Fall 2025

Software Requirements Specification

SENTRA

Ryaan Farrukh

Kencho Lodhen

Anand Krishna Anil Kumar

Jasmine

Github:

https://github.com/rfarrukh0/Sentra

Executive Summary

0.1 Background

The proposal introduces a centralized fraud detection platform designed to empower smaller fintech firms and startups to collectively combat rising fraud and scams in the digital finance sector, with features for collaborative intelligence-sharing, machine learning-based detection, and secure integration. Fraud losses in the U.S. financial sector reach as much as $20 billion annually, with smaller fintech firms and startups disproportionately affected due to lack of advanced tools. Isolated systems allow fraudsters to repeat their methods across companies, making collaboration essential.

0.2 Description

0.3 Company Value Add

Participating companies gain enhanced fraud detection capabilities without the need to individually invest in advanced infrastructure. The system also reduces operational and reputational risk by blocking fraudsters more quickly and efficiently.

0.4 End-User Value Add

Customers experience a reduced risk of financial loss and identity theft, as well as higher overall confidence in digital banking platforms, thanks to enhanced fraud prevention and industry-wide protections.

0.5 Scope

0.5.1 What is Included

- Collaborative fraud intelligence sharing
- Real-time monitoring and alerting
- Machine learning-powered anomaly detection
- Secure APIs for onboarding fintech partners
- Web dashboard for reporting and visualization

0.5.2 What is Not Included

- Direct access to raw customer data between firms
- Out-of-industry use cases (e.g., non-financial industries)
- Legacy integration with non-digital banking tools
- On-premises deployment (cloud-based only)

0.6 Justification

The two-semester scope is justified by the project's complexity: integrating diverse data sources, designing robust ML and API solutions, ensuring security and privacy compliance, and creating operational front/back-end components. The platform delivers significant value by reducing fraud, safeguarding customers, and elevating industry standards for trust and security.

Table of Contents

Section 1

1.1 Document Authors

Ryaan Farrukh
Kencho Lodhen
Anand Krishna Anil Kumar
Jasmine

1.2 Document Revision History

WEEK	DATE	Revisions
1	September 10 - 11, 2025	• Made the business proposal
2	September 17 – 18, 2025	• Entire SRS section 1 and section 2.1
3	Sept 24, 2025	• SRS Section 2.2-2.5 draft + revision of previous sections, Finished up SRS section 2
4	Oct 1, 2025	• Began function requirements and NFR draft
5	Oct 4, 2025	• Finished sections 2.6, and 2.7
6	Oct 9, 2025	• Revised the non-functional requirements
7	Oct 11,2025	• Made the UI draft
8	Oct 12, 2025	• Completed section 2.8
9	Oct 29, 2025	• Discussion on section 3
11	Oct 30, 2025	• Completed the diagrams
12		•
13		•
14		•

1.3 Document Purpose

This Software Requirements Specification (SRS) defines the purpose, scope and objectives of Sentra. Sentra is a centralized fraud detection platform, and this document provides a shared reference for the team, instructors and shareholders guiding development over PRJ566 + PRJ666

1.4 Audience

The audience includes:

- Course Instructor
- Team Members
- Industry Stakeholders

1.5 Group Agreement

Team #: Group 7

Project Title: Sentra – Centralized Fraud Detection Platform

Project Time Frame: September 2025 - April 2026 2 Semesters (8 months)

Team Members

Ryaan Farrukh
Kencho Lodhen
Anand Krishna Anil Kumar

Team Leadership

Ryaan Farrukh (Project Lead)

Team Functions/Roles

Ryaan Farrukh – Project Manager and Machine Learning developer

Kencho Lodhen – Backend development and API integration

Anand Krishna Anil Kumar – UI + UX Designer and Documentation

Team Meetings

Weekly online via MS Teams (Saturday 3pm) + bi-weekly in-class check-ins.

Team Problems

Limited team size increases workload between members.

Team Commitment

The undersigned members agree to work together on the project until the end of the PRJ666 next Semester. They recognize that as a team and individually they are equally responsible for the quality of all deliverables.

Name	Date	Signature
Ryaan Farrukh	September 18, 2025	R.F
Kencho Lodhen	September 18, 2025	K.L
Anand Krishna Anil Kumar	September 18, 2025	A.K

Section 2

2.1 Project Proposal

2.1.1 Project Background

As financial services are rapidly digitized, such rapid digitization opens a broader attack surface for fraud. In 2024, consumers reported $12.5B worth of fraud losses to the U.S. Federal Trade Commission, which is up 25% YoY-and the FBI's IC3 has recorded more than $16 billion in cybercrime-related losses [1][2]; this figure is the highest ever reported. Such figures are likely to be underestimations because they are based on voluntary reporting.

Reports within the financial service industry have also shown increases, for example, 1 in 20 verification attempts are flagged as fake and a 21% rise in fraud attempts [3][4]. This highlights just how quickly fraudulent tactics-such as synthetic IDs, credential stuffing, and deepfakes have evolved.

Existing Solutions (what and who they're for):

- ***FICO Falcon Fraud Manager*** - An enterprise-grade platform providing real-time monitoring across transaction types, blending AI/ML models trained with insights gained from the Falcon Intelligence Network, which pulls information from 10,000+ institutions to improve detection whilst reducing false positives. Very powerful, but usually only adopted by large financial institutions. [5]
- ***SAS Fraud Managem****ent* - A comprehensive fraud platform that features machine learning, cross payments and events, real-time scoring and decisioning and alert/case management. Strong capabilities, usually placed in enterprise environments. [6]
- ***Kount (An Equifax Company)*** - AI/ML risk scoring and millisecond decisions. Leverages the Identity Trust Global Network (tens of billions of annual interactions) to assess risk using device, identity, and behavioral signals. More common in e-commerce/fintech onboarding and payments. [7]
- ***NICE Actimize (Enterprise Fraud Management / IFM-X)*** - Real-time cross-channel fraud prevention with AI-driven analytics and end-to-end case management with widespread adoption through mid-to-large financial institutions. [8]

Feature Comparison: focus on beneficial features

Product	Target Users	Cost/ Accessibility	Ease of Intergration	Intelligence pool	Typical deployment	Intergration with small FinTech
FICO Falcon	Huge firms	High	Complex	Yes	Enterprise	None
SAS Fraud Mgmt	Large enterprise	High	Complex	Not stated	Enterprise	Limited
Kount	Large Fintech	Medium	Moderate	Minimal	Cloud	Moderate
NICE Actimize	Mid-large banks	High	Complex	Not stated	Enterprise	Limited
Sentra	Small fintech	Low, Startup friendly	Easy, Simple APIS	Yes, Open cross firm sharding	Cloud	Strong

The existing offerings are all very effective, but they are mostly enterprise centric. Some of them use vendor-run networks (FICO, Kount) for sharing, but access is tied to purchasing their entire enterprise platforms, making them quite costly and complicated for small firms. There is little that pertains to open, collaborative sharing geared toward small fintech. Sentra fills this very gap with simple, privacy-respecting, cross-firm intelligence sharing and right-sized integrations for startups and small institutions. [5][7]

2.1.2 Problem Statement

Who has the problem?

- **Primary:** Smaller fintechs/startups that do not have the advantage of large bank budgets, data network effects and operational manpower.
- **Secondary:** Their customers (in terms of identity theft and direct financial loss) and regulators/industry (systemic risk and reputational loss). Record-level national loss figures and sector-specific spikes show the burden is meaningful and rising. [1][2][3]

Why is it important?

Losses keep increasing and the attacks are becoming increasingly sophisticated (synthetic identities, deepfakes, real-time payment scams), eroding customer confidence and increasing operational and regulatory pressure on smaller providers. [4]

Why existing solutions don’t solve it for small fintech?

- Fit and complexity: These platforms are built for very large enterprise deployment.[6][8]
- Access to pool: Where networks do exist (e.g., Falcon Intelligence Network; Kount's Identity Trust Global Network), they are tied to those vendors' stacks, creating a barrier for small teams trying to build an open, cross-firm hub. [5][7]
- Isolation in practice: Many solutions still stress single-institution monitoring; vendor pages say nothing about shared, peer-to-peer intelligence exchanges designed for small fintech participation. [6][8]

2.1.3 Product Vision

Sentra is a cloud-based centralized fraud detection platform for smaller fintech firms that:

- Allows for a collaborative intelligence regarding fraud (preservation of the privacy of patterns, entities, and behavior/device fingerprints) so repeated attacks will be blocked among participants.
- Provides ML-based anomaly detection in real time together with risk scores through secure APIs, along with a dashboard for analysts/compliance to investigate the alerts and trends.
- Emphasizes integration as well as security and compliance (encryption in transit/at rest, audit trails, and data minimization).

Vision: The most trusted fraud-intelligence hub for small fintech that turns an isolated defense into collective defense and reduces time-to-signal from weeks to minutes.

Main value: Losses and false negatives are reduced through pooled network effects, while costs/complexities lessen for smaller teams, improving customer protection and customer trust.

2.2 Stakeholders and Users

Internal:

- Project Lead (Ryaan Farrukh): Manages team activities, coordinates project timelines, ensures all deliverables meet QA standards and communicates with faculty.

- Backend Developer (Kencho Lodhen): Develops the fraud detection engine, handles integration with APIs and manages backend infra.
- UI/UX Designer (Anand Krishna Anil Kumar): Designs web dashboards, ensures user friendly interfaces and maintains clear project documentation.
- Testing and QA (shared): Each team member is responsible for individual and shared testing including unit and integration testing. Each member will also contribute to the validation and training of ML models.

External:

- Partner Fintech businesses: Benefits directly from shared fraud intelligence by reducing losses and risk.
- Incubators/Accelerators: Interested in supporting and innovative fraud prevention solution that can strengthen future fintech startups allowing higher success rates.
- Developers at Partner Firms: Intergration with our solution with existing fintech systems.
- Regulators: Ensures that platform complies with financial regulations and standards.
- End customers (indirect): Customers experience safer transactions and reduced risk of theft.

2.3 Project Scope

In Scope

- Development of Sentra's centralized fraud detection platform.
- Design and implementation of ML models for detection and risk scores.
- Creation of secure APIs for fintech firms to integrate with.
- Development of a web dashboard to monitor, report and visualize anomalies and data.
- Incorporation of encryption and data minimizations to reduce overall risk.
- Full system documentation from system design, architecture and usage guides.

Out of Scope

- On premises deployment (Our solution will be cloud based).
- Legacy systems integration beyond APIs.
- Development of any mobile application for end consumers.
- Services for non-financial industries.

2.4 System Risks

Privacy Risk

- Exposure of sensitive financial and customer data if encryption or access controls are not properly developed and implemented.

Integration Risk

- Partner fintech firms may use different data forms or incompatible APIs causing onboarding and integration issues, furthering workloads and possibly delayed deadlines.

Model Risk

- ML models may produce false positives/negatives reducing reliability and accuracy to determine real fraud attempts.

Adoption Risk

- Small fintechs may hesitate to share sensitive information due to competition, cost or uncertainty in solution.

Scalability Risk

- Large volumes of data processing could overwhelm resources if cloud services and scaling are not properly optimized or configured resulting in high latency or system downtimes.

