Suraksha Chakra

A Unified Digital Shield for India's Informal Economy and Disaster Resilience

The Suraksha Chakra Team
Open Innovation Track: Fintech, AI, Governance

December 31, 2025

Abstract

India's informal economy faces a critical vulnerability paradox: the 90% of the workforce driving the nation's growth is often the most exposed to climate disasters. When urban catastrophes strike, Micro, Small, and Medium Enterprises (MSMEs) suffer a "Triple Death": Asset Loss, Paper Loss, and Identity Loss. Current solutions are fragmented across weather apps, government portals, and payment wallets. We propose **Suraksha Chakra**, a single unified web application that transforms from a daily business utility tool into a disaster resilience ecosystem. The application operates in three seamless phases: *Prepare* (building digital twins via daily use), *Survive* (offline mesh, voice triage, and family reunification), and *Recover* (instant credit via India Stack). This paper details the technical architecture, algorithms, and flow for this unified ecosystem.

---

1 Introduction

The contemporary digital landscape of India is characterized by a paradox of rapid modernization and persistent vulnerability. While the nation has leapfrogged into a digital-first economy through the proliferation of the Unified Payments Interface (UPI) and the Open Network for Digital Commerce (ONDC), the structural integrity of this new economy remains heavily dependent on continuous connectivity. The informal sector, which constitutes nearly 80% of India's workforce and contributes significantly to the GDP, has increasingly adopted digital tools for bookkeeping, payments, and supply chain management. However, this digitalization has introduced a critical dependency: the assumption of an "always-on" grid.

In the context of India's diverse and often volatile geography—prone to cyclones, floods, landslides, and engineered internet shutdowns—this assumption represents a single point of catastrophic failure. When the cellular network fails, the modern digital toolkit evaporates. A Kirana store owner utilizing cloud-based ledgers cannot verify credit; a gig worker relying on platform algorithms cannot receive tasks; and a family trapped in a flood cannot communicate their location or access their digital financial safety nets. This report posits that true resilience in the Global South cannot merely be an add-on to existing apps; it requires a fundamental re-architecture of how digital applications function at the edge.

Suraksha Chakra (Safety Circle) is proposed not merely as a disaster management application, but as a continuity protocol for the informal economy. By fusing hyper-local commerce with ad-hoc mesh networking and edge-based artificial intelligence, Suraksha Chakra bridges the gap between economic prosperity and survival. This document serves as an exhaustive technical and strategic feasibility report, analyzing the architectural requirements, market positioning,

and implementation pathway for a "Super Progressive Web Application" (PWA) capable of functioning in a disconnected world.

We identify three critical failure points in the current disaster response mechanism:

1. **The Asset Gap:** Inventory ruined by floodwater creates an immediate loss of net worth.
2. **The Governance Gap (The Panchnama Paralysis):** Government relief relies on physical inspection by a *Talathi*, a process taking 3–6 months.
3. **The Inclusivity Gap (The Mobility Paradox):** The elderly and *Divyangjan* (Persons with Disabilities) are often physically unable to access relief centers or ATMs even when funds are available.

Suraksha Chakra addresses these gaps through a unified "Super App" architecture that incentivizes daily usage to build a pre-disaster data layer, ensuring resilience is proactive rather than reactive.

1.1 The Vulnerability of the Informal Sector

The digitization of India's Micro, Small, and Medium Enterprises (MSMEs) has been driven by a suite of fintech applications designed to bring the informal ledger online. Platforms such as *Khatabook*, *Vyapar*, *MyBillBook*, and *Zoho Books* have revolutionized how small merchants manage credit and inventory. These applications provide immense value during "peacetime," offering features like automated payment reminders, GST-compliant invoicing, and inventory tracking. They have successfully replaced the physical "bahi khata" (ledger book) with a digital twin.

