Here is the updated Product Requirements Document (PRD). I have incorporated all the feedback points (Feedback Loop, Pagination, Configuration, Architecture, and Terminology) and marked the specific additions with **[Gemini]** so you can easily spot the enhancements.

Product Requirements Document (PRD)

Fraud Detection & Prevention Platform for Telecom & MSP

Version: 1.1 [Gemini: Updated]
Date: December 2025
Author: [Your Name]
Project Type: Capstone / Portfolio Project
Target Audience: Hiring managers, product/technical leaders at telcos, MSPs, and tech companies

---

Executive Summary

This PRD outlines the development of a **fraud analytics and detection prototype** for the Telecom and Managed Service Provider (MSP) domains. The prototype is a **portfolio project** designed to demonstrate end-to-end product and technical execution, combining problem framing, solution design, clean engineering, observability, and governance thinking—all critical capabilities for product management and technical leadership roles.
Phase 1 (8–10 weeks) will deliver a working system detecting two priority fraud types: **MSP contract/invoice anomalies** and **identity/account-based fraud**, with a clear architectural foundation for scaling to Phase 2 (call-based fraud) and Phase 3 (cross-industry collaboration).

---

1. Project Overview

1.1 What We Are Building

An **end-to-end fraud detection and analytics system** that:

- **Ingests diverse data batches [Gemini: Refined terminology from "streams" to "batches" to match Phase 1 architecture]** mimicking real operational contexts: MSP invoice/contract records, subscriber account activity, identity/credential events, and billing anomalies.

- **Detects fraud patterns and anomalies** using rule-based and simple ML approaches, producing risk scores with clear explanations for human analysts.
- **Provides analyst workflows** with dashboards, alerts, case management, and KPI tracking to show how fraud operations teams would adopt and use this system.
- **Emphasizes operational realism**: batch/micro-batch processing (near real-time ready for Phase 1; true inline blocking deferred to Phase 2), practical governance, compliance awareness, and clear documentation of limitations.

[Gemini: Added Visual Placeholder]
System Architecture Diagram
(Placeholder: A high-level diagram illustrating the flow: Synthetic Data Generators -> Ingest API -> Detection Engine (Rules/Analytics) -> PostgreSQL DB -> FastAPI -> React UI)

1.2 Why This Project Matters

For Hiring Managers & Technical Leaders:
This project demonstrates:

- **Problem-space expertise**: Understanding of telecom/MSP fraud types, business impact, regulatory context, and cross-stakeholder dynamics (telcos, banks, regulators).
- **Product thinking**: Clear personas, use cases, success metrics, documented constraints, and awareness of trade-offs (false positives vs. operational burden).
- **Technical execution**: Clean, typed, well-tested code; observability; secure by design; reproducible deployments; and honest documentation of edge cases and scope limitations.
- **Portfolio-grade delivery**: A public GitHub repo, working prototype on a real platform, >80% test coverage, clear architecture, and evidence of iterative refinement.

For the Fraud/Risk Community:
This project provides a reusable template and reference implementation for fraud detection in telco/MSP contexts, showcasing best practices in:

- Synthetic data generation for fraud scenarios.
- Rule-based + analytical detection pipelines.
- Analyst-friendly UI and alerting workflows.
- Operational KPIs and process design.

1.3 Users & Personas (Real-World Framing)

Primary Users (Operational):

1. **Fraud Analyst (Telco/MSP)** – Reviews alerts, investigates cases, provides disposition; uses risk scores and related-entity views to triage and decide on action.
2. **Operations/Network Engineer** – Monitors CDR data, billing anomalies, and contract fulfillment; needs anomaly detection to catch operational or fraudulent issues early.
3. **Fraud Manager/Risk Officer** – Sets policies, tunes thresholds, reviews KPIs, and ensures compliance with regulatory expectations.

Secondary Users (Strategic):

- **Compliance / InfoSec Officer** – Understands how fraud detection interacts with data privacy, identity management, and joint liability frameworks (e.g., GDPR, PSD3).

- **Business/Revenue Teams** – Needs visibility into fraud losses, false positive costs, and conversion impact of friction added by fraud controls.

Your Actual Audience (Why You're Building This):

- Hiring managers, product leads, and senior engineers at Google, Apple, high-growth fintech/telco startups evaluating your **product sense, technical depth, and execution discipline**.