Security Risk

- Potential for breaches or exploitation by attackers in attempts to manipulate fraud signals and/or to bypass detection systems

Regulatory Risk

- Lack of compliance with global and national stands could result in fines, reputation damage or at worse consequences such as being shut down altogether

2.5 Operating Environment

Hosting Environment

- Sentra will be deployed on a secure cloud infra (AWD, Azure or GCP) to maintain scalability requirements and uptime

Client Access

- End users will access the system through a web-based dashboard that runs on standard browsers (Chrome, Edge, Firefox, Safari)
-

Integration

- Partner firms will connect through a REST API secured with TLS encryption and OAuth authentication

Supported Platforms

- Platform APIs will be designed to integrate with modern tech stacks (Python, Node.js Java)

Security

- The development and operating environment must comply with the financial security requirements such as PCI DSS (Payment Card Industry Data Security Standard) and data regulations such as GDPR (General Data Protection Regulation)

Data Managment

- Logs, fraud alerts and models will all be secured in an encrypted large-scale database hosted in the cloud with automated backups.

2.6 Functional Requirements

2.6.1 Requirements Gathering Process

To identify all the functional and nonfunctional requirements for our solution, we simulated interviews with key personnel. These personas reflect real users who will interact or benefit from the system we create.

Person 1: Fraud Analyst (External User)

- Role: Reviews alert and investigates them
- Goal: Needs accurate and quick alerts to act quickly
- Frustrations: Current systems produce too many false alerts and too cluttered

Sample Interview Q&A

- *Q: How quickly do you need alerts?*
  A: “Within seconds, otherwise the fraudster can get away”
- *Q: What information do you need in an alert*
  A: “Transaction details, risk scores and why it was flagged”
- *Q: What is frustrating in dashboards*
  A: “Too many extra screens, I need a clean and simple dashboard”

Keywords -> Requirements

- “Within seconds” -> NFR1: System should return fraud risk scores within 2 seconds
- “Risk score and reasons” -> FR6: APIs should return risk scores and reasons
- “Clean and simple” -> FR8: Dashboard should show alert simple with summary

Person 2: Compliance Officer (External User)

- Role: Makes sure fraud reporting meets legal and regulatory requirements
- Goal: Needs accurate and exportable reports and full audit trails.
- Frustrations: Manual reporting wastes time and the logs are incomplete

Sample Interview Q&A

- *Q: What do you need most for compliance?*
  A: “Exportable reports with timestamps.”
- *Q: What helps with audits?*
  A: “Complete logs of every fraud event and action.”

Keywords -> Requirements

- “Exportable reports” -> FR10: Generate compliance reports
- “Complete logs” ->FR4: Store all alerts and logs in a secure database with audit trails.
- “Compliance” -> NFR5: Comply with PCI DSS and GDPR.

Person 3: Fintech Developer (External User)

- Role: Integrates our APIs into partner systems
- Goal: Needs simple and secure APIs
- Frustrations: Enterprise systems are too complex and difficult to integrate

Sample Interview Q&A

- Q: *What matters most when integrating APIs?*
  A: “They must be secure and return info fast”
- Q: *What slows you down?*
  A: “Inconsistencies in code and poor documentation”

Keywords -> Requirements

- “Secure and fast” -> FR7: APIs should use OAuth 2.0/TLS and respond within 2 seconds.
- “Poor documentation” -> NFR8: Dashboard and APIs should be intuitive and consistent.

Person 4: Project Lead (Internal User)

- Role: Oversees the project and ensures the stability of our system
- Goal: Needs to be scalable and reliable for multiple partners
- Frustrations: Downtime reduces partner trust and scaling issues increases cost

Sample Interview Q&A

- Q: *What is critical for system stability?*
  A: “The system must be scalable for lots of partners”
- Q: *How much downtime is allowed?*
  A: “Almost none, we aim for 99.9% uptime”

Keywords -> Requirements

- “Scale for lots of partners” -> NFR6: System should scale to support 50+ partners.
- “99.9% uptime” -> NFR3: System should provide 99.9% uptime excluding maintenance.
- “Without crashing” -> NFR2: Cloud environment should scale to handle load.

2.6.2 Functional Requirements (FR)

FR1 – Real-Time Fraud Detection
The system shall analyze incoming transaction data in real time to identify potentially fraudulent activities. Using predefined rules and trained machine-learning models, the platform must process transaction payloads (including transaction ID, user ID, amount, device fingerprint, and time) and return a risk score within seconds.
Purpose: Enables immediate action by fintech partners before fraudulent transactions are completed.
Benefit: Reduces the time window available to fraudsters and helps minimize financial losses.

FR2 – Alert Generation and Case Management
Whenever a transaction exceeds a defined risk threshold, the system shall automatically create a fraud alert entry. Each alert must include details such as the transaction ID, risk score, partner firm, timestamp, and detection reason. Alerts will be stored in a secure database and made viewable through the analyst dashboard.
Purpose: Provides analysts and compliance officers with centralized visibility over flagged events.
Benefit: Streamlines investigations, ensures traceability, and supports audit requirements.

FR3 – Cross-Firm Intelligence Sharing
The platform shall allow participating fintech firms to share anonymized fraud intelligence through a secure data exchange layer. Shared insights may include device fingerprints, behavioral anomalies, or transaction pattern identifiers without exposing personal customer data.
Purpose: Creates a collective defense network where detection in one firm benefits all participants.
Benefit: Enhances overall fraud detection accuracy while maintaining strict data privacy.

FR4 – Secure Partner API Integration
Sentra shall provide RESTful APIs for partner fintech systems to submit transaction data, request risk scores, and retrieve alerts. The APIs must require OAuth 2.0 authentication and TLS 1.3 encryption for all communications.
Purpose: Ensures partners can safely and easily integrate their applications without compromising security.
Benefit: Simplifies onboarding and promotes adoption by small fintech firms.

FR5 – Centralized Logging and Audit Trails
All system activities, including API calls, alerts, data exchanges, and administrative actions, shall be logged automatically. Logs must include timestamps, actor IDs, and action details. Audit logs shall be immutable and retained for regulatory review.
Purpose: Maintains transparency and traceability of every event for compliance and troubleshooting.
Benefit: Supports legal audits, meets PCI DSS and GDPR obligations, and enhances platform trust.

FR6 – Analyst Dashboard
The system shall include a secure web-based dashboard for analysts and administrators. The dashboard will display alerts, fraud trends, partner statistics, and ML performance summaries through charts and tables. Users shall be able to filter results by partner name, date, transaction type, or risk score.
Purpose: Provides an intuitive interface for users to monitor fraud activity and act efficiently.
Benefit: Improves productivity and reduces analyst fatigue caused by cluttered interfaces.

FR7 – Role-Based Access Control
Sentra shall implement user roles (Admin, Analyst, Partner Developer) to restrict access based on privileges.
Purpose: Protects sensitive data and ensures users only access the functions relevant to their responsibilities.
Benefit: Enhances system security and regulatory compliance.

FR8 – Reporting and Export Features
Authorized users shall be able to generate downloadable compliance and performance reports in PDF or CSV format. Reports will summarize fraud cases, model accuracy, and incident resolution status.
Purpose: Assists compliance officers in meeting reporting requirements.
Benefit: Saves manual effort and ensures documentation consistency across partner organizations.

FR9 – System Administration
The system shall provide administrative tools to register new partner fintech firms, manage user accounts, update ML models, and configure thresholds for fraud detection.
Purpose: Gives system administrators full control of operational management.
Benefit: Simplifies maintenance and supports scalability as more partners onboard.

2.7 Nonfunctional Requirements

NFR1 - Performance and Reliability

The platform must provide near-instantaneous detection and high system uptime:

- Average API response time for fraud risk scoring shall be **≤ 2 seconds** under normal load (up to 500 requests per minute).
- The cloud infrastructure must ensure **99.9 % uptime**, excluding scheduled maintenance.
- Automated health checks and logging mechanisms must restart failed services and alert the admin when downtime is detected.

Purpose: Guarantees users receive timely, uninterrupted service during fraud detection operations.

Benefit: Builds partner confidence and prevents financial losses caused by delayed responses.

NFR2 – Security and Compliance

- All system data—both in transit and at rest—must be encrypted using **TLS 1.3** and **AES-256** standards. The system shall adhere to recognized data protection frameworks such as **PCI DSS v4.0** for payment security and **GDPR Article 32** for data privacy. Multi-factor authentication must be implemented for all admin and partner accounts.

Purpose: Ensures regulatory compliance and protects sensitive financial information from unauthorized access or leakage.
Benefit: Prevents reputational damage, legal penalties, and loss of trust among partner fintech firms.

NFR3 – Scalability and Maintainability

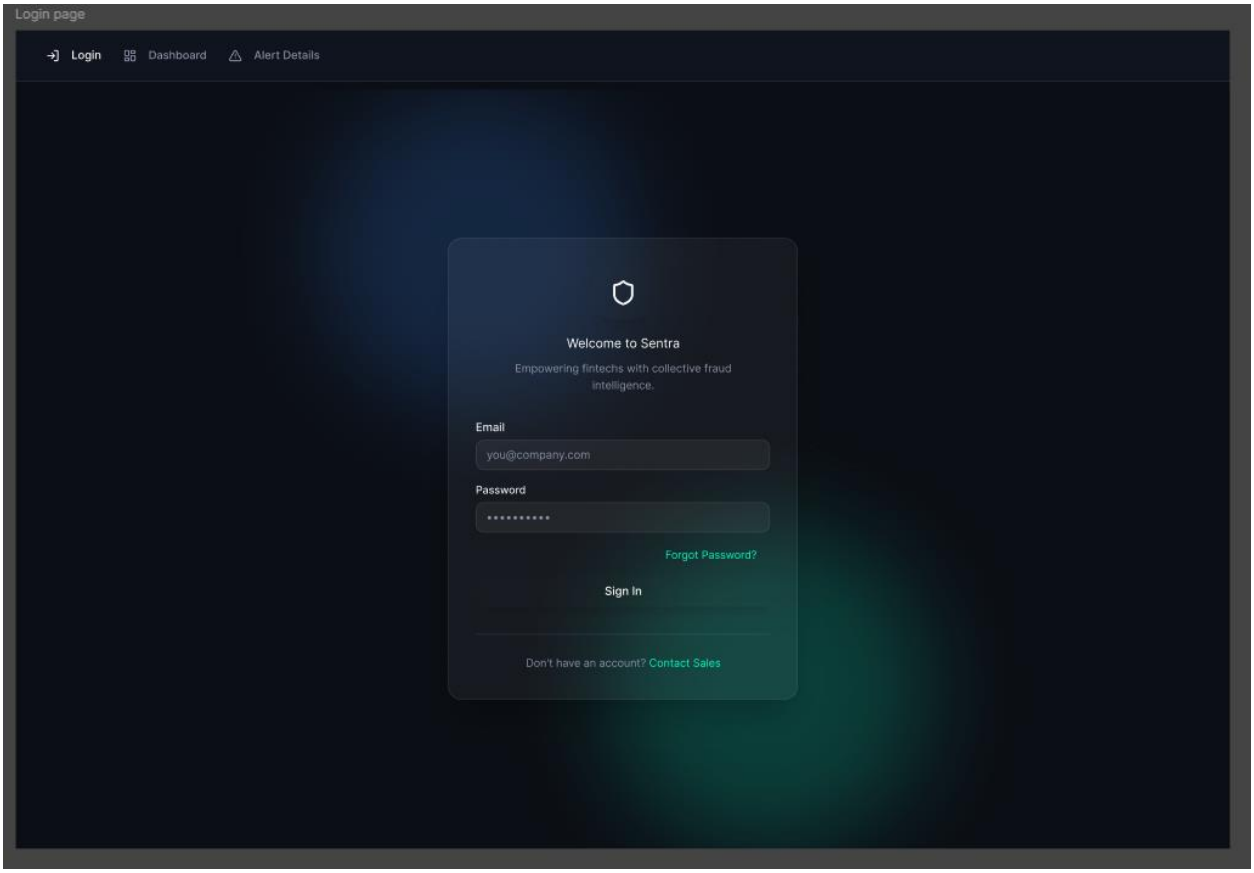
- The system architecture must be modular, containerized, and deployable on scalable cloud services such as AWS Elastic Beanstalk or GCP Compute Engine. The backend should be designed to handle a growing number of partner integrations without major reconfiguration.