However, the architecture of these incumbent solutions is fundamentally cloud-centric. Data is synchronized to central servers to ensure consistency and enable multi-device access. While efficient, this architecture creates a dependency that becomes a liability during infrastructure disruptions. In the event of a cyclone in Odisha or a flood in Chennai, the merchant loses access to their financial history precisely when they need it most to secure emergency credit or liquidate stock. The current ecosystem lacks a "defensive" mode—a capability to revert to a localized, peer-to-peer operational state that preserves economic data and facilitates trade without an ISP.

1.2 The Limitations of Institutional Disaster Response

Parallel to the fintech ecosystem exists the government-led disaster management infrastructure. Applications like *NDMA Sachet* and *ISRO Bhuvan* represent the state-of-the-art in public warning systems. Sachet utilizes the Common Alerting Protocol (CAP) to disseminate geo-targeted warnings to citizens, integrating weather data from the India Meteorological Department (IMD). Bhuvan provides high-resolution geospatial data for damage assessment and planning.

Despite their utility, these platforms operate on a "Command and Control" model. Information flows unilaterally from the government to the citizen. They treat the user as a passive recipient of aid rather than an active agent of resilience. Critically, these applications are functionally isolated from the user's economic reality. They do not interface with the merchant's inventory to identify available food stocks during a shortage, nor do they connect with the gig worker's vehicle to organize evacuation logistics. Furthermore, they are often heavily dependent on the same cellular infrastructure that is likely to fail during a major catastrophe. The "Missing Middle" is a solution that is owned by the community, operates on distributed hardware, and treats economic continuity as a vital component of disaster survival.

2 Application Workflow & UX Design

The Suraksha Chakra web application is designed with a "State-Aware" UI that transitions between two distinct modes: **Peace Mode** (Daily Utility) and **War Mode** (Disaster Response).

2.1 User Onboarding Flow

The entry point leverages India's lightweight authentication stack to ensure accessibility for semi-literate users.

