



Crisis Triage Ecosystem

Singapore's Trusted Loop from Signal to Rescue

Building clarity, safe deployment, and public calm: step-by-step

Sena Lim | [Email](#) | [LinkedIn](#)

Core Product Framework Used

Framework	Purpose
Double Diamond	Structure discovery, definition, development, and delivery of solution
Jobs-To-Be-Done	Map each persona's functional and emotional needs under crisis
Custom Prioritization Framework (Mission alignment, Value across roles, Feasibility)	Score root causes, problems, modules, and features to prioritize what matters most
Flywheel	Strategy model to reinforce trust, clarity, and speed with every cycle
Objective-based Roadmap	Align each product phase (POC → MVP → Rollout) to outcomes, not just delivery

Assumptions That Shaped Product Decisions

Prioritization (What to Build First)

	Assumption	Decision Impact	If False / Mitigation
1.1	Team has low risk appetite	P1 = mission-critical core loop only. Bold bets delayed to P3.	Restructure roadmap / combine phases / bold bets can shift earlier post-MVP trust signals
1.2	GovTech assets are reusable	Prioritize modules that can be built “on top”; faster build = higher feasibility in P1 prioritization	POC validates reuse feasibility per module. If blocked, fallback to standalone version or simplify feature
1.3	Public-facing updates reduce panic & noise	Rescue Status Tracker included in P1; essential to reduce repeat calls and reassure civilians	If public channels are unreliable, fallback to SMS push only or PA broadcast protocol

Strategy (How to Build Trust and Scale)

	Assumption	Decision Impact	If False / Mitigation
2.1	Decision-making is hierarchical	Manual override rights built into every module; Commander has final say	If Command delegates control, more autonomous AI could be enabled in P3 (future state)
2.2	Toggle-based control is culturally accepted	Enables modular fallback and supervisor trust reviews. Strategy supports partial rollout	If toggling is resisted, switch to fixed phasing with internal fallback drills only
2.3	AI triage must earn trust before scaling	MVP is gated on triage accuracy and supervisor adoption; builds credibility first	If accuracy fails, human-in-loop review is permanent. Explainability labels added in P2

Solution Concept (Module Design and Loop Logic)

	Assumption	Decision Impact	If False / Mitigation
3.1	Team must stay mission-ready, minimal time to train	Reuse of GovTech assets; solution designed with familiar UX & minimal friction	If reuse still confuses, run quick daily micro-trainings. Include visual SOP handbooks
3.2	GovTech assets are reusable (repeat)	All core modules build “on top”; reduces build time and avoids policy blockers	Same as above. Redesign feature as standalone microservice if needed
3.3	Public feedback loop reduces panic & noise (repeat)	Tracker closes the loop between distress report and public calm. Key to flywheel logic.	If ineffective, delay Co-Pilot; reinforce Command-wide broadcasts instead
3.4	System must scale in modular layers	All components (triage, tasking, tracker, etc.) designed for standalone test & toggle	If ops demand tighter integration, bind tasking+triage+tracker into a single MVP interface

Rollout Sequencing (Why Phased, What Tested When)

	Assumption	Decision Impact	If False / Mitigation
4.1	Command-driven culture avoids big bang launches	Phased rollout adopted. POC → MVP → Rollout all trust-gated	If Command demands big bang, propose core-only launch + fallback plan pre-approved
4.2	Toggle control acceptable (repeat)	Enables rollback at any point; fallback drills part of quarterly ops	If toggling is rejected, revert to fixed milestone-based de-risked rollout
4.3	AI must prove trust before rollout (repeat)	Every phase is gated by operator trust + override logs	If trust signals aren’t clear, pause rollout. Add side-by-side validation mode
4.4	Modular rollout is viable	Each core component is tested separately; minimal cross-dependency in early phases	If full integration is required, start with bundled MVP of triage+tasking+tracker
4.5	Sandbox and drills are feasible	POC runs in sandbox using synthetic/historic data; MVP zone tested via mock drills	If live drills are blocked, use real flood archives + tabletop simulation only

Scenario: Record-Breaking Floods

Severe flooding disrupts multiple low-lying districts:

- ⚡ Power outages hit older HDB blocks
- 🚇 MRT lines suspended in flood zones
- 🏥 Hospitals face surge in trauma & infections
- 📱 Social feeds show families trapped, elderly stranded, children separated
- 🌧️ More rain expected in next 12 hours

Key Realities for Command & Field Teams:

- 📡 Thousands of fragmented incident reports: real, outdated, duplicate
- 🚑 Field teams must deploy fast but lack clear assignments
- li>• 😞 Civilians panic with no trusted rescue updates
- 🙋 Volunteers & donations surge but remain chaotic

A crisis loop overwhelmed by noise, confusion, and panic.