Purpose: Allows the platform to grow from a few initial fintech partners to 50+ organizations.

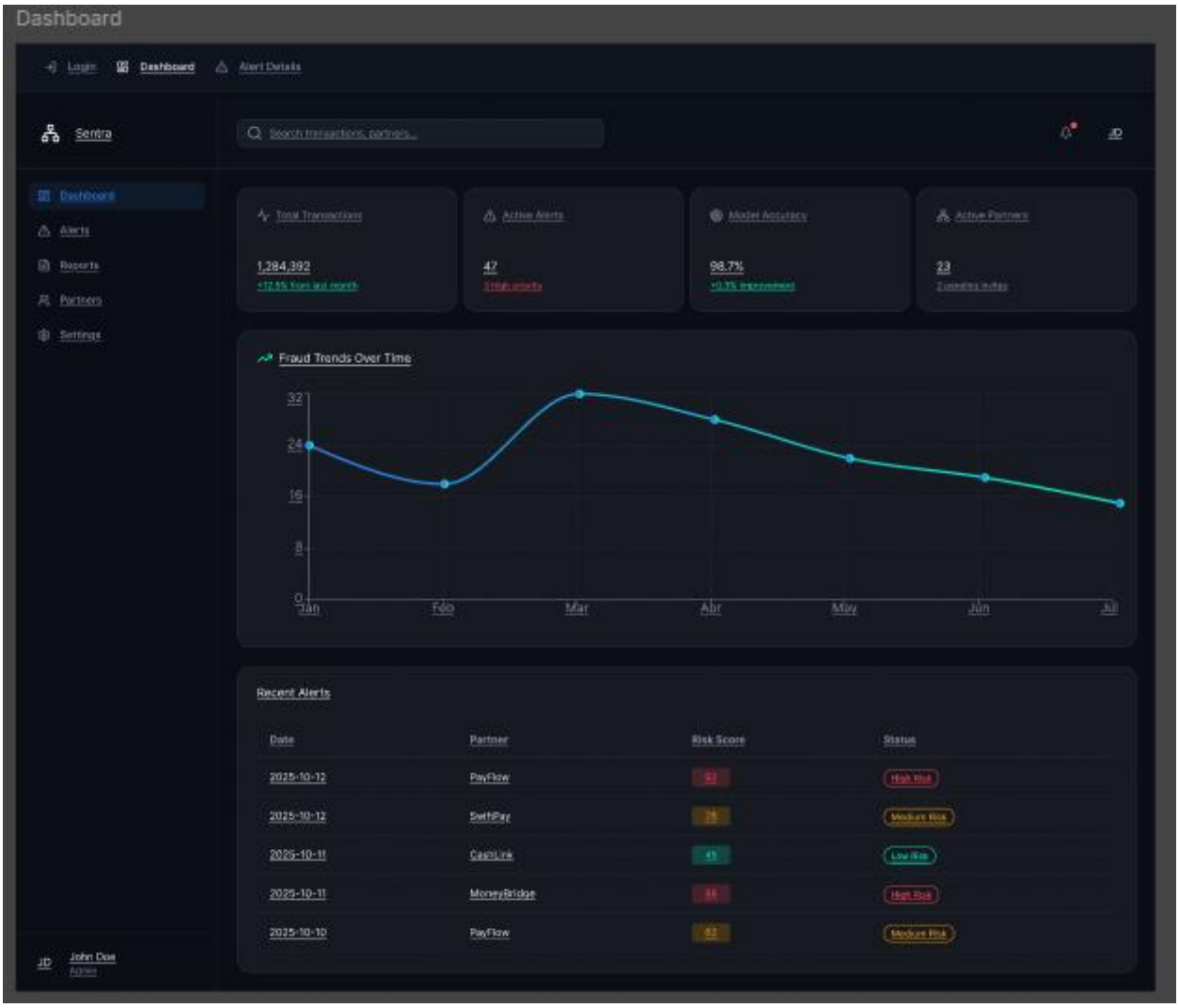
Benefit: Ensures long-term sustainability, easier updates, and reduced maintenance cost.

2.8 UI/UX Interface Mock-ups

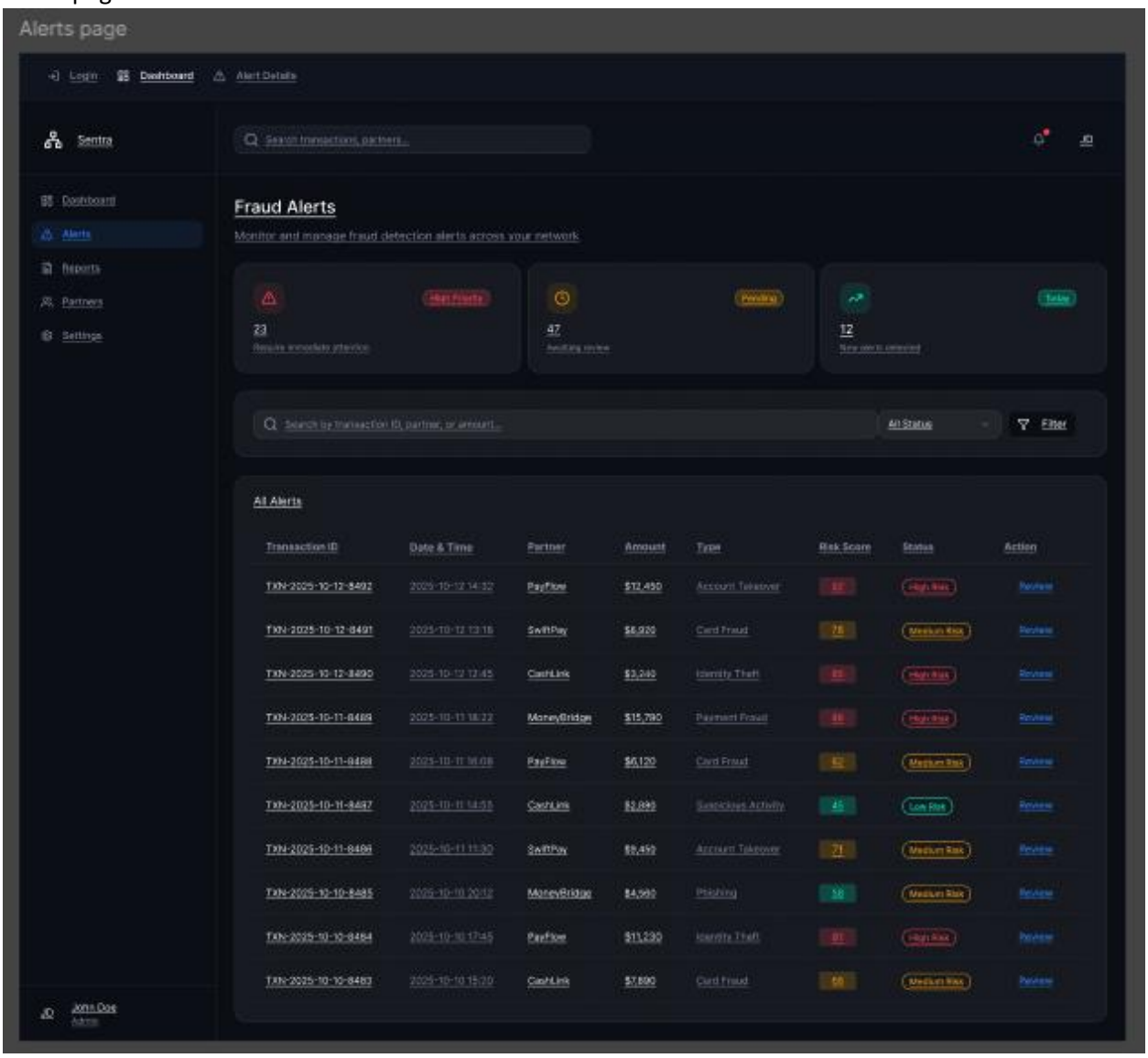
Login page:



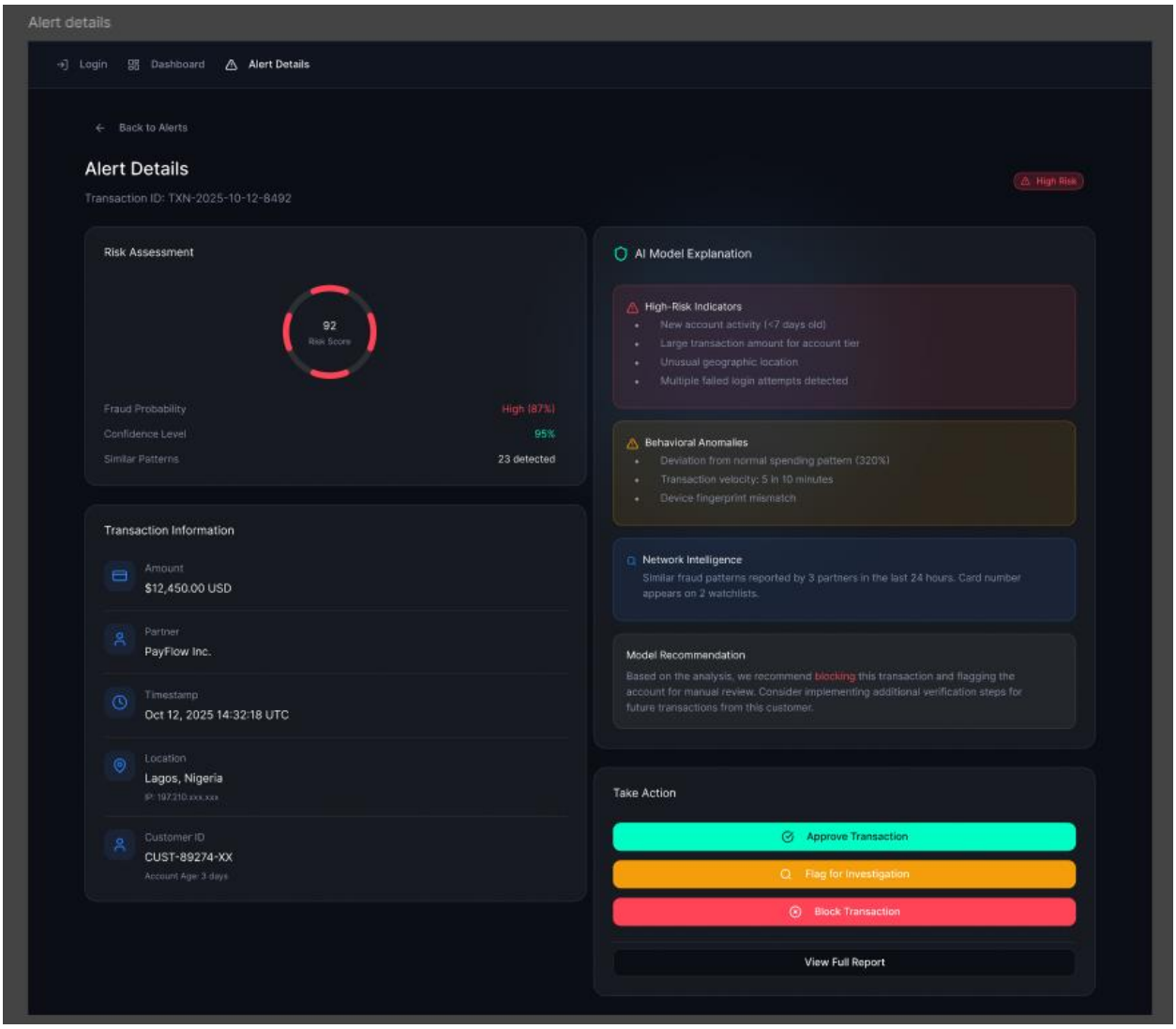
Dashboard:


Dashboard
Login
Dashboard
Alert Details
Sentra
Search transactions, partners...
Dashboard
Alerts
Reports
Partners
Settings
Total Transactions
1,284,392
Active Alerts
47
Model Accuracy
98.7%
Active Partners
23
Fraud Trends Over Time
32
24
16
8
0
Jan
Feb
Mar
Abr
May
Jun
Jul
Recent Alerts
Date
Partner
Risk Score
Status
2025-10-12
PayFlow
High Risk
2025-10-12
SwiftPay
Medium Risk
2025-10-11
CashLink
Low Risk
2025-10-11
MoneyBridge
High Risk
2025-10-10
PayFlow
Medium Risk
JD
John Doe
Admin

Alerts page:

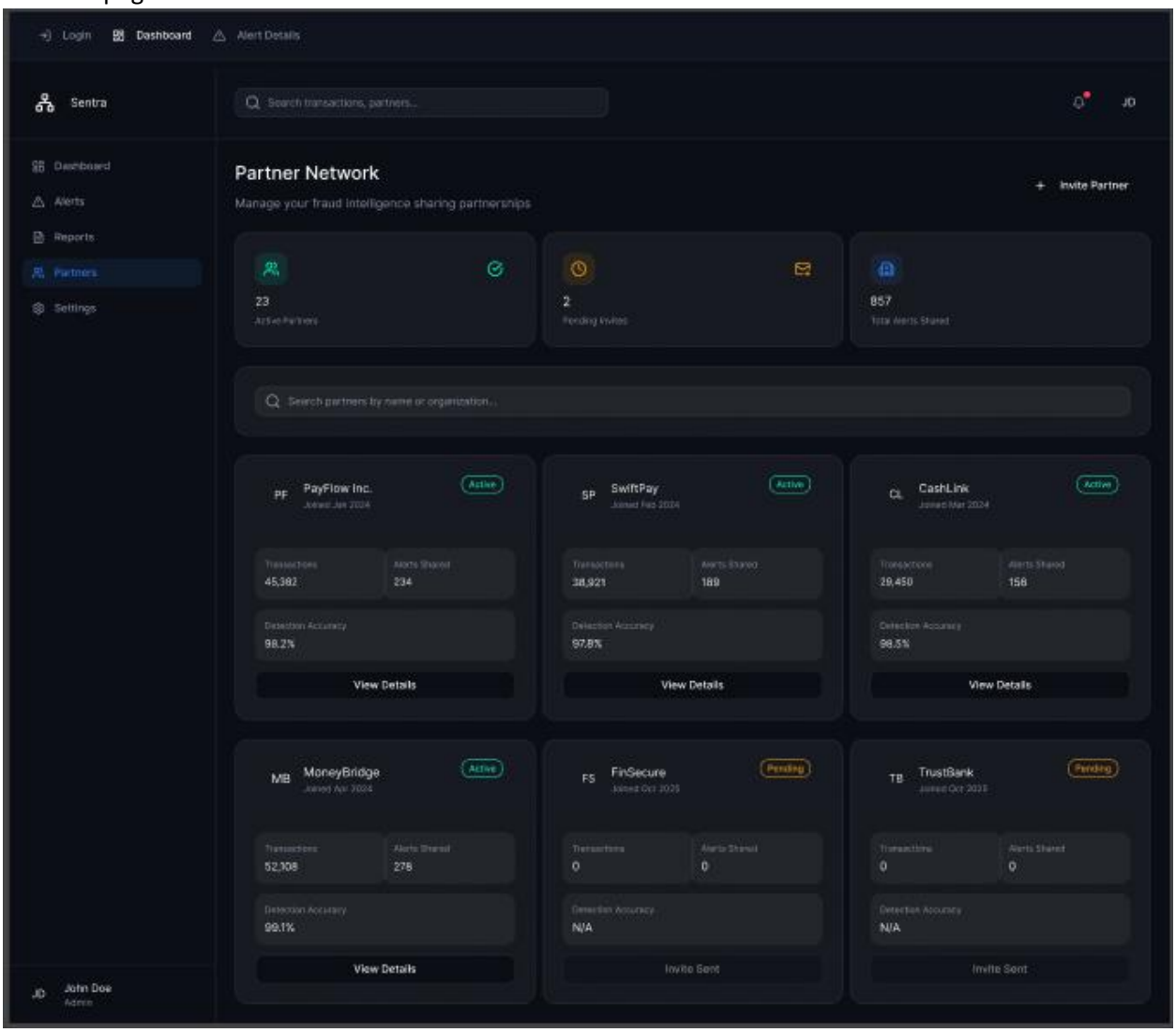

Alerts page
Login
Dashboard
Alert Details
Sentra
Search transactions, partners...
Dashboard
Alerts
Reports
Partners
Settings
Fraud Alerts
Monitor and manage fraud detection alerts across your network
23
47
12
Search by transaction ID, partner, or amount...
All Status
Filter
All Alerts
Transaction ID
Date & Time
Partner
Amount
Type
Risk Score
Status
Action
TXN-2025-10-12-8492
PayFlow
$12,450
Account Takeover
TXN-2025-10-12-8491
SwiftPay
$6,920
Card Fraud
TXN-2025-10-12-8490
CashLink
$3,240
Identity Theft
TXN-2025-10-11-8489
MoneyBridge
$15,780
Payment Fraud
TXN-2025-10-11-8488
PayFlow
$6,120
Card Fraud
TXN-2025-10-11-8487
CashLink
$2,890
Suspicious Activity
TXN-2025-10-11-8486
SwiftPay
$8,450
Account Takeover
TXN-2025-10-10-8485
MoneyBridge
$4,960
Phishing
TXN-2025-10-10-8484
PayFlow
$11,230
Identity Theft
TXN-2025-10-10-8483
CashLink
$7,890
Card Fraud
John Doe
Admin

Alert details:

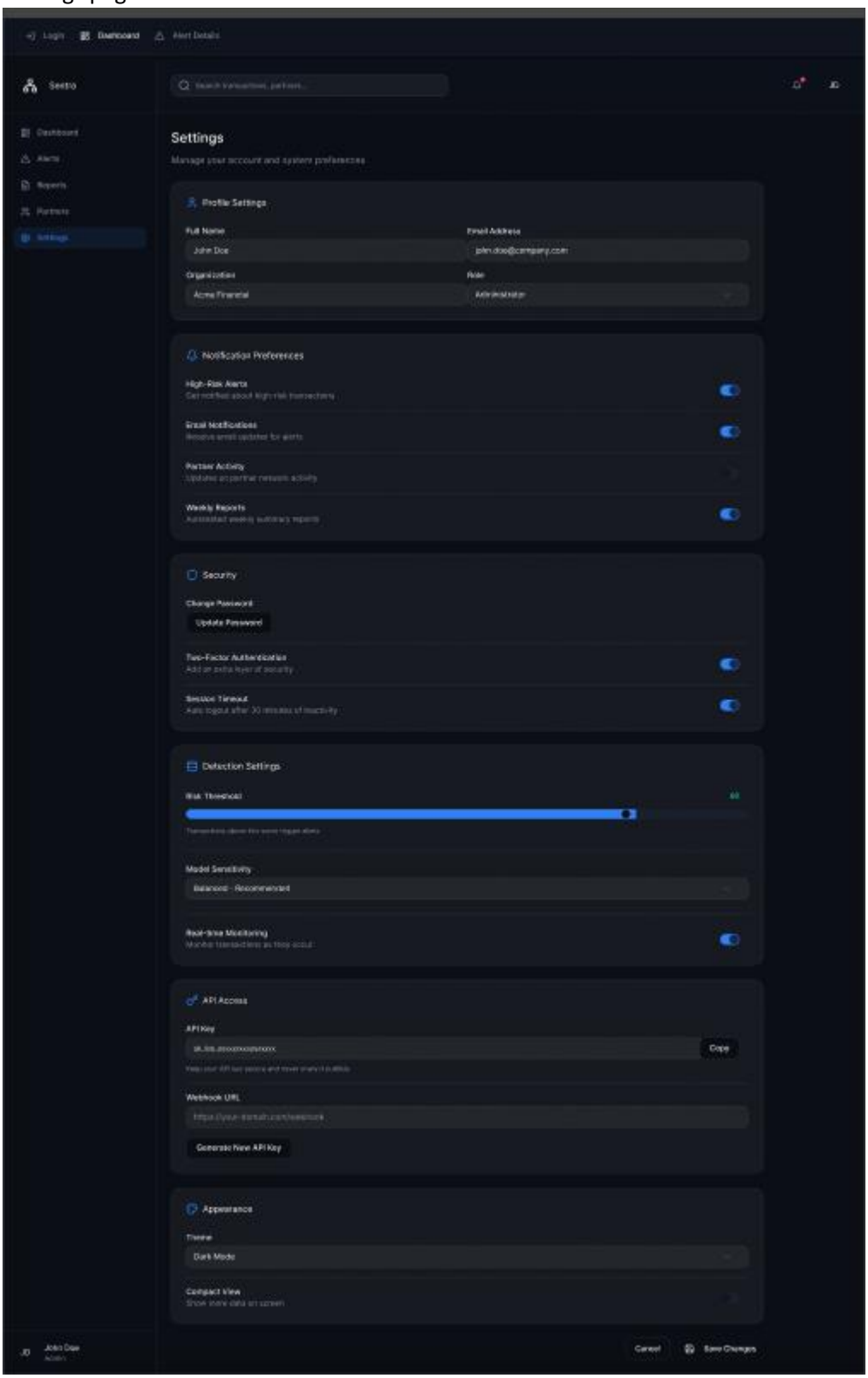


Reports page:

Partners page:



Settings page:



References (Section 2)

[1] Federal Trade Commission, “New FTC Data Show a Big Jump in Reported Losses to Fraud to $12.5 Billion in 2024,” Mar. 10, 2025. [Online]. Available: https://www.ftc.gov/news-events/news/press-releases/2025/03/new-ftc-data-show-big-jump-reported-losses-fraud-125-billion-2024

[2] Federal Bureau of Investigation, “FBI Releases Annual Internet Crime Report,” Apr. 23, 2025. [Online]. Available: https://www.fbi.gov/news/press-releases/fbi-releases-annual-internet-crime-report