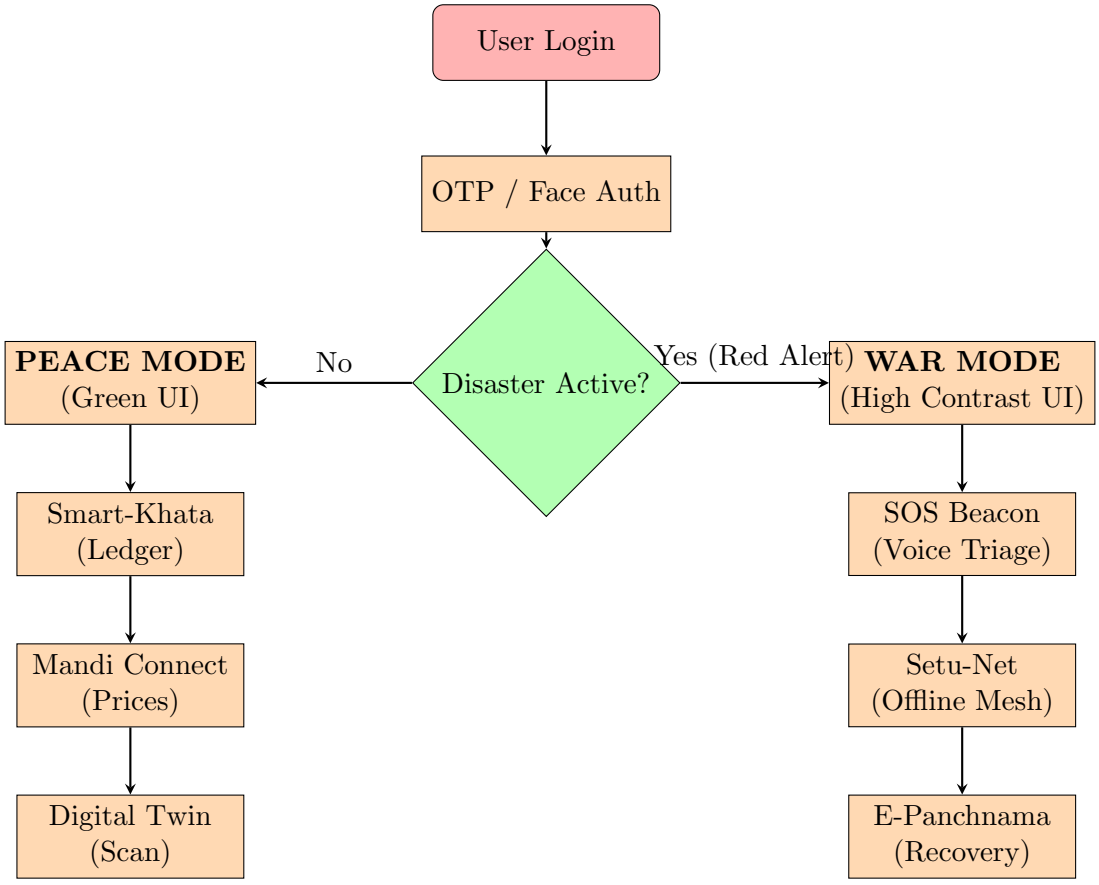

Figure 1: Application Logic Flowchart: From Login to Modular Access

2.2 UX Modules

- **Login:** Uses Firebase Phone Auth (OTP). For elderly users previously registered, we allow **FaceID** login (via FaceRD API) to bypass typing difficulties during panic.
- **Peace Dashboard:** Focuses on *Profitability*. Shows credit owed (Udhaar), wholesale prices (Mandi), and the "Suraksha Score" (Resilience Rating).
- **War Dashboard:** Focuses on *Survivability*. High contrast buttons, Voice-First interface (Bhashini), and offline-ready indicators.

3 Technical Architecture

We architect Suraksha Chakra as a Progressive Web App (PWA) wrapped in a Native Shell to access hardware features like Bluetooth and Camera.