2. Business Context & Problem Statement

2.1 Industry Landscape

Telecom and MSP fraud is a **multi-billion-dollar problem** globally. Fraud takes many forms:

Telecom Fraud Types (per LexisNexis and industry sources):

- **Subscription Fraud**: Criminals use stolen identities to apply for services; telcos bear loss.
- **First-Party Fraud**: Genuine account holder intentionally defaults or files chargebacks.
- **Synthetic Identity Fraud**: Combination of real + fabricated data to open fake accounts.
- **Account Takeover (ATO)**: Fraudsters gain control of genuine accounts via compromised credentials, change details, make unauthorized transactions.
- **Credential Testing**: Attackers test stolen credentials against accounts (often from dark web data breaches) to confirm access.
- **Bot Attacks**: Automated attacks on login/signup; bot attacks at login rose **597% globally** (2022, LexisNexis data).

MSP-Specific Fraud:

- **Invoice/Contract Anomalies**: Duplicate invoices, overlapping service periods, inflated labor rates, ghost projects, or time-entry fraud.
- **Labor Time Fraud**: Overbilling for services not rendered or falsifying billable hours.
- **Subscription Abuse**: Free trial abuse, rapid account cycling, unauthorized resale of access.

2.2 Current State & Gaps

Today, most telcos and MSPs:

- Rely on **manual review** and legacy rule engines (often inflexible, hard to update).
- Lack **real-time behavioral context** and cross-source correlation.
- Struggle to **balance fraud prevention** with customer experience (high false positives → operational burden and customer churn).
- Have **unclear responsibility chains** when fraud crosses organizational boundaries (e.g., a telco and a bank both impacted by APP scams).
- Face **growing regulatory pressure** to demonstrate proactive fraud detection and coordinated industry response.

2.3 Our Approach: Phase-Based Evolution

Phase 1 (Capstone MVP): Build a polished, documented prototype that proves:

- Ability to model fraud scenarios in synthetic data.
- Effective rule + analytics-based detection for 2 priority fraud types.
- Operational workflows (alert triage, case management, KPI tracking).
- High engineering standards (test coverage, observability, clear documentation).

Phase 2 (Roadmap): Add ML depth, more fraud types, richer UX, and deeper analytics.
Phase 3 (Vision): Enable cross-industry fraud intelligence sharing (inspired by GSMA "Scam Signal" and PwC joint-responsibility frameworks).

---

3. Core Requirements (Prioritized)

3.1 Must-Have (Phase 1 MVP)

3.1.1 Synthetic Data & Scenarios

- [ ] **MSP Contract/Invoice Dataset**
  - Schema: Contract ID, MSP ID, Service Type, Quantity, Unit Cost, Invoice Date, Paid Date, Time Entry Records, Labor Hours, Billing Rate.
  - Normal behavior: Consistent billing, realistic invoice cycles, aligned time entries.
  - Fraud scenarios:
    - **Duplicate invoices**: Same contract, same dates, multiple invoice records.
    - **Overlapping periods**: Time entries covering same hours by multiple contractors.
    - **Rate anomalies**: Sudden spikes in hourly rates or bulk discounts.
    - **Ghost projects**: Invoices with no corresponding time entries or contracts.
  - **Target scale**: 50k–100k invoice records, 5–10% flagged as anomalous.
- [ ] **Identity/Account-Based Dataset**
  - Schema: Subscriber ID, Account Creation Date, Device ID, IP Address, Email, Phone Number, Login Events, Credential Change Events, Payment Methods, Account Status.
  - Normal behavior: Stable login patterns, periodic credential updates, consistent device/IP.
  - Fraud scenarios:
    - **Credential testing**: Multiple failed logins from different IPs followed by success.
    - **Account takeover (ATO)**: Abrupt device/IP changes, rapid credential resets, suspicious payment method additions.
    - **Synthetic identity**: New account with rare device/email combinations, unusual behavioral patterns.
    - **SIM swap indicators** (Phase 2): Phone number changes without

    corresponding device/account reconciliation.
  - **Target scale**: 100k–200k subscriber records with 5–8% flagged anomalies.
- [ ] **Synthetic Data Generation** (Python / Pandas)
  - Realistic distributions: Call durations, invoice amounts, login frequency (daily, weekly patterns).
  - Fraud-scenario generators: Separate logic for creating each fraud type to avoid overfitting detection to generation.
  - **Noise injection**: Random variations (missing fields, delays, data quality issues) to prevent toy-like perfection.
  - **Labeled data**: Clear ground truth for testing and measuring detection accuracy.

