**Project Document**

**Title:** AI-Driven Fraud Detection & Alert System for Retail Investors

**1. Introduction**

India’s securities market has seen massive participation from **10+ crore retail investors**. While this democratizes wealth creation, it also exposes small investors to risks such as:

* Pump-and-dump schemes
* Insider trading leaks
* Circular trading & operator plays
* Fake news or rumor-based manipulation

These practices **erode trust** and **cause financial losses** for retail investors.

Our solution: **AI-driven Fraud Detection & Alert System** that monitors market + news data, detects suspicious activity, and provides **real-time risk alerts** with clear explanations.

This aligns with **SEBI’s mandate** of protecting investors and ensuring transparency.

**2. Problem Statement**

* Retail investors lack **tools to detect fraud/manipulation** in real time.
* Current systems (broker apps, news) only show prices but don’t flag risks.
* False rumors on social media/news spread fast, retail investors fall prey.
* No existing **scalable, AI-powered, explainable fraud alerting system** for small investors.

**3. Objectives**

1. **Detect fraud signals** from market data & news.
2. **Alert investors in real-time** about high-risk stocks.
3. Provide **explainable reasons** (why a stock was flagged).
4. Design a system that can be **scaled & integrated** into broker apps.
5. Ensure **compliance + cybersecurity** (no PII, model governance).

**4. System Overview**

**🔹 Core Idea**

* Ingest stock market + news data.
* Extract suspicious patterns (volume spikes, circuit hits, rumor headlines).
* Use **ML models + rule-based signals** to generate risk scores.
* Deliver **alerts via dashboard/email/push**.
* Provide **transparent explanation** for each alert.

**5. Features**

**✅ Basic Features (MVP)**

* Market anomaly detection (volume/price spikes, abnormal volatility).
* Circuit breaker hit frequency check.
* Sentiment analysis of financial news.
* Composite Risk Score (Low/Medium/High).
* Streamlit dashboard with stock risk levels.
* Email alert system.

**🚀 Advanced Features (Stretch Goals)**

* Order-book anomaly detection (spoofing, cancel-replace patterns).
* Graph-based suspicious trader clustering.
* Social media sentiment integration.
* What-if simulator (expected loss if investor bought risky stock).
* Broker API integration (Zerodha, Groww).

**6. System Architecture**

**🔹 Components**

1. **Data Ingestion**
   * Market Data: OHLCV (Open, High, Low, Close, Volume).
   * News Data: Curated headlines (labeled manually + scraped).
   * [Stretch] Order-book snapshots (synthetic for demo).
2. **Feature Engineering**
   * Z-score Volume Spike.
   * Price Gap (Open vs Previous Close).
   * Intraday Volatility (True Range).
   * Circuit breaker frequency.
   * News Sentiment Score.
3. **Model Layer**
   * **Isolation Forest** (unsupervised anomaly detection).
   * **XGBoost** (supervised anomaly scoring with weak labels).
   * **DistilBERT / MiniLM** for news sentiment classification.
4. **Backend (API + Processing)**
   * FastAPI for model scoring.
   * PostgreSQL + TimescaleDB for time-series data.
   * Redis for caching latest risk scores.
5. **Frontend (UI)**
   * Streamlit Dashboard (fast hackathon prototype).
   * Stock table (name, price, risk score, explanation).
   * Trend charts + news headlines.
   * Alert log.
6. **Alerts**
   * Email/SMS/Webhook notifications.
   * Push notifications (stretch).
7. **Explainability Layer**
   * SHAP values / feature importance.
   * Human-readable explanation:

“Stock flagged due to 3× volume spike + 2 negative news items in 24h.”

**7. Tech Stack**

* **Language:** Python 3.11
* **Libraries (ML):** Pandas, Scikit-learn, XGBoost, Hugging Face Transformers
* **Database:** PostgreSQL + TimescaleDB
* **Backend:** FastAPI (Dockerized)
* **Frontend:** Streamlit (demo) / React (production)
* **Visualization:** Plotly, ECharts
* **Alerts:** SMTP for email, Firebase for push (stretch)
* **MLOps:** MLflow for versioning, Docker for deployment

**8. Development Plan**

**🕒 Week 1 (Hackathon Timeline)**

* **Day 1–2:**
  + Collect OHLCV + news dataset.
  + Implement feature engineering (volume z-score, volatility, sentiment).
* **Day 3–4:**
  + Train baseline ML models (Isolation Forest + XGBoost).
  + Create weak labels for fraud patterns.
* **Day 5:**
  + Build API (FastAPI) for model scoring.
  + Connect PostgreSQL database.
* **Day 6:**
  + Build Streamlit dashboard (charts + risk table).
  + Integrate alerts (email).
* **Day 7:**
  + Add explainability (SHAP + human-readable rules).
  + Final polish + record demo video.

**9. Metrics for Evaluation**

* **Precision @ High-Risk Alerts** (don’t spam users).
* **Latency:** < 60 seconds from event → alert.
* **Coverage:** # of stocks monitored.
* **Explainability:** % alerts with clear reason.
* **Scalability:** Ability to handle 500+ stocks.

**10. Cybersecurity & Compliance**

* No PII collected (investor safe).
* Data provenance: timestamped, verified sources.
* Model governance: versioning + drift monitoring.
* Legal disclaimer: “Advisory tool, not investment advice.”

**11. Impact**

* Protects **10 crore retail investors** from scams.
* Reduces **losses from manipulative schemes**.
* Enhances **trust in Indian securities market**.
* Scalable to broker apps, AMC dashboards, regulatory supervision.

**12. Future Enhancements**

* Full social media monitoring (Twitter, Telegram groups).
* Deep Learning models (transformers for fraud pattern detection).
* Regulatory dashboard for SEBI → track manipulative operators.
* Integration with **CBDC/UPI 2.0** for secure trading ecosystem.

**13. Team Roles**

* **ML Lead** → Data ingestion, feature engineering, fraud detection models.
* **Backend Lead** → API, DB, alerting system, Dockerization.
* **UI/UX + Pitch Lead** → Dashboard, charts, storytelling, pitch deck.
* [Optional] Finance mentor for domain validation.

**14. Pitch Lines (for Judges)**

* “We detect manipulative market patterns **before they hurt retail investors**.”
* “Our alerts are **explainable, precise, and real-time** — built for scale.”
* “Designed for **broker integration & regulator dashboards**, ensuring transparency and investor safety.”

**15. Conclusion**

This project is **not just a hackathon demo**, but a **scalable blueprint** for India’s retail investor safety ecosystem. By combining **AI, ML, NLP, and scalable architecture**, it offers a **practical, regulator-aligned fraud detection system** that can make India’s markets safer and more transparent.

⚡ Done — this is the **complete project document** you can:

* Submit in hackathon registration,
* Use as development guide,
* Pitch to judges.

Do you want me to also prepare a **1-page executive summary (with problem → solution → impact)** that you can copy-paste into the **hackathon submission form**?

Top of Form

Bottom of Form