Persona: Field Responders & SAF

Who:

- Rescue squads, roadblock teams, evacuation boat crews working in rapidly changing flood zones

Key Needs:

Functional:

- Clear, real-time task orders
- Safe, updated routes that adapt to changing conditions
- Simple check-in and status reporting tools

Emotional:

- Feel protected and guided, even in unpredictable terrain
- Trust that task info is reliable, so they can act confidently

Key Challenges:

- Blocked or changing routes
- Unclear priorities when calls overlap
- Fatigue & unsafe working conditions
- Limited boats/equipment



Persona: Crisis Command & Control (C2)

Who:

- National Ops Centre staff, planners, deployment commanders coordinating multiple regions under crisis surge

Key Needs:

Functional:

- Accurate, deduplicated incident data
- Clear severity ranking to spot critical cases first
- Unified situational picture across zones

Emotional:

- Feel in control under data chaos
- Trust that the triage signals are reliable

Key Challenges:

- Flooded with duplicate, false, and stale reports
- Manually sorting and validating under time pressure
- Missing real-time field updates



Persona: Civilians

Who:

- Families trapped in homes, elderly alone, stranded commuters needing rescue and reassurance

Key Needs:

Functional:

- Simple way to request help
- Clear, real-time rescue status & ETA

Emotional:

- Feel reassured help is truly on the way
- Less panic and repeat distress calls

Key Challenges:

- Limited comms (power out, no data)
- Rumors and misinformation spread fast
- Anxiety when no rescue confirmation is received



Persona: SCDF Paramedics & Medics

Who:

- On-site paramedics, mobile field hospitals, medical staff triaging patients during crisis surges

Key Needs:

Functional:

- Quick, digital triage tools
- Real-time sync with hospitals and supply status

Emotional:

- Feel supported during high-pressure surges
- Reduce moral stress when resources are tight

Key Challenges:

- No patient history at site
- Shortage of meds, beds, or supplies
- Manual hand-offs and paper-based triage slow care



Persona: Donors & Volunteers

Who:

- Individual donors, NGOs, community shelters wanting to help with supplies and manpower

Key Needs:

Functional:

- Verified, real-time list of shortages
- Clear safe drop-off routes and instructions

Emotional:

- Feel confident their help is needed and well-used
- Avoid wasting resources or adding chaos

Key Challenges:

- Hard to verify credible needs
- Risk of duplicate or mismatched aid
- No clear safe path for volunteers to reach shelters

Root Causes Behind the Chaos

Fragmented & Duplicate Data Flow

Duplicated or fake reports flood in via calls, SMS, social media: no single pipe

Manual Triage Bottleneck

Staff must manually read, sort, and verify reports

Static Tasking & Routing

Field teams get static tasks; no dynamic reroute when conditions shift

Poor Civilian Feedback Loop

People submit distress calls but get no confirmation or status

Medical Bottlenecks

Paramedics lack patient records; paper-based triage; supply status unclear

Ad Hoc Donation Flow

Donations and volunteers surge but are not matched to verified shortages; safe routing unclear

These deep system gaps multiply panic, slow rescue and waste effort.

Prioritized Root Causes

Prioritized using: Mission Alignment, Value Unlocked, Feasibility within policy constraints.

Root Cause	Priority	Rationale
Fragmented & Duplicate Data	P1	Overloads Command, creates confusion downstream
Manual Triage Bottleneck	P1	Slows everything from dispatch to rescue
Static Tasking & Routing	P1	Makes it unsafe for field teams to adapt
Poor Civilian Feedback Loop	P1	Drives panic and repeat alerts
Medical Bottlenecks	P2	Slows care; smaller loop impact
Ad Hoc Donations Flow	P3	Wastes goodwill; non-core loop

P1s shape POC & MVP; P2s & P3s follow once core loop earns trust.