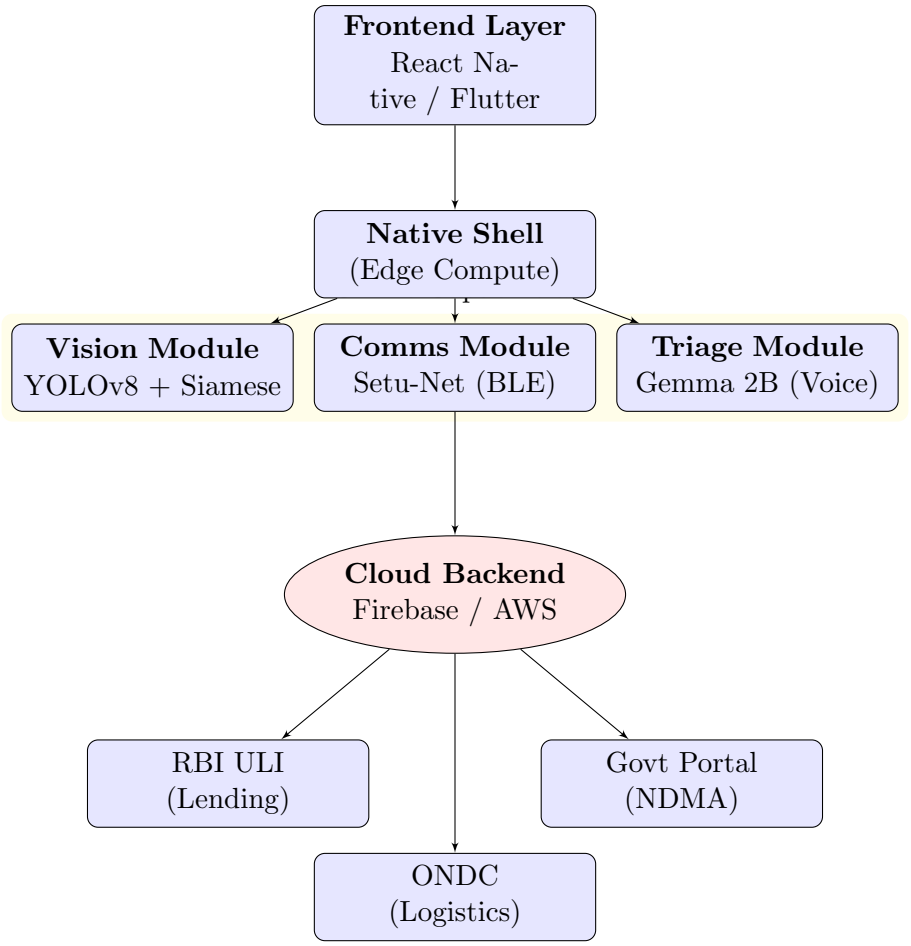

Figure 2: System Architecture Diagram

4 Methodology & Core Algorithms

Suraksha Chakra is designed as a single Progressive Web App (PWA) with a reactive "State Engine." It detects the user's context—connectivity status and disaster alerts—to switch interfaces instantly.