3.1.2 Detection Engine

- [ ] **Rule-Based Anomaly Detection**
  - **MSP Fraud Rules**:
    - Duplicate invoice detection.
    - Overlapping time-entry detection.
    - Rate anomaly detection.
    - Ghost project detection.
  - **Identity Fraud Rules**:
    - Credential testing pattern.
    - Device/IP change detection.
    - Rapid credential change.
    - Payment method churn.
  - **[Gemini: Added] Configuration Management**:
    - Fraud thresholds (e.g., "7 days", ">2σ", "3 attempts") must not be hardcoded.
    - Extract all magic numbers to a config.yaml or .env file to demonstrate maintainability and ease of tuning without code redeploys.
- [ ] **Analytical Anomaly Detection**
  - **Z-score / percentile-based outlier detection** on numeric features (invoice amount, contractor rate, login frequency).
  - **Rolling window aggregation**: Detect unusual velocity (e.g., 10x normal invoice volume in a week).
  - **Entity-level features**: Account age, historical fraud indicators, related entity risk (e.g., if a contractor is flagged, related contracts at higher risk).
- [ ] **Risk Scoring & Explanation**
  - Composite risk score per entity (0–100 scale).
  - **Explainability**: List contributing factors with weights.
  - **Severity tiers**: Green (<30), Yellow (30–70), Red (>70).

3.1.3 API & Backend Services

- [ ] **Data Ingestion API**
  - POST /api/v1/ingest/contracts & POST /api/v1/ingest/accounts.

    - **Idempotent processing**.
    - **Validation & error handling**.
- [ ] **Alert API**
    - GET /api/v1/alerts – List alerts with filters.
    - **[Gemini: Added] Pagination Support**: Must support server-side pagination (limit/offset) to handle high volumes (e.g., >10k alerts) without crashing the UI or API.
    - GET /api/v1/alerts/{alert_id} – Detail view.
    - PATCH /api/v1/alerts/{alert_id} – Update status and disposition.
- [ ] **Risk Score API**
    - GET /api/v1/risk-score/{entity_type}/{entity_id}.
    - **Response time**: p95 <300 ms.

3.1.4 Analyst Dashboard (Web UI)

- [ ] **Alert List View**
    - Table of recent alerts: timestamp, fraud type, entity, risk score, status.
    - **[Gemini: Added] Pagination UI**: Implementation of "Next/Previous" or infinite scroll backed by the paginated API, ensuring scalability for large datasets.
    - Filters: date range, fraud type, risk tier, status.
- [ ] **Alert Detail View**
    - Risk score breakdown, entity profile, related alerts, timeline view.
    - Analyst workflow: buttons to disposition.
- [ ] **Trend Dashboard**
    - Charts: alerts per day, distribution by risk tier, analyst throughput.

3.1.5 Alerting & Workflow (MVP)

- [ ] **Alert Queue / Case Management**
    - Simple database table: alert ID, entity details, risk score, status, disposition, notes.
    - **[Gemini: Added] The Data Flywheel (Ground Truth Feedback)**:
        - Explicitly capture the analyst's disposition (True Positive vs. False Positive) in a structured format suitable for retraining future ML models.
        - This creates a "labelled dataset" from operational usage, a critical feature for AI product maturity.
- [ ] **Notification Stub**
    - Optional webhook/email integration stub.

3.1.6 Engineering Excellence

- [ ] **Test Coverage**
    - 80% coverage for backend logic.
- [ ] **Code Quality**
    - Type hints, Docstrings, Linting (isort, black, flake8).
    - **No TODOs** in core logic.
- [ ] **Edge Case Documentation**

    - Handled vs. unhandled edge cases clearly listed in README.
- [ ] **GitHub Repository**
    - Public repo with clear structure, CI/CD pipeline, ADRs, and README.

