, or even large-scale infrastructure failures — coordination is the difference between effective response and chaos. Build a platform that helps authorities, responders, or communities manage any type of crisis more effectively using real-time data, intelligent allocation, and communication under degraded conditions.

Core Objectives (Must Demonstrate):

1. **Situational awareness** — provide a real-time, evolving picture of the crisis on a map or spatial interface, with affected areas, resources, and requests visually represented.
2. **Intelligent resource allocation** — when help requests come in, the system must assign resources based on priority, proximity, capacity, and constraints — not just first-come-first-served.
3. **Resilience under degraded conditions** — demonstrate how the system handles communication failures, delayed data, duplicate reports, or partial information.

Exploration Directions (Pick any or propose your own)

- **Natural disaster response:** floods, earthquakes, cyclones — manage rescue teams, shelters, medical aid, and supply distribution across affected zones with blocked roads and damaged infrastructure.
- **Pandemic management:** track case clusters, hospital capacity, vaccine distribution, and containment zones — optimize resource deployment as the situation evolves.
- **Industrial accident:** chemical spill, factory explosion, or mine collapse — model evacuation zones, hazmat spread, and coordinate specialized rescue teams.
- **Urban crisis (fire, building collapse):** model a specific scenario in detail with floor plans, trapped persons, structural integrity assessment, and rescue team coordination.
- **Crowd management:** festivals, protests, or stadium events that escalate — detect crowd density, predict crushes, manage exits, and deploy security.
- **Community resilience:** build a platform where citizens can report needs, volunteer resources, and self-organize — with verification and intelligent matching.
- **Multi-agency coordination:** model different agencies (police, medical, fire, military) with separate resource pools and enable cross-agency resource sharing with approval workflows.
- **Communication resilience:** simulate mesh networking, store-and-forward messaging, or priority-based bandwidth allocation for critical messages when the network is degraded.
- **Post-crisis analytics:** generate situation reports, response time analysis, and lessons-learned summaries from the simulated crisis data.

Live Evaluation: Judges will inject new crisis events mid-demo (expanding disaster zone, mass-casualty event, blocked routes, network outage) and verify intelligent, prioritized response. They will test how the system handles delayed and duplicate reports.

Hardware Bonus: Use a phone sending GPS coordinates as a "field rescue unit" that updates position on the map in real time, or use LEDs/buzzers to indicate alert levels.

Problem Statements Summary

#	Theme	Core Challenge
1	Intelligent Video Analytics	Real-time vision system that detects events and delivers actionable insights from live video
2	Connected Healthcare	Health monitoring with intelligent clinical reasoning and resilience to real-world conditions
3	Autonomous Navigation	Mission planning and simulation for drones/vehicles with dynamic adaptation to constraints
4	IoT & Sensor Intelligence	Large-scale sensor data platform with anomaly detection and real-time sense-making
5	AI at the Edge	Adaptive AI inference that adjusts strategy based on network and resource conditions
6	Smart Industry	Digital twin or process simulator with predictive control and intelligent optimization
7	AgriTech & Environment	Data-driven agriculture or environmental system with real-time external data integration
8	Smart City	Urban infrastructure simulation with competing strategies and dynamic event handling
9	Smart Education & HCI	Adaptive system that senses and responds to user behavior in real time
10	Crisis Management	Crisis coordination platform with intelligent allocation and communication resilience

How to Read These Problem Statements:

- **Core Objectives** are mandatory — your solution must demonstrate all of them.
- **Exploration Directions** are suggestions — pick one, combine several, or propose something entirely different within the theme.
- **The more original and well-executed your approach, the better.** Don't limit yourself to what's listed.
- Remember: judges evaluate **working prototypes**, not ideas. Scope your ambition to what you can build and demo in 24 hours.

Good luck! Build something that works.

— Organizing Committee, SENSE, VIT Chennai