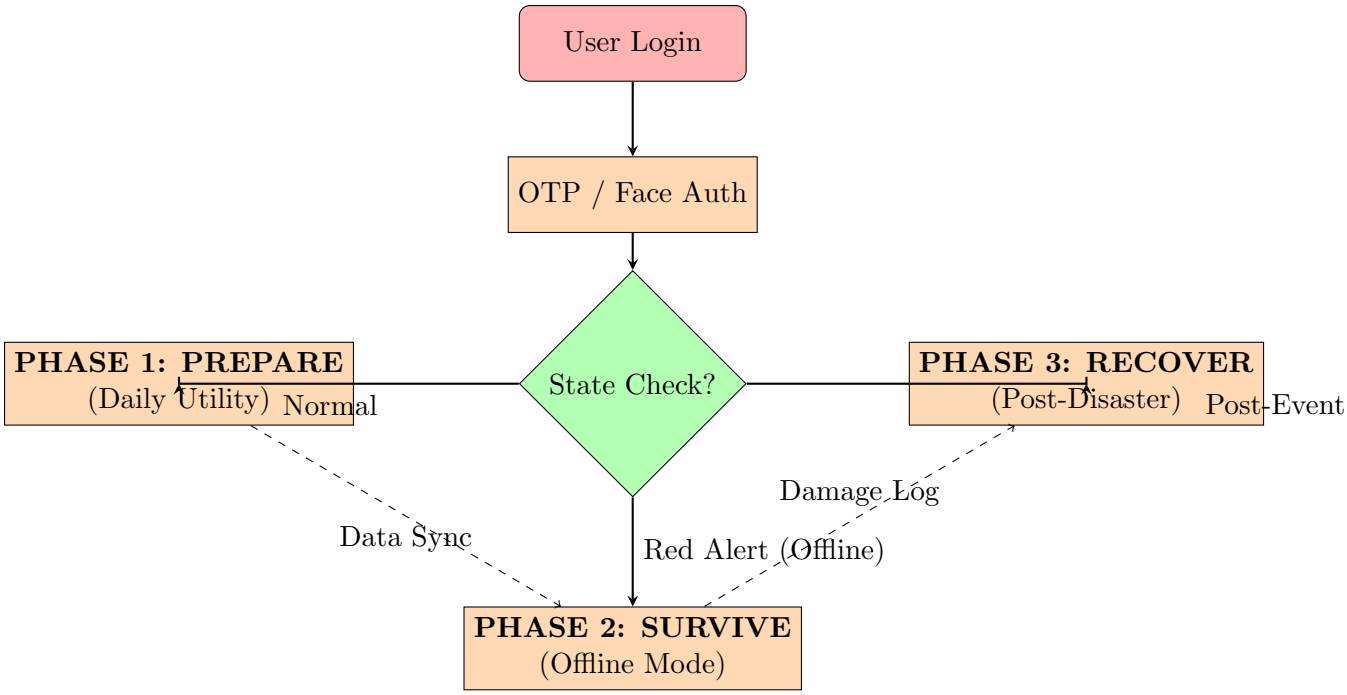

Figure 3: The Seamless State Machine: Context-Aware UI Switching

5 Phase 1: Prepare (Daily Utility - "Dukan-Sahayak")

This phase focuses on increasing the shopkeeper's income daily, ensuring high retention and data generation.

5.1 Modules

- **Smart-Khata (The Digital Ledger):** Replaces the physical red notebook. Shopkeepers record customer credit (*Udhaar*) here. This creates a **Digital Transaction Trail**, ensuring that even if the shop is washed away, the credit history remains in the cloud as proof of business solvency.
- **Mandi-Connect (Price Discovery):** Uses ONDC APIs to show daily wholesale rates. By logging purchases, the AI builds a real-time **Inventory Twin** without requiring manual insurance scans.
- **Gamified Suraksha Score:** A business health score (0–800). Higher scores unlock cheaper wholesale loans during peacetime and priority rescue during wartime.

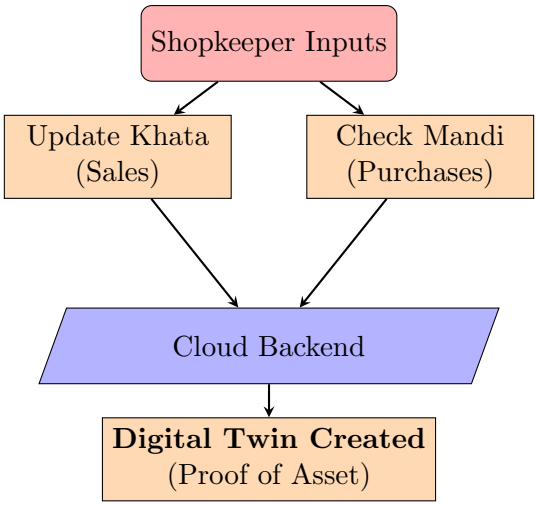

Figure 4: The "Resilience Hack": How Daily Use Builds the Digital Twin

6 Phase 2: Survive (The "Setu-Net" & "Mohalla Mesh")

When telecom towers fail, the app switches to an offline mesh network.

6.1 Setu-Net: Offline Mesh Protocol

The app uses Bluetooth Low Energy (BLE) to create a "Mohalla Mesh" (Neighborhood Network). SOS messages hop from phone to phone until they reach a functional gateway.

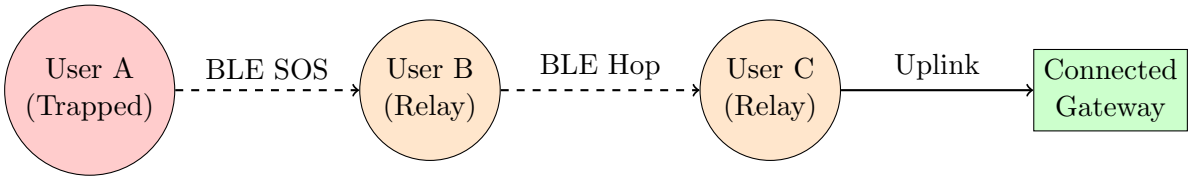

Figure 5: Setu-Net Daisy Chain Topology

Algorithm 1 Setu-Net Mesh Routing Protocol

Input: Message Content M, Time-to-Live TTL
Output: Transmission Status
$Status \leftarrow$ PENDING
if CheckConnectivity() == ONLINE **then**
    Upload M to Cloud Server
    $Status \leftarrow$ SENT
else
    $Packet \leftarrow$ {ID: UniqueID, Data: M, TTL: TTL, Hops: 0}
    $Neighbors \leftarrow$ ScanBLEDevices()
    for each $Node$ in $Neighbors$ **do**
        if $Node \notin$ VisitedList **then**
            Connect via GATT Profile
            Send $Packet$ to $Node$
            Update VisitedList
        end if
    end for
    StartListener for ACKs
    $Status \leftarrow$ FLOODED
end if
Return $Status$