3.1.7 Deployment & Observability

- [ ] **Containerization**
    - Docker images, docker-compose.yml, deployment to Railway.io.
- [ ] **Basic Observability**
    - Structured logging, Metrics (Prometheus format), Health checks, Dashboard.
- [ ] **Secrets & Configuration**
    - Environment variables, no hardcoded secrets.

3.2 Should-Have (Phase 2)

- [ ] **Enhanced ML/Analytics** (Unsupervised anomaly detection, Feature store).
- [ ] **Multi-Channel Coverage** (Call/SMS event data).
- [ ] **Richer UX & Configuration** (Configurable rules via UI).

3.3 Nice-to-Have (Phase 3 & Future)

- [ ] **Cross-Industry Collaboration & "Fraud Signal API"**.
- [ ] **Advanced Analytics & Modeling** (Graph-based detection, Explainable AI).

---

4. Non-Functional Requirements

4.1 Performance

Phase 1 Targets (batch/micro-batch processing):

- **Data Processing**:
    - Ingest 100k synthetic records in <60 seconds.
    - Rule engine latency: <1 second per batch.
- **API Response Times**:
    - GET /alerts: p95 <500 ms for typical queries.
    - **[Gemini: Added] Pagination**: Response time must remain <500ms even as total record count grows to 1M+ (via efficient LIMIT/OFFSET queries).
    - GET /risk-score: p95 <300 ms.

4.2 Reliability & Error Handling

- Graceful Degradation, Data Consistency (ACID), Idempotency.

4.3 Code Quality

- Type Hints, Docstrings, Test Coverage >80%, Linting.

- **[Gemini: Added] Configuration**: All business logic thresholds (time windows, risk weights) extracted to config files, not hardcoded.

4.4 Security & Compliance (Prototype-Level)

- No real PII (Synthetic only), Basic Auth, Compliance Narrative (GDPR/PSD3 awareness).

4.5 Observability & Monitoring

- Structured Logging, Metrics (throughput, latency, error rate), Health Checks.

---

5. Technical Stack

(Unchanged from v1.0 - Python/FastAPI, PostgreSQL, React, Docker, Railway.io)

---

6. Data Model & Synthetic Scenarios

6.1 MSP Fraud: Contract/Invoice Schema

(Standard schemas for MSPs, Contractors, Contracts, Invoices, TimeEntries)

6.2 Identity Fraud: Account Schema

(Standard schemas for Subscribers, Accounts, Devices, LoginEvents, PaymentMethods)

6.3 Alerting & Feedback Schema [Gemini: Added Section]

[Gemini: Rationale]: This schema extension captures the "Ground Truth" needed to retrain models in Phase 2.

```
Alerts
├── alert_id (PK)
├── entity_type (contractor, account)
├── entity_id
├── risk_score
├── risk_tier (low, medium, high)
├── created_at
└── status (new, investigating, closed)

CaseDisposition
├── case_id (PK)
├── alert_id (FK)
├── analyst_id
├── disposition (true_positive, false_positive, benign_anomaly)
├── reason_code (suspected_fraud, data_error, known_pattern)
├── notes
└── timestamp
```

6.4 Synthetic Data Generation Strategy

(Unchanged: Realistic distributions, fraud injection, noise, separation of concerns)

---

7. Development Phases & Timeline

(Unchanged: 8–10 weeks plan)

---

8. Key Performance Indicators (KPIs)

8.1 Technical KPIs

(Unchanged)

8.2 Operational KPIs

(Unchanged)

8.3 Business KPIs

(Unchanged)

---

9. Technical Constraints & Edge Cases

9.1 Handled in Phase 1

- **[Gemini: Added] Large Result Sets**: Handled via API pagination; UI does not crash on high alert volume.
- Missing fields, duplicate events, clock skew, DB failures, malformed JSON, Type safety, Test isolation.

9.2 Explicitly Out of Scope (Phase 1)

(Unchanged)

---

10. Governance, Compliance & Documentation

(Unchanged)

11. Risk Mitigation

(Unchanged)

12. Success Metrics Summary

(Unchanged)

Document Version History

Version	Date	Author	Changes
1.0	Dec 2025	[Your Name]	Initial PRD; Phase 1 scope.
1.1	Dec 2025	[Your Name]	**[Gemini]** Added Feedback Loop schema, Pagination requirements, Configuration extraction, and Architecture visual placeholder.