[3] Veriff, “Top fraud trends in digital banking for 2025 – and how to stay one step ahead,” Jul. 30, 2025. [Online]. Available: https://www.veriff.com/fraud/learn/top-fraud-trends-in-digital-banking-for-2025-and-how-to-stay-one-step-ahead

[4] Veriff, “The growing threat of deepfakes in financial services and why a trust infrastructure is the future,” Aug. 20, 2025. [Online]. Available: https://www.veriff.com/identity-verification/the-growing-threat-of-deepfakes-in-financial-services-and-why-a-trust-infrastructure-is-the-future

[5] FICO, “FICO® Falcon® Fraud Manager,” [Online]. Available: https://www.fico.com/en/products/fico-falcon-fraud-manager

[6] SAS, “SAS Fraud Management,” [Online]. Available: https://www.sas.com/en_us/software/fraud-management.html

[7] Kount (Equifax), “Fraud Detection Software / Identity Trust Platform,” [Online]. Available: https://kount.com/fraud-detection-software

[8] NICE Actimize, “Enterprise Fraud Management,” [Online]. Available: https://www.niceactimize.com/fraud-management

Section 3

3.1 Data Flow Diagrams

3.1.1 Context Diagram (Level 0)

The context diagram shows Sentra's interactions with external entities:

External Entities:

- Partner Fintech Systems
- Fraud Analysts
- Compliance Officers
- System Administrators
- Partner Developers

Major Data Flows:

- Transaction Data → Sentra Platform
- Risk Scores & Alerts → Partner Systems
- Fraud Intelligence (anonymized) → Shared Intelligence Pool
- Dashboard Access → Analysts/Officers
- Configuration & Management → Administrators

3.1.2 Level 1 DFD - Core System Processes

Process 1.0: Authenticate & Authorize

- Input: Login credentials, OAuth tokens
- Output: Session tokens, access permissions
- Data Store: User Accounts DB

Process 2.0: Process Transaction Data

- Input: Transaction payloads from Partner APIs
- Output: Validated transaction records
- Data Store: Transaction Log DB

Process 3.0: Analyze Fraud Risk

- Input: Transaction data, historical patterns, ML models
- Output: Risk scores, detection reasons

- Data Store: ML Models DB, Fraud Patterns DB

Process 4.0: Generate Alerts

- Input: High-risk transactions (score > threshold)
- Output: Fraud alerts with details
- Data Store: Alerts DB, Audit Logs

Process 5.0: Share Intelligence

- Input: Anonymized fraud patterns
- Output: Shared intelligence indicators
- Data Store: Shared Intelligence Pool

Process 6.0: Display Dashboard

- Input: Alert queries, filter parameters
- Output: Visualizations, reports, trends
- Data Store: Alerts DB, Partner DB, Analytics DB

Process 7.0: Manage System

- Input: Admin commands, configuration changes
- Output: Updated settings, new partners, user accounts
- Data Store: Configuration DB, Partner DB, User Accounts DB

3.1.3 Level 2 DFD - Fraud Analysis Subsystem (Process 3.0)

Process 3.1: Validate Input

- Checks data completeness and format.
- Filters out malformed requests

Process 3.2: Extract Features

- Extracts transaction amount, time, location
- Generates device fingerprint hash.
- Calculates velocity metrics (transactions/hour)

Process 3.3: Apply Rule Engine

- Checks predefined fraud rules (e.g., amount > $10,000)
- Flags suspicious patterns (e.g., multiple failed attempts)

Process 3.4: Run ML Model

- Passes features to trained ML model.
- Generates probability score (0-100)

Process 3.5: Calculate Risk Score

- Combines rule engine and ML outputs.
- Normalizes to final risk score.
- Determines detection reasons.

Process 3.6: Check Shared Intelligence

- Queries shared pool for matching device/behavior patterns.
- Adjusts risk score based on cross-firm data.

3.2 User Stories and related Use Case Scenarios

User Story 1: Real-Time Fraud Detection

As a Partner Fintech System
I want to submit transaction data and receive instant risk scores
So that I can block fraudulent transactions before they complete

Related Use Case: UC1 - Submit Transaction for Analysis

Actors: Partner System (Primary), Sentra API (System)

Preconditions:

- Partner has valid OAuth token.
- API endpoint is operational.

Main Flow:

1. Partner system sends POST request with transaction payload (ID, user_id, amount, device_fingerprint, timestamp)
2. Sentra validates authentication token.
3. Sentra validates transaction data format.
4. Sentra processes transaction through fraud detection engine
5. Sentra returns JSON response with risk_score, transaction_id, and detection_reason
6. Partner system receives response within 2 seconds.

Postconditions:

- Transaction logged in database
- Risk score returned to partner
- Alert generated if score > threshold

Alternative Flows:

- 2a. Invalid token → Return 401 Unauthorized
- 3a. Malformed data → Return 400 Bad Request with error details
- 6a. Timeout → Partner retries with exponential backoff.

User Story 2: Alert Investigation

As a Fraud Analyst
I want to view and investigate fraud alerts in a dashboard
So that I can act on suspicious transactions quickly

Related Use Case: UC2 - View Fraud Alerts

Actors: Fraud Analyst (Primary), Dashboard UI (System)

Preconditions:

- Analyst has logged in with valid credentials.
- Alerts exist in the system.

Main Flow:

1. Analyst navigates to Alerts page.
2. System displays list of alerts sorted by timestamp (newest first)
3. Each alert shows: Transaction ID, Partner Name, Risk Score, Timestamp, Status
4. Analyst applies filters (date range, partner, risk score range)
5. System updates alert list based on filters.
6. Analyst clicks on an alert to view details.
7. System displays full alert details including transaction data, detection reason, device fingerprint.

Postconditions:

- Analyst has visibility into flagged transactions.
- Alert status can be updated (e.g., "Under Review")

Alternative Flows:

- 2a. No alerts found → Display "No alerts to show" message.
- 4a. Invalid filter → Show validation error

Related Use Case: UC3 - Update Alert Status

Main Flow:

1. Analyst is viewing alert details (from UC2)
2. Analyst selects new status from dropdown (Under Review, False Positive, Confirmed Fraud, Resolved)
3. Analyst optionally adds investigation notes.
4. Analyst clicks "Update Status."
5. System validates analyst permissions.
6. System updates alert status in database
7. System logs action in audit trail

8. System displays success message.

User Story 3: Cross-Firm Intelligence Sharing

As a Partner Fintech
I want to contribute anonymized fraud patterns to a shared pool
So that all participants can benefit from collective detection

Related Use Case: UC4 - Share Fraud Intelligence

Actors: Sentra Platform (Primary), Partner Systems (Secondary)

Preconditions:

- Fraud confirmed by partner
- Pattern meets sharing criteria (severity, confidence level)

Main Flow:

1. Sentra identifies confirmed fraud case.
2. System anonymizes sensitive data (removes user_id, transaction details)
3. System extracts shareable indicators (device fingerprint hash, behavioral patterns, IP geolocation)
4. System adds indicators to Shared Intelligence Pool
5. System notifies other partners of new intelligence (optional)
6. Future transactions are checked against shared pool.

Postconditions:

- Anonymized fraud pattern stored in shared pool.
- All partners benefit from updated detection rules.

User Story 4: Partner Onboarding

As a System Administrator
I want to register new fintech partners
So that they can integrate with the platform securely

Related Use Case: UC5 - Register New Partner

Actors: Administrator (Primary), Registration System (System)

Preconditions:

- Administrator logged in
- Partner has completed registration form.

Main Flow:

1. Admin navigates to Partners page.
2. Admin clicks "Add New Partner."
3. Admin enters partner details (Company Name, Contact Email, Industry Type)
4. System validates input.
5. System generates unique API credentials (Client ID, Client Secret)
6. System stores partner record in database
7. System displays API credentials (one-time view)
8. Admin securely shares credentials with partner.

Postconditions:

- Partner registered in system
- API credentials generated
- Partner can begin integration.

Alternative Flows:

- 4a. Duplicate partner name → Show error and suggest modification.

User Story 5: Compliance Reporting

As a Compliance Officer
I want to generate and export fraud reports
So that I can meet regulatory requirements

Related Use Case: UC6 - Generate Compliance Report

Actors: Compliance Officer (Primary), Reporting System (System)

Preconditions:

- Officer logged in with appropriate permissions
- Data exists for selected period.

Main Flow:

1. Officer navigates to Reports page.

2. Officer selects report type (Fraud Summary, Alert Resolution, Partner Activity)
3. Officer specifies date range and filters (partner, status)
4. Officer clicks "Generate Report."
5. System queries database for relevant records
6. System compiles report data
7. System generates PDF/CSV file.
8. System displays download link.
9. Officer downloads report

Postconditions:

- Report generated and logged.
- Officer has exportable documentation.

User Story 6: Dashboard Monitoring

As a Fraud Analyst
I want to view real-time fraud trends and statistics
So that I can identify patterns and respond proactively

Related Use Case: UC7 - View Dashboard Metrics

Actors: Analyst (Primary), Dashboard System (System)

Preconditions:

- Analyst logged in
- System has collected transaction and alert data.

Main Flow:

1. Analyst accesses main Dashboard page
2. System displays key metrics:
    - Total transactions today
    - Fraud alerts today
    - Detection rate (%)
    - Top fraud types
3. System shows visualizations:

- Alerts over time (line chart)
- Risk score distribution (histogram)
- Alerts by partner (pie chart)

4. System auto-refreshes data every 30 seconds
5. Analyst can click on chart elements to drill down.

Postconditions:

- Analyst has current visibility into fraud activity.

User Story 7: API Integration

As a Partner Developer
I want to integrate Sentra's API into our system
So that we can submit transactions and receive risk scores

Related Use Case: UC8 - Integrate Partner System

Actors: Developer (Primary), Sentra API (System)

Preconditions:

- Partner has API credentials.
- Developer has access to API documentation.

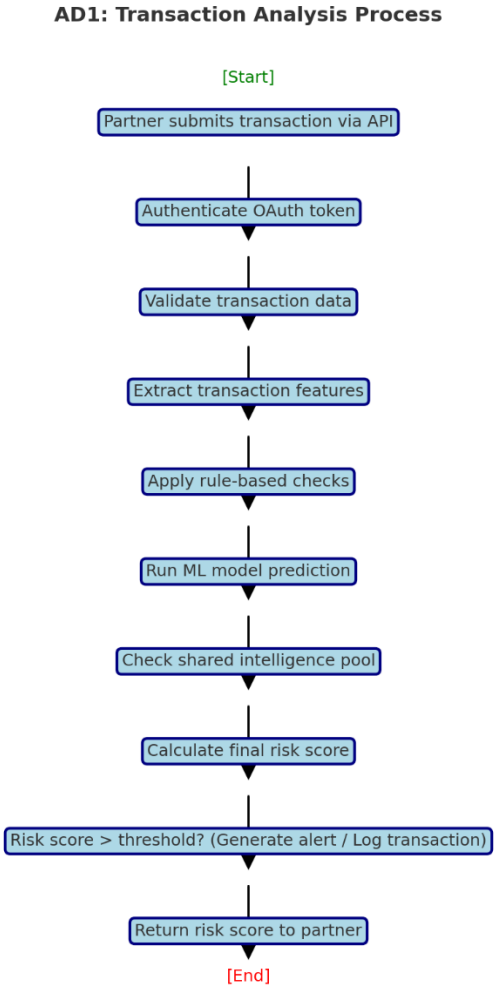
Main Flow:

1. Developer implements OAuth 2.0 authentication flow.
2. Developer obtains access token.
3. Developer sends test transaction via POST /api/v1/analyze.
4. System validates request format.
5. System returns risk assessment response.
6. Developer handles response in application logic
7. Developer implements error handling for timeouts, invalid responses.
8. Developer completes integration testing.