6.2 Semantic Voice-Triage (The 911 for India)

An On-Device Small Language Model (SLM) listens to local dialects. It distinguishes between low priority (panic) and high priority (immediate danger) to prioritize mesh bandwidth.

Algorithm 2 Semantic Voice Triage Logic

Input: Audio Stream A (User Voice)
Output: Priority Packet P
Initialize $SLM \leftarrow$ LoadOnDeviceModel('Gemma-2B-Quantized')
$Text \leftarrow$ SpeechToText(A, LocalDialect)
$SemanticScore \leftarrow SLM$.Analyze($Text$)
$AudioEnergy \leftarrow$ FFT(A)
$Keywords \leftarrow$ ["Trapped", "Bleeding", "Neck Level"]
if $Text$ contains $Keywords$ **OR** $SemanticScore > 80$ **then**
    $Priority \leftarrow$ CRITICAL (Red)
else if $AudioEnergy >$ Threshold **then**
    $Priority \leftarrow$ HIGH (Orange)
else
    $Priority \leftarrow$ STANDARD (Green)
end if
$P \leftarrow$ ConstructMeshPacket($ID, GPS, Priority, Text$)
Broadcast P via *Setu-Net*

6.3 Khoya-Paya Protocol (Reunion Engine)

Rescue camps act as scanning nodes to reunite families separated during evacuation.

Algorithm 3 Khoya-Paya Matching Algorithm

Input: Camera Feed at Shelter (Img_{input})
Data: Encrypted Family Profiles from Cloud (DB_{faces})
$Face_{embedding} \leftarrow$ FaceNet(Img_{input})
for each $Profile$ in DB_{faces} **do**
    $Score \leftarrow$ CosineSimilarity($Face_{embedding}, Profile_{hash}$)
    if $Score > 0.85$ **then**
        $ParentContact \leftarrow Profile_{metadata}$.Phone
        Trigger WhatsApp Alert($ParentContact$, "Child Found at Shelter A")
        Break
    end if
end for

7 Phase 3: Recover (The "Phoenix Engine")

This phase automates the financial and logistical recovery.

7.1 Tatkal-Claim (The E-Panchnama)

The core technology here is a Siamese Neural Network. This AI model takes two inputs: the "Before" data (the Inventory Twin created in Phase 1) and the "After" video (recorded by the user post-disaster). It compares them to calculate the percentage of loss. It generates a legally valid "E-Panchnama" (Damage Certificate) that is tamper-proof and ready for government submission.

Algorithm 4 Siamese Network Damage Verification

Input: Video V_{pre} (Digital Twin), Video V_{post} (Disaster)
Output: DamageReport D
$F_{pre} \leftarrow$ ExtractFrames(V_{pre})
$F_{post} \leftarrow$ ExtractFrames(V_{post})
$Inventory_{pre} \leftarrow$ YOLOv8(F_{pre})
$Embeddings_{pre} \leftarrow$ SiameseNet(F_{pre})
$Embeddings_{post} \leftarrow$ SiameseNet(F_{post})
$DifferenceMap \leftarrow$ EuclideanDist($Embeddings_{pre}, Embeddings_{post}$)
$DamageScore \leftarrow$ AnalyzeMap($DifferenceMap$)
if $DamageScore > 0.75$ **then**
    $ClaimStatus \leftarrow$ AUTO_APPROVE
    $LoanAmount \leftarrow Inventory_{pre}$.Value $\times$ 0.6
else
    $ClaimStatus \leftarrow$ MANUAL_REVIEW
end if
Return $D(ClaimStatus, LoanAmount)$

7.2 The Double-Barrel Credit Relief

To restart a business, a shopkeeper needs two things: cash for immediate repairs and stock to sell. Suraksha Chakra splits the relief request into two streams:

1. **Cash Liquidity:** The E-Panchnama acts as a solvency proof, triggering the RBI's **ULI (Unified Lending Interface)**. This allows banks to disburse an instant micro-loan (e.g., 50,000) directly to the user's Jan Dhan account without a physical visit.

