ClaimChain by team AutoMed

Team Leader : J M Tarun Team Members : Srilekha R, Aswin Raaj PS, Abhinaya

Problem Statement: Health insurance claims are often delayed, rejected, or underpaid due to manual paperwork, vague policy clauses, lack of real-time medical verification, and administrative bottlenecks. Over 60% of patients in India experience discharge delays because claim processing is either offline or partially digital, while fraud claims cost insurers thousands of crores annually. The problem is especially severe during emergencies like cardiac events, where the patient's health worsens while paperwork lags behind. Despite the availability of wearable health data from smartwatches, insurance workflows remain disconnected from real-time health insights. There is a pressing need for a secure, automated, and intelligent claim verification and processing system that is both fast and fraud-resistant.

Target Audience & Context: Our solution targets three primary groups: (1) middle-class policyholders in urban India with wearable devices, (2) gig workers and informal sector employees covered under micro-insurance platforms, and (3) rural citizens under public health schemes like Ayushman Bharat. All face the same core problem claims are delayed, untrustworthy, or unclear. With rising smartwatch adoption (over 100M users in India) and growing investments in digital health infrastructure, there's a clear opportunity to bridge real-time health data with insurance systems. We aim to bring transparency, automation, and fairness to the claim process using a GenAI-powered pipeline.

Use of Gen-AI: ClaimChain uses Generative AI in a multi-modal and multi-task pipeline to automate the insurance claim process. First, it ingests real-time sensor data (e.g., heart rate, ECG anomalies, fall detection) from smartwatches. A fine-tuned GenAI model then classifies the incident, assesses severity, and generates a human-readable claim summary. The same model (or an adjacent agent) parses the user's policy document using language reasoning and constraint extraction to determine eligibility, exclusions, co-pay limits, etc. Simultaneously, GenAI references local hospital pricing data and adjusts the recommended payout using contextual cues (e.g., policy coverage, recent activity logs, device confidence). The claim packet—complete with justification, timestamp, policy mapping, and payout calculation—is then formatted and passed to a blockchain smart contract for autonomous execution. GenAI ensures interpretability, personalization, and compliance—making the system scalable, fair, and auditable.

Solution Framework: ClaimChain is a real-time, AI-powered insurance claim automation framework. It integrates three technologies:

1. Wearable IoT Devices (e.g., Noise smartwatches):
    - Collect continuous health metrics: HR, ECG, SpO_2, steps, falls
    - Synced with smartphone apps and cloud (via Google Fit API or custom BLE interface)
2. GenAI Claim Reasoning Engine:
    - Event Detection: Triggered by anomaly (e.g., sudden drop in heart rate + fall)
    - Claim Summary Generation: LLM converts raw data into natural language medical event report
    - Policy Parser: Reads and interprets user's insurance policy using NLP and semantic similarity
    - Payout Estimator: Uses local hospital pricing + policy constraints to calculate the claim amount
    - All logic is handled by orchestrated LLM agents (via LangChain, OpenAI function calling or fine-tuned open-source LLMs)

3. Smart Contract on Blockchain:
    - Smart contract deployed on Ethereum L2 (e.g., Polygon) validates:
        - Claim legitimacy
        - Coverage match
        - Dynamic payout computation
    - Executes payout to hospital or user wallet
    - All transactions are immutable and auditable

Diagram: **[Smartwatch → GenAI Summary + Policy Parsing → Claim Packet → Blockchain Smart Contract → Payout]**
This architecture eliminates paperwork, accelerates approvals, deters fraud, and empowers trust.

Feasibility and Execution:
The system can be implemented using existing tools:

- Hardware: Budget smartwatches (e.g., Noise, boAt) under ₹1000
- Data Ingestion: Google Fit API or BLE + SQLite-based edge syncing
- GenAI Layer: GPT-4 (or LLaMA 3) with LangChain for policy reasoning and summary generation
- Hospital Pricing: Public or scraped APIs or static datasets
- Blockchain: Solidity smart contract deployed on Polygon testnet
  Initial deployment can focus on one condition (e.g., cardiac events) for a focused PoC. With limited users, we can simulate or ingest actual wearable data and run a closed-loop test.

Scalability & Impact: The solution is highly scalable across both B2B (insurance companies) and B2G (public health schemes). It can be deployed in:

- Urban settings via employer or retail insurance partners
- Rural areas through Ayushman Bharat using subsidized wearables
- Micro-insurance platforms serving gig workers or daily wage earners

With adoption, ClaimChain could reduce average claim approval time by 90%, cut fraud by 60–70%, and improve user trust. It can support future expansion into chronic condition monitoring, preventive care rewards, and cross-border portability of insurance claims via verifiable health events on-chain.

Conclusion & Minimum Lovable Product: ClaimChain is an intelligent, real-time claim automation platform that bridges the gap between real-world health events and insurance trust. Its unique combination of wearable data, GenAI reasoning, and smart contracts eliminates delays, fraud, and opacity. Even a minimal version with heart attack detection, GenAI summary, and payout simulation can demonstrate its viability and spark interest from insurers, OEMs, or digital health partners.

Workflow: 48-Hour Execution Plan
To demonstrate the core functionality of ClaimChain within 48 hours, we'll build a focused, testable prototype centered around **heart-related emergencies**. Our goal is to simulate the full pipeline: from smartwatch data → GenAI reasoning → blockchain claim execution.

Phase 1: Setup & Data Simulation (Hours 0–6)

- Collect or simulate **Noise smartwatch data** (heart rate spikes, ECG anomalies, falls)
- Use **Google Fit API**, or upload **CSV logs** mimicking real-time sensor feeds
- Pre-load a few anonymized **insurance policy documents (PDF or text)**
- Prepare a **static hospital pricing table** (CSV format)

Phase 2: GenAI Pipeline (Hours 6–20)

- Use **GPT-4/Gemini Ai via LangChain** or **LLaMA 3** to:
  - Parse smartwatch data and **detect health incidents**
  - Generate a **natural language claim summary**
  - Parse user insurance policy and **extract eligibility & exclusions**
  - Calculate **estimated payout** using hospital price and policy coverage

Phase 3: Smart Contract Development (Hours 20–30)

- Write a **Solidity smart contract** on **Polygon Mumbai Testnet** to:
  - Validate input fields from the Claim Packet
  - Check limits, exclusions, and matching event type
  - Emit an **approval event** or simulate a payout transfer to wallet address
- Deploy using **Remix IDE** or **Hardhat**

Phase 4: End-to-End Integration (Hours 30–40)

- Build a **simple front-end (React or Streamlit)** to:
  - Upload simulated health data
  - Trigger GenAI pipeline
  - Preview AI-generated claim summary
  - Push to blockchain and show transaction hash / payout simulation