Prioritized Problems & Opportunities

Root Cause	Priority	Problem Statement	Opportunity Statement
Fragmented & Duplicate Data, Manual Triage Bottleneck	P1	Command teams cannot trust incident data because reports are fragmented and duplicated	HMW help Command cut through data noise and focus on the most critical incidents so they stay calm and deploy help quickly?
Static Tasking & Routing	P1	Field responders risk confusion and unsafe routes because tasking stays static even when ground conditions change	HMW keep frontline teams updated with clear missions and safe routes so they avoid danger and stay confident in changing conditions?
Poor Civilian Feedback Loop	P1	Civilians panic and make repeat calls because they do not receive clear rescue confirmation	HMW make it easy for civilians to report distress and feel reassured by knowing when help is coming?
Medical Bottlenecks	P2	Paramedics and medics struggle to triage patients and sync supplies because they lack patient history and real-time stock visibility.	HMW help medics triage patients quickly and coordinate with hospitals so they feel supported and save more lives under surge?
Ad Hoc Donations Flow	P3	Donors and volunteers risk sending aid that does not match real needs because verified shortage lists and clear drop-off routes are missing	HMW guide donors and volunteers to verified needs and safe ways to help so they feel useful and avoid adding to chaos?

These are the human problems that rooted in system gaps.
We tackle the P1s first to secure the core loop. P2s and P3s strengthen resilience step-by-step.

Crisis Triage Ecosystem: Strategy







Goal:

Protect lives and sustain public trust by closing the loop from signal to rescue, under any crisis scale.

Flywheel:



Guiding Principles: Our Guardrails:

- | | | |
|---|--|--|
|  Mission-Critical First
Triage, dynamic tasking, civilian calm: non-negotiable |  Minimum Viable Loop
Build only what completes the core rescue loop first |  Layered Scale
Starts local; expands multi-zone with zero downtime |
|  Trust by Design
Real-time updates cut panic and repeat calls |  Resilient & Usable
SMS, voice, mobile; fallback-ready; low training burden |  Policy-Aligned
Security, privacy, and command audits at every step |

These guardrails steer every design, delivery, and scale decision.

Crisis Triage Ecosystem: Solution Concept

Singapore's crisis backbone:

Strengthening any one enabler powers the whole loop: clearer signals, sharper triage, safer tasking, calmer public, cleaner next signals.



Brain: AI Triage Engine

- Deduplicates & ranks reports in real time
- Filters noise for accurate triage



Arms & Legs: Dynamic Tasking

- Sends live mission updates & safe routes
- Keeps responders protected & focused



Heartbeat: Rescue Status Tracker

- Civilians request help & see live ETAs
- Builds public calm & trust



Circulation: Med Triage & Supply Sync

- Field medics link triage to hospital loads & supply stock



Immune System: Verified Donations Board

- Community aid matches real, verified gaps



Backbone: Shared Command Dashboard

- Feeds all triaged signals, tasks, and status into a single live picture for HQ: the trusted backbone at any crisis scale

Every layer deepens the backbone: proven first with the core loop, then strengthened step-by-step with medical sync, surge prediction, and community trust features

Trusted Base: Extend Only Where Needed

Reuse proven national tools to keep build cost low and adoption easy:

- **GovTech NLP Microservices:** base text classification & deduplication
- **Dispatcher SOP Rules:** severity scoring logic
- **Call Transcript Pipeline:** voice-to-text input for triage
- **OneMap Routing & Clusters:** smart routes, incident heatmaps
- **C2 Task Log & Roles:** task assignments & permissions
- **FormSG:** secure public distress form
- **GovSMS & WhatsApp API:** official fallback comms & status push
- **SGSecure, [Gov.sg](https://www.gov.sg):** trusted crisis broadcast channels
- **eMEDS & NEHR Connector:** medics link triage to hospitals
- **Smart Logistics APIs:** live stock & supply tracker
- **OneService & SG Cares:** verified donations & trusted community ID