2. **Stock Liquidity:** The app integrates with FMCG distributors (like Unilever or ITC). Using the damage report, it issues "Inventory Credit," essentially placing a restock order that the shopkeeper can pay for later, ensuring the shop has goods to sell immediately.

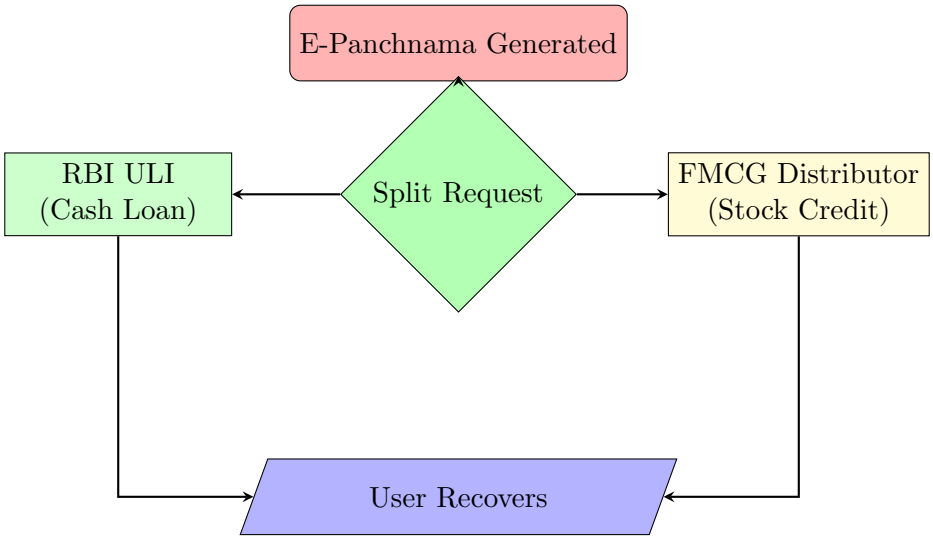

Figure 6: Double-Barrel Credit Relief Flow

7.3 Cash-on-Wheels (Bank Mitra Hailing)

In many disaster zones, ATMs are submerged or run out of cash. This feature operates like a ride-hailing app but for money. It connects the user to a mobile "Bank Mitra" (Banking Correspondent) who arrives on a bike or boat. The transaction is authenticated using AePS (Aadhaar Enabled Payment System), allowing the user to withdraw cash using just their fingerprint.