Postconditions:

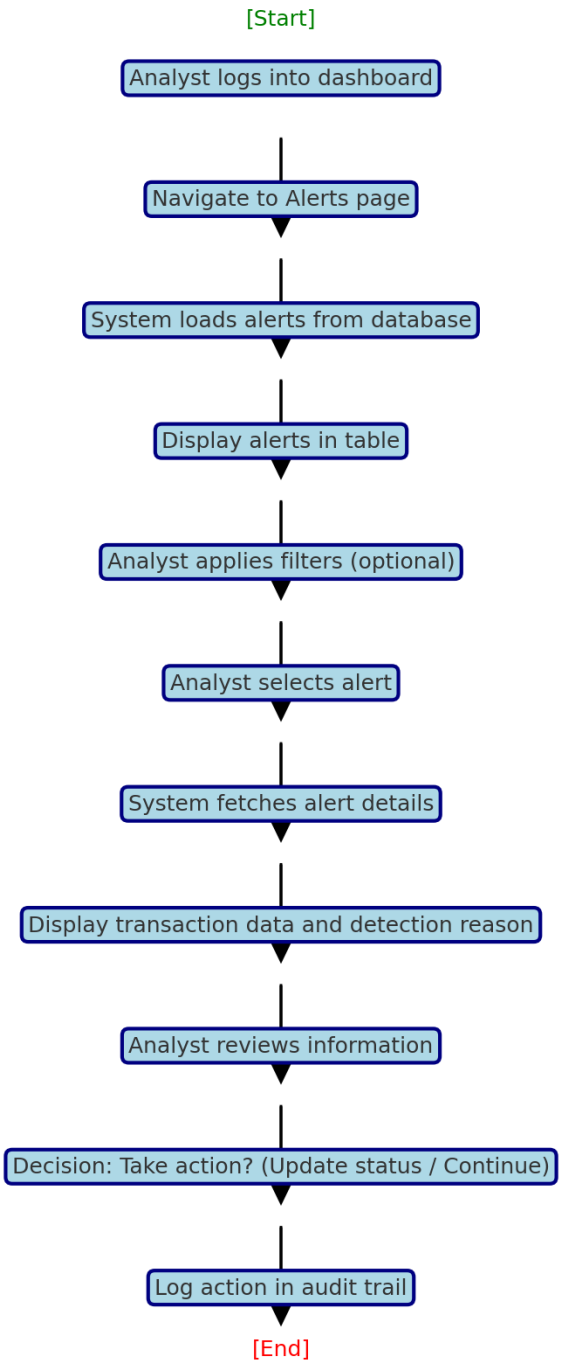
- Partner system successfully integrated.
- Transactions flow to Sentra for analysis

3.3 Activity Diagrams

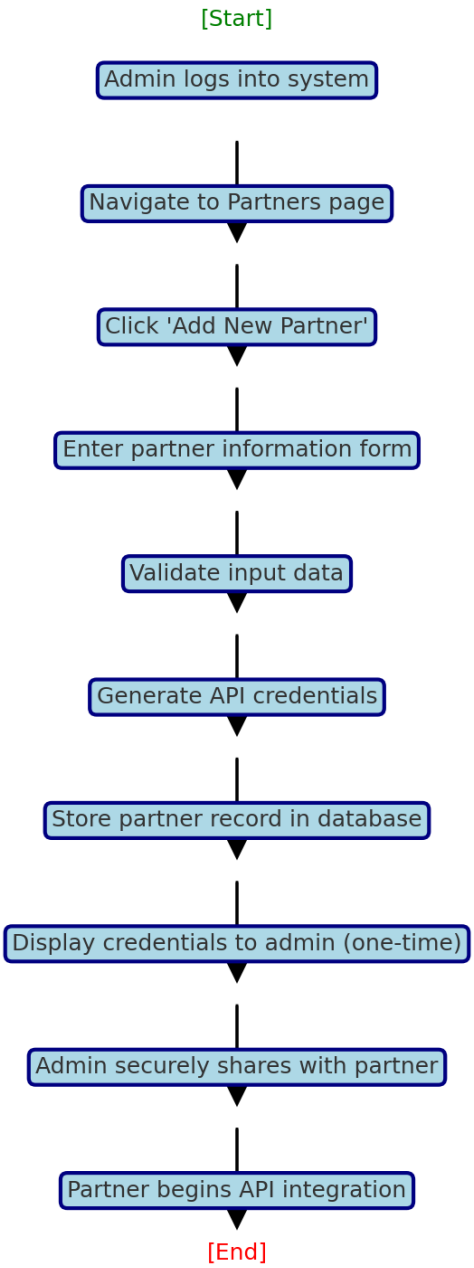


Description: This diagram shows the complete flow from transaction submission to risk score delivery, including validation, analysis, and alert generation steps.

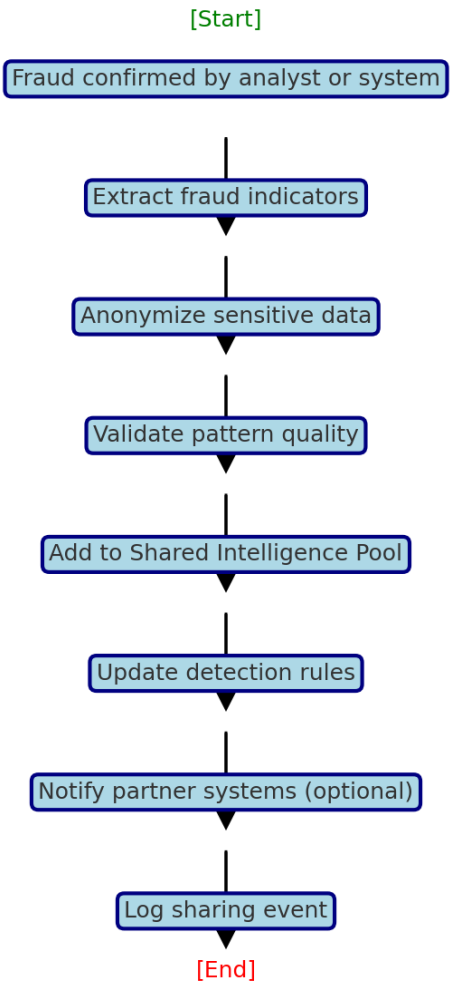
AD2: Alert Investigation Workflow



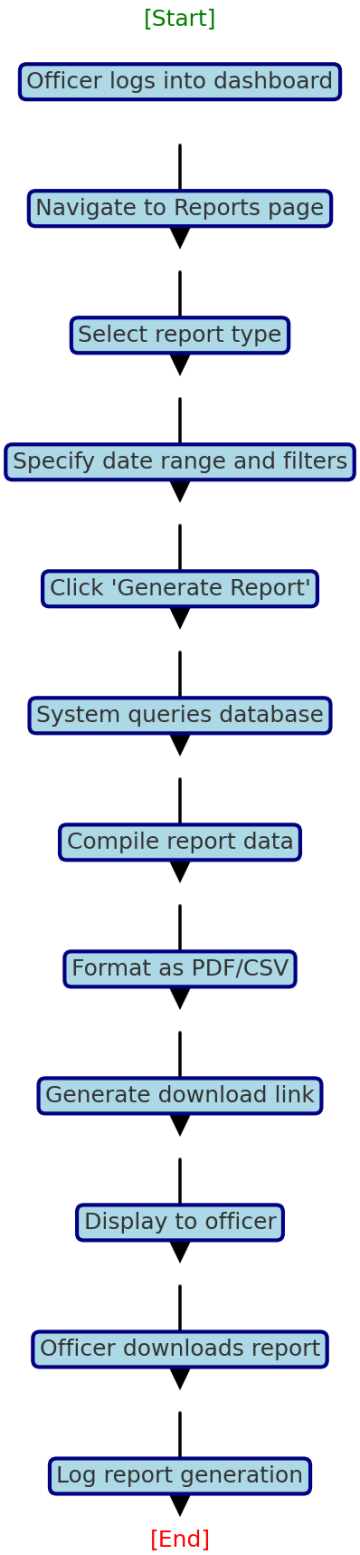
AD3: Partner Onboarding Process



AD4: Intelligence Sharing Process



AD5: Report Generation Workflow



3.4 Business Rules

Business Rule #	Description	Activity Diagram	Related UCS	UI Mock-up
BR1	All API requests must include valid OAuth 2.0 token	AD1	UC1, UC8	N/A (API)
BR2	Risk scores must be calculated within 2 seconds	AD1	UC1	N/A
BR3	Alerts are automatically generated when risk score > 75	AD1	UC1, UC2	UI 2.8 (Alerts)
BR4	Only Admin and Analyst roles can update alert status	AD2	UC3	UI 2.8 (Alert Details)
BR5	Transaction data must include transaction id, user id, amount, timestamp, device fingerprint	AD1	UC1	N/A (API)
BR6	Shared intelligence must be anonymized before storage	AD4	UC4	N/A
BR7	Only Admins can register new partners	AD3	UC5	UI 2.8 (Partners)
BR8	API credentials are displayed only once during partner creation	AD3	UC5	UI 2.8 (Partners)
BR9	Compliance Officers or Admins can only generate compliance reports	AD5	UC6	UI 2.8 (Reports)
BR10	All system actions must be logged in audit trail with timestamp and actor id	AD1, AD2, AD3, AD4, AD5	All	N/A
BR11	Dashboard metrics auto-refresh every 30 seconds	N/A	UC7	UI 2.8 (Dashboard)
BR12	Users must authenticate with username and password; Admins require MFA	N/A	All	UI 2.8 (Login)
BR13	Failed login attempts are limited to 5 within 15 minutes	N/A	All	UI 2.8 (Login)
BR14	Alert statuses: Pending, Under Review, False Positive, Confirmed Fraud, Resolved	AD2	UC2, UC3	UI 2.8 (Alert Details)
BR15	Reports are retained for 7 years to meet regulatory requirements	AD5	UC6	N/A
BR16	All data at rest must be encrypted using AES-256	N/A	All	N/A
BR17	All data in transit must use TLS 1.3	AD1	UC1, UC8	N/A
BR18	System must support minimum 50 concurrent partner connections	N/A	UC1	N/A

BR19	ML models must be retrained monthly with new fraud data	N/A	N/A	N/A
BR20	Partners can only view their own transaction data and alerts	AD2	UC2	UI 2.8 (Alerts)
BR21	Risk score scale: 0-25 (Low), 26-50 (Medium), 51-75 (High), 76-100 (Critical)	AD1	UC1	UI 2.8 (Dashboard)
BR22	Device fingerprints are hashed before storage	AD1, AD4	UC1, UC4	N/A
BR23	System must maintain 99.9% uptime (excluding scheduled maintenance)	N/A	All	N/A
BR24	API rate limiting: 500 requests per minute per partner	AD1	UC1	N/A

Section 4 – Domain Class

4.1 Overview

The domain model for Sentra represents the main entities involved that define the operations. It shows the interaction between the partners, users and internal components.