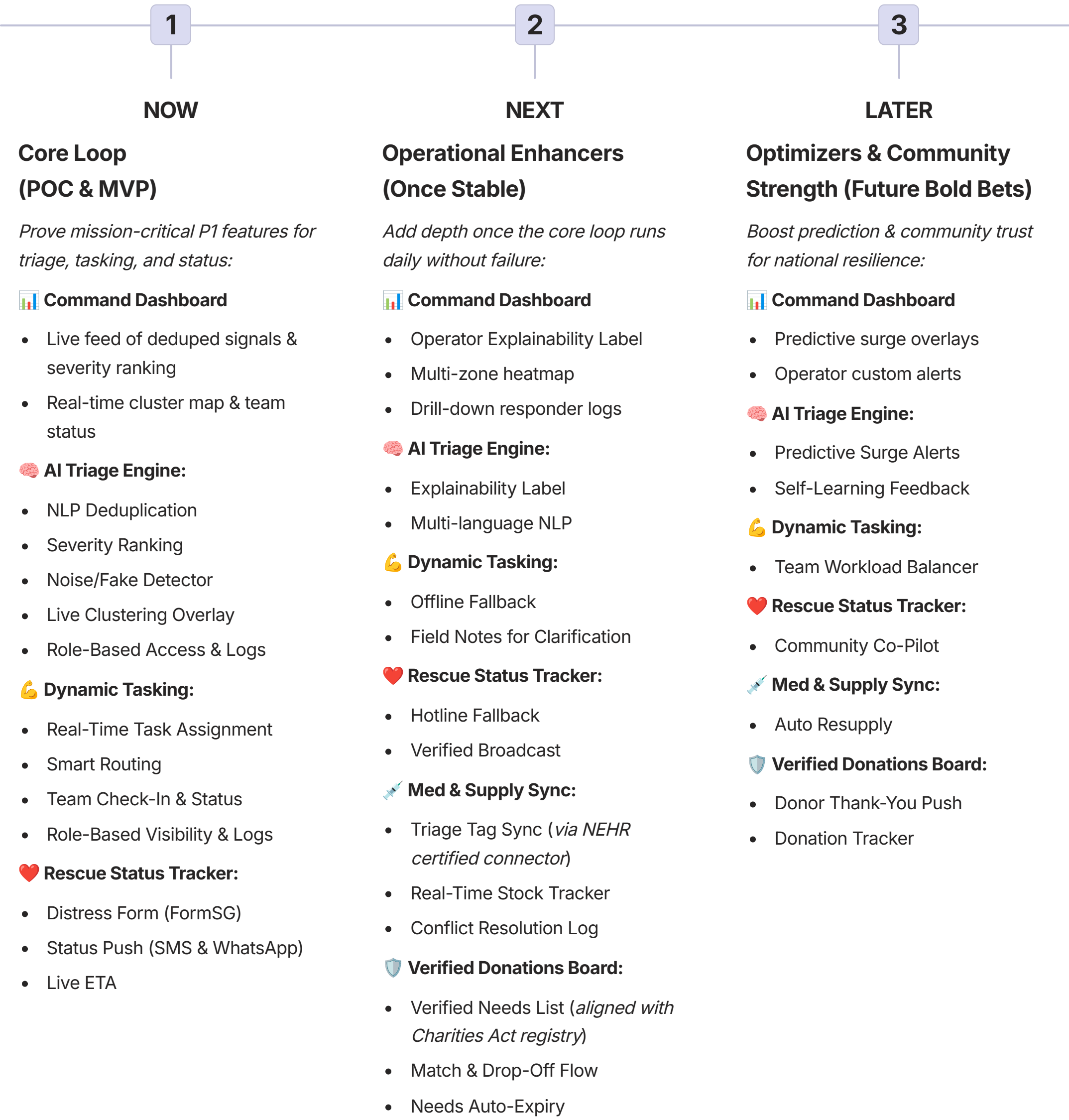
*Trusted base, extended only where needed: **faster adoption, lower cost, zero waste.***



Roadmap: Build the Trusted Loop, Layer by Layer

Each layer deepens the Shared Command Dashboard, growing from core loop to full national crisis picture

Prioritized using: Mission Alignment, Value Unlocked, Feasibility within policy constraints.



Proof of Concept (POC)

Goal:

Prove the core loop (NOW scope) works technically and is adoptable in controlled drills.