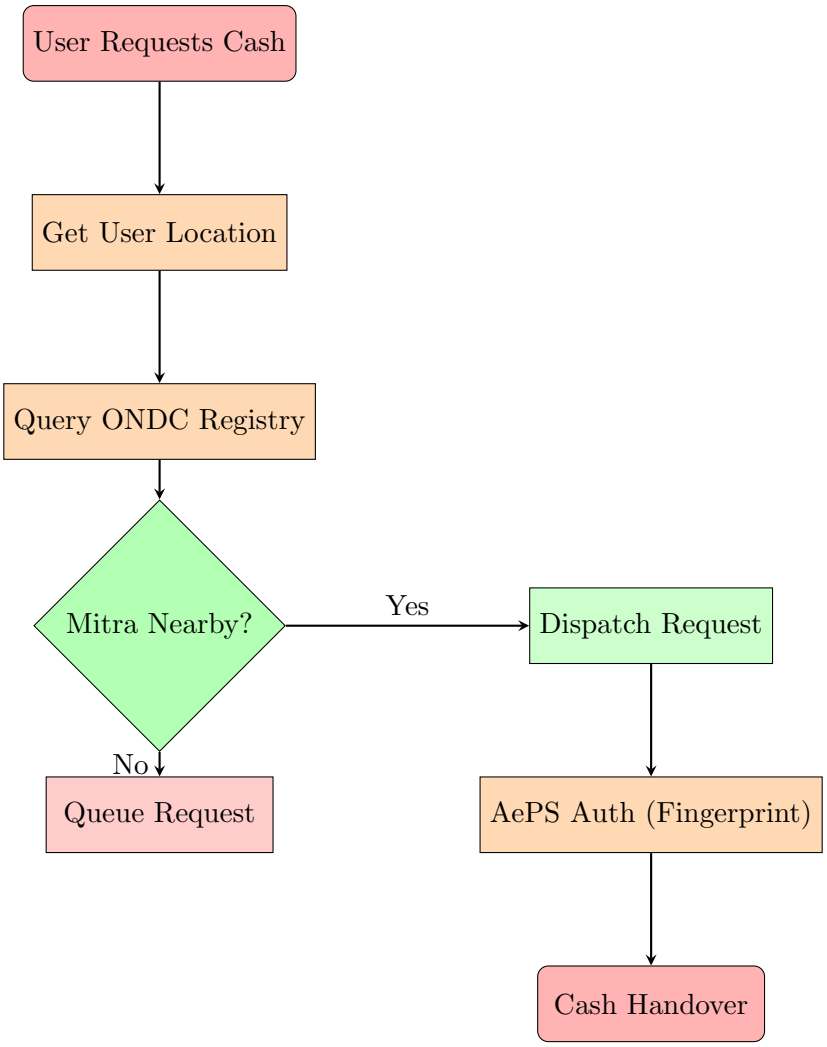

Figure 7: Cash-on-Wheels Hailing Logic

Algorithm 5 Cash-on-Wheels Dispatch Algorithm

Input: User Location L_u, Withdrawal Amount A
Output: Assigned Mitra ID M_{id}
$Registry \leftarrow$ ONDC.FetchProviders(Type = 'Banking Correspondent')
$Candidates \leftarrow []$
for each $Mitra$ in $Registry$ **do**
    $Distance \leftarrow \text{GeoDist}(L_u, Mitra.Location)$
    if $Distance < 5km$ **AND** $Mitra.CashBalance \geq A$ **then**
        $Candidates$.Add($Mitra$)
    end if
end for
if $Candidates$ is Empty **then**
    Return "No Agent Available"
else
    $Selected \leftarrow$ SortByDistance($Candidates$)[0]
    SendRequest($Selected.ID, L_u, A$)
    Return $Selected.ID$
end if

8 Conclusion

Suraksha Chakra represents a paradigm shift from "Disaster Management" to "Disaster Resilience." By creating a single web application that offers daily economic value through *Smart-Khata* and *Mandi-Connect*, we ensure high adoption rates in the informal sector. When crisis strikes, this same platform leverages Edge AI and the India Stack to automate rescue and recovery. We believe that by digitizing the informal economy's assets, we can protect the livelihoods of millions against the growing threat of climate change.

References

[1] Reserve Bank of India. *Statement on Developmental and Regulatory Policies: Unified Lending Interface (ULI)*. RBI Press Release, August 2024.

[2] Bridgefy Inc. *Mesh Networking for Disaster Relief: Case Studies from Mexico Earthquakes*. 2020.

[3] Open Network for Digital Commerce (ONDC). *Building Trust in India's Digital Commerce Ecosystem*. Government of India, 2023.

[4] National Disaster Management Authority (NDMA). *Guidelines on Disability Inclusive Disaster Risk Reduction (DiDRR)*. Government of India, 2022.