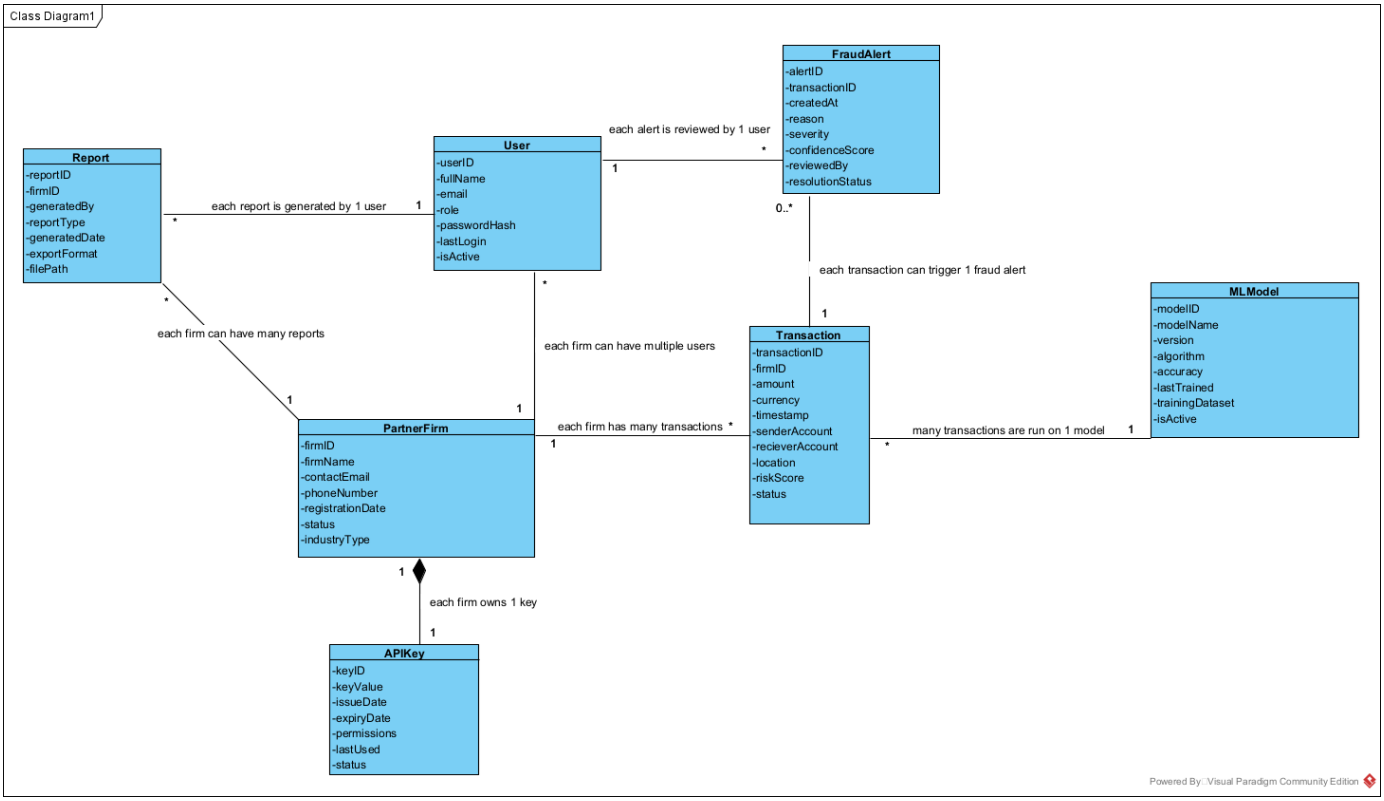
4.2 Class Descriptions

Class Name	Description
PartnerFirm	Represents a firm integrated with Sentra
User	Represents people within a partner firm (Admin, Analyst
APIKey	Secure auth for partner firms
Transaction	Represents financial transactions
FraudAlert	Generated when a transaction is flagged
Report	Represents a report for partner firms
MLModel	ML model used

4.3 Class Relationships

From	To	Type	Cardinality	Description
PartnerFirm	User	Association	1 -> *	Each partner firm can have multiple users
PartnerFirm	APIKey	Composition	1 -> 1	Each firm owns 1 Api key
PartnerFirm	Transaction	Association	1 -> *	Each firm sends multiple transaction
PartnerFirm	Report	Association	1 -> *	Each firm generates multiple reports
Transaction	FraudAlert	Association	1 -> 0..1	Each transaction may be flagged and make a fraud alert
Transaction	MLModel	Association	*-> 1	Each transaction is checked by 1 ML model
FraudAlert	User	Association	*->1	Each user is assigned multiple alerts
Report	User	Association	*->1	Each user generated multiple reports

4.4 Class Diagram



Section 5 – Database

5.1 Overview

Sentra requires a secure, reliable, and structured data storage solution to support fraud detection, alert generation, partner onboarding, and compliance reporting.
Given the regulatory environment (PCI DSS, GDPR) and the relational nature of entities such as Partners, Users, Alerts, and Transactions, a **Relational Database Management System (RDBMS)** is chosen.

The RDBMS ensures:

- ACID compliance
- Strong referential integrity
- Traceable audit logs
- Support for structured queries essential for fraud analytics
- Reliability for financial and regulated workloads

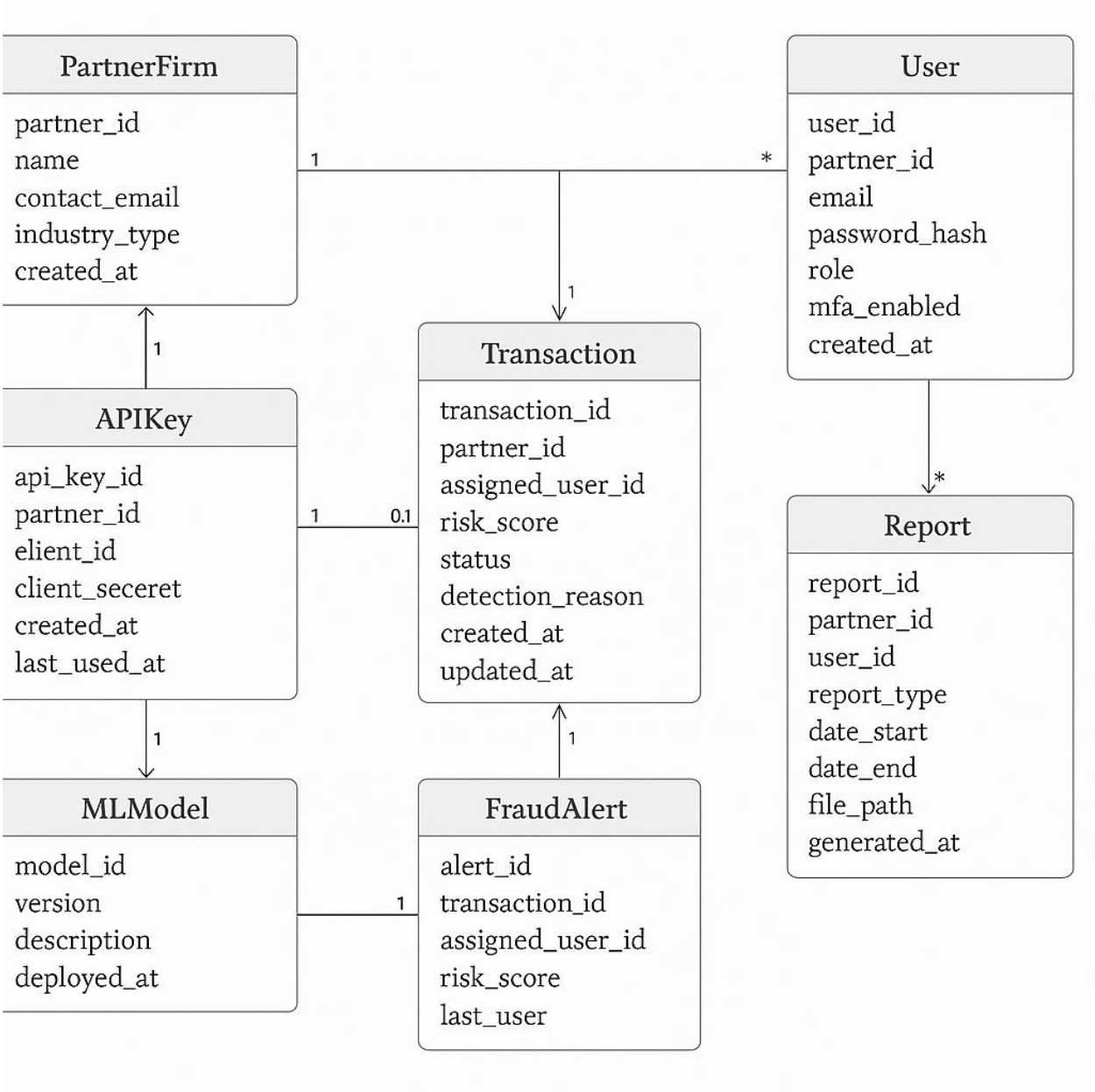
5.2 Entity Relationship Diagram (ERD)

ERD Description (Text-based for SRS)

The following describes each entity and their relationships:

- **PartnerFirm (1 →) User*
  Each partner has multiple users (Admin, Analyst, Developer).
- **PartnerFirm (1 → 1) APIKey**
  A partner holds one API key used for secure OAuth-based integration.
- **PartnerFirm (1 →) Transaction*
  Partners submit multiple transaction records for analysis.
- **Transaction (1 → 0..1) FraudAlert**
  A transaction may result in an alert if risk score exceeds threshold.
- *FraudAlert (→ 1) User**
  Alerts are assigned to one analyst or admin for investigation.
- **PartnerFirm (1 →) Report*
  Each partner can generate compliance and fraud summary reports.
- *Transaction (→ 1) MLModel**
  Every transaction is scored using the currently deployed ML model.

ERD Diagram (Structured Layout)



5.3 Data Dictionary

The following tables define Sentra's core database schema.