Key Focus:

- Validate **AI Triage**: deduplication, severity ranking, clustering
- Test **Dynamic Tasking**: live task push, check-in, safe routing
- Trial **Status Tracker**: distress form, push notifications, live ETA

 **Outcome:** Proven loop is technically feasible and safe to move to MVP pilot

Success Metrics:

90%

NLP accuracy

30 sec

Task assignment speed

95%

Drill check-in success

30 sec

Status push delivery

Safeguards:

- Sandbox & drills only: no live crisis
- Manual rollback always ready
- Supervisors review trust signals & logs

MVP (Live Stress Test)

Goal:

Run the core loop live in daily ops, proving reliability under real surge conditions

Key Focus:

- Live signals deduplicated & ranked on the fly
- Real-time tasking adapts as conditions change
- Civilians get trusted updates that reduce panic

 **Outcome:** Core loop stable in live ops, safe to scale zone-by-zone

Success Metrics:

90%

NLP live accuracy

30 sec

Task assignment speed

95%

Check-in success

30 sec

Status push delivery

Controls:

- Pilot in a single high-risk zone
- Daily usage logs + supervisor review
- Toggle & manual SOP always on standby

Full Rollout

Goal:

Expand the proven loop to all high-risk zones, embed as standard daily SOP

Key Actions:

- Scale core modules to 100% high-risk coverage
- Supervisor sign-offs before each zone goes live
- Operator refresher training at each phase
- Quarterly fallback drills to ensure resilience

 **Outcome:** Core loop becomes daily norm, clears gate to layer NEXT & LATER roadmap features

Monitoring Metrics:

- % zones live (100% high-risk zones covered)
- Fallback drill pass rate (100% quarterly)
- Operator trust score ($\geq 80\%$)
- SOP compliance (Zero critical failures)
- Audit trail coverage (100% user actions logged & reviewed)

Safeguards:

- Zone-by-zone rollout, not big bang
- 24/7 monitoring + clear escalation SOP
- Security & privacy checks before new feeds

Rollout Strategy

Phase	What We Do	How We Control Risk
POC (3 steps)	Test each module in sandbox → Integrate core loop in sandbox → Controlled scenario drill in safe training setting Dashboard shows sandbox signals + dummy data	Synthetic & historic data; trained teams; fallback SOP; quarterly revert drills
MVP (3 steps)	Run core modules live in a moderate-risk flood district → Gradually expand zone/load → Life stress drill Dashboard runs live in pilot zone	Small, controlled blast radius; real daily signals; toggles on' daily trust signals reviewed
Full P1 Rollout	Scale proven core loop to all high-risk zones Dashboard becomes daily situational command tool	Zone-by-zone scale; supervisor sign-off; fallback SOP always ready
Add P2: Med & Supply	Layer Med & Supply Sync once core loop is stable	Security/privacy checks' supervisors monitor trust signals
Add P3: Optimizers (Bold bets)	Add surge prediction, self-learning AI, Community Co-Pilot, and donor engagement tools	Only when mature; same toggles, fallback, audit trail, supervisor sign-off

Key Actions & Safeguards (Always On)

- Toggle¹ & fallback ready: Any module can switch off; revert to SOP immediately
- Quarterly fallback drills: Teams stay fluent in manual SOP
- Supervisor trust reviews: Logs & signals flagged
- Operator training & refreshers: Before each phase; keeps all levels confident
- Internal briefings each phase: Before each scale step
- Security/privacy checks: For new feeds/modules
- 24/7 monitoring + escalation SOP
- Full audit trail & override accountability

Assumptions & Trade-Offs:

- Assumes legacy SOP fallback always valid, trained teams ready before pilot, and cross-agency data sharing approved.
- Some advanced features may be toggled off in high-risk surge to prioritise stability.
- Each phase includes pivot gates: we scale only when daily trust is proven

Build trust step-by-step: prove each gate, control blast radius, no surprises.

How This Reflects My Product Practice

What this case shows:

- 1 Mapping fragility under pressure — and designing systems that respond with **clarity, calm, and care**.
- 2 Bridging urgent triage with long-term capability — so today's patch can become tomorrow's foundation.
- 3 Holding space for grief, burnout, and complexity — and still building momentum that feels safe.
- 4 Designing **governance, rituals, and role clarity** to bring structure without rigidity.

This is how I build in times of strain:

Not just to survive — but to help teams breathe, act, and rebuild with quiet strength.

[Explore more case studies](#) | [Email](#) | [LinkedIn](#)