Table: PartnerFirm

Field Name	Type	Constraints	Description
partner_id	INT	PK, auto-increment	Unique ID for each partner firm
name	VARCHAR(150)	NOT NULL, UNIQUE	Registered company name
contact_email	VARCHAR(150)	NOT NULL	Primary contact for onboarding
industry_type	VARCHAR(100)	NULL	Type of fintech industry
created_at	DATETIME	NOT NULL	Timestamp of registration

Table: APIKey

Field Name	Type	Constraints	Description
api_key_id	INT	PK, auto-increment	Unique API key record
partner_id	INT	FK → PartnerFirm(partner_id)	Owner firm of this API key
client_id	VARCHAR(255)	NOT NULL, UNIQUE	OAuth client identifier
client_secret	VARCHAR(255)	NOT NULL	Encrypted OAuth secret
created_at	DATETIME	NOT NULL	Issuance timestamp
last_used_at	DATETIME	NULL	Last API usage

Table: User

Field Name	Type	Constraints	Description
user_id	INT	PK, auto-increment	User record ID
partner_id	INT	FK → PartnerFirm(partner_id)	Firm user belongs to
email	VARCHAR(150)	NOT NULL, UNIQUE	Login email
password_hash	VARCHAR(255)	NOT NULL	Hashed password
role	ENUM	('Admin','Analyst','Developer')	Role-based access
mfa_enabled	BOOLEAN	DEFAULT FALSE	MFA requirement for admins

Field Name	Type	Constraints	Description
created_at	DATETIME	NOT NULL	Account creation timestamp

Table: MLModel

Field Name	Type	Constraints	Description
model_id	INT	PK, auto-increment	ML model identifier
version	VARCHAR(50)	NOT NULL	Model version label
description	TEXT	NULL	Optional notes
deployed_at	DATETIME	NOT NULL	Timestamp when model was deployed

Table: Transaction

Field Name	Type	Constraints	Description
transaction_id	BIGINT	PK	Transaction unique ID
partner_id	INT	FK → PartnerFirm(partner_id)	Submitted by firm
user_ref_id	VARCHAR(100)	NULL	Internal partner-side user ID
amount	DECIMAL(12,2)	NOT NULL	Transaction amount
device_fingerprint	VARCHAR(255)	NOT NULL	Hashed device fingerprint
timestamp	DATETIME	NOT NULL	Time of transaction
risk_score	INT	NULL	ML + rule engine risk score
model_id	INT	FK → MLModel(model_id)	Model used for scoring
created_at	DATETIME	NOT NULL	When record was stored

Table: FraudAlert

Field Name	Type	Constraints	Description
alert_id	INT	PK, auto-increment	Alert identifier
transaction_id	BIGINT	FK → Transaction(transaction_id)	Triggering transaction
assigned_user_id	INT	FK → User(user_id)	Analyst handling the case
risk_score	INT	NOT NULL	Final risk score
status	ENUM	('Pending','Under Review','False Positive','Confirmed Fraud','Resolved')	Investigation status
detection_reason	TEXT	NULL	Model/rule explanations
created_at	DATETIME	NOT NULL	When alert was generated
updated_at	DATETIME	NULL	Last status update

Table: Report

Field Name	Type	Constraints	Description
report_id	INT	PK, auto-increment	Report identifier
partner_id	INT	FK → PartnerFirm(partner_id)	Firm requesting the report
user_id	INT	FK → User(user_id)	Officer who generated it
report_type	ENUM	('FraudSummary','AlertResolution','PartnerActivity')	Report category
date_start	DATE	NOT NULL	Start of reporting period
date_end	DATE	NOT NULL	End of reporting period
file_path	VARCHAR(255)	NOT NULL	Storage location for PDF/CSV

Field Name	Type	Constraints	Description
generated_at	DATETIME	NOT NULL	Timestamp of generation

5.4 Summary

The database structure supports:

- Real-time fraud scoring
- Secure partner integration
- Scalable and auditable alert handling
- Long-term regulatory reporting
- Cross-firm intelligence sharing

The RDBMS ensures **data integrity, security, compliance, and performance**, aligning with Sentra's requirements and industry standards.

Section 6 – Project Management

6.1 Work Breakdown Structure

The WBS for Sentra follows an industry common Agile work structure. Instead of traditional phase the phases are broken down into deliverables, user stories and detailed items that reflect how the system is going to be developed step by step.

Level 1 Overview – Main Goal

Sentra is a centralized Fraud Detection platform for Fintech Firms and small stage Startups

Level 2 Overview – Major Deliverables

- 1.1 - Fraud Detection Engine
- 1.2 - Alert Management System
- 1.3 - Partner Firm Integration
- 1.4 - Compliance And Reporting
- 1.5 - User and Roles
- 1.6 - Security and Auth
- 1.7 - Dashboard and Monitoring
- 1.8 - Deployment and Config

Level 3 Overview – Sub Features

1.1 - Fraud Detection Engine

- 1.1.1 - As a system, I can analyze incoming transactions in real time
- 1.1.2 - As a system, I can assign risk scores to each transaction
- 1.1.3 - As a system, I can decide whether a transaction is fraudulent and should be flagged

1.2 - Alert Management System

- 1.2.1 - As an analyst, I can view fraud alerts
- 1.2.2 - As an analyst, I can update alert statuses

1.3 - Partner Firm Integration

- 1.3.1 - As a partner firm, I can submit transactions using an API
- 1.3.2 - As a partner, I will receive fraud analysis results

1.4 - Compliance And Reporting

- 1.4.1 - As a user, I can generate a fraud report
- 1.4.2 - As a user, I can export a report in CSV or PDF format

1.5 - User and Roles

- 1.5.1 - As an admin, I can create and manage user accounts
- 1.5.2 - As an admin, I can assign roles

1.6 - Security and Auth

- 1.6.1 - As a secure system, I can authenticate partner firms with API keys
- 1.6.2 - As a secure system, I can secure access using encryption

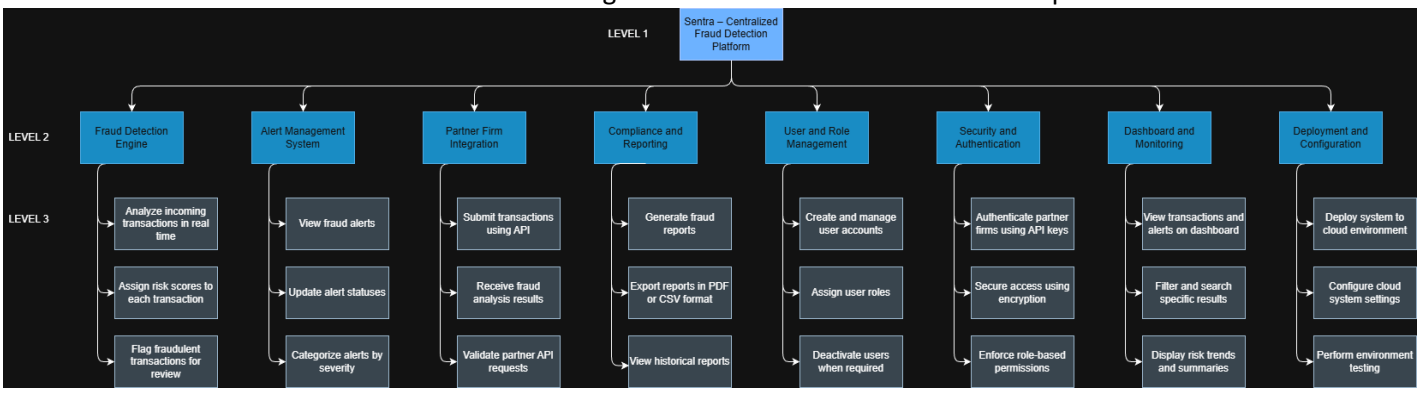
1.7 - Dashboard and Monitoring

- 1.7.1 - As a user, I can view transactions and alerts on a dashboard
- 1.7.2 - As a user, I can filter results and search specifics

1.8 - Deployment and Config

- 1.8.1 - As a developer, I can deploy the system to the cloud environment
- 1.8.2 - As a developer, I can configure the system settings on the cloud.

The visualization of the WBS is in a hierarchical diagram to demonstrate the relationnships



6.2 Milestones & Acceptance Criteria

Milestones are aligned with the Level 3 WBS items and represent the deliverables. Each milestone is evaluated by a real role.

Milestone	Related WBS	Milestone Title	Description	Acceptance Criteria	Evaluated By
M1	1.1.1 - 1.1.3	Fraud Detection Engine Completion	Core fraud detection logic is implemented and works, including risk scores and flagging	System successfully analyzes live test transactions and assigns risk scores and successfully flags fraudulent transactions	Lead AI Engineer
M2	1.2.1 - 1.2.3	Alert Management Module Deployment	Alert system allowing reviews updating and categorization is implemented	Alerts are generated and displayed correctly; severity and status are also displayed and saved to database	Fraud Operations
M3	1.3.1 - 1.3.3	Partner API Integration	Secure API endpoints are developed and allow partner firms to submit transactions	API accepts authenticated requests and processes the data and returns the fraud scores	Backend Engineer
M4	1.4.1 - 1.4.3	Reporting and Compliance Module Completion	Compliance system supports reports generations and exporting	Reports are generated correctly and exportable in PDF or CSV	Compliance Officer
M5	1.5.1 - 1.5.3	User and Role Management Module Implementation	Admin tools for user creation and modification are operational	Admin users can create update and delete users and assign new roles	Product Manager
M6	1.6.1 - 1.6.3	Security Framework Enforcement	Security such as authentication and API authorizations are active	API requests are authenticated; the data is encrypted and unauthorized access is blocked	Security Engineer
M7	1.7.1 - 1.7.3	Dashboard Functionality Completion	Interactive dashboard for monitoring	Real time alert data and summary and fraud metrics are displayed with	UX/UI Designer

			transactions and alerts is active	filters and search functions	
M8	1.8.1 - 1.8.3	System Deployment and Testing	Full system deployed to cloud environments and tested for stability and performance	Platform operates without any failures or downtimes	DevOps Engineer

This breakdown and milestone mapping ensures that the project will progress in steps and maintain deadlines and alignment with business rules.

Section 7 – Product Backlog & Implementation Schedule

7.1 - Overview

This section shows the Hybrid agile product backlog for Sentra including the user stores, acceptance criteria's, WBS items and priority ranking.

7.2 - Product Backlog

ID	User Story Title	Description	Acceptance Criteria	WBS	Priority	Sprint
US1	Real Time Fraud Detection	Partner submits transaction -> return risk score	Returns risk score and logs transaction	1.1.1-1.1.3	1	1
US2	Alert Generation	System creates alerts for flagged transactions	Alerts stored, correct fields	1.2.1	2	2
US3	View Alerts	Analyst views alert list	Sorting + filtering works	1.2.1	3	2
US4	Update alert status	Analyst changes status	Saved + audit logged	1.2.2	4	3
US5	Intelligence Sharing	Store anonymized patterns	Data anonymized; stored	1.1.3	5	3

US6	Partner API	Secure API for partners	OAuth, valid JSON, errors	1.3.1-1.3.3	6	4
US7	Risk Score Response	Return risk + reasons	Correct JSON format	1.3.2	7	4
US8	Compliance Reports	Export PDF/CSV	Report generates correctly	1.4.1-1.4.3	8	5
US9	User & Roles	Admin manages accounts	CRUD works, RBAC enforced	1.5.1-1.5.3	9	6
US10	Auth & Security	Login + encryption	MFA admin, TLS 1.3	1.6.1-1.6.3	10	6
US11	Dashboard Metrics	Real-time charts	Auto-refresh 30s	1.7.1	11	7
US12	Dashboard Filters	Filter alerts/tx	Filters work	1.7.2	12	7
US13	Partner Onboarding	Register new firm	API keys generated	1.5.1	13	8
US14	Deployment	Deploy to cloud	App runs via HTTPS	1.8.1	14	9
US15	Config Settings	Adjust thresholds	Saved + applied	1.8.2	15	9
US16	Shared Intelligence Use	Use shared patterns in scoring	Score adjusts correctly	1.1.3	16	10
US17	ML Model Update	Upload new model	New model used	1.1.2	17	10

7.3 - User Stories

- US1: As a partner system, I want to submit transactions and receive a risk score so that I can block fraud instantly.
- US2: As the system, I want to create alerts for risky transactions so analysts can review them.
- US3: As an analyst, I want to view fraud alerts so I can investigate quickly.
- US4: As an analyst, I want to update alert statuses so I can track investigations.
- US5: As a partner, I want to share anonymized fraud patterns, so other firms benefit.
- US6: As a developer, I want secure APIs so we can integrate easily.
- US7: As a developer, I want risk scores with reasons, so our system understands decisions.
- US8: As a compliance officer, I want reports so I can meet regulations.
- US9: As an admin, I want to manage users, so the system stays secure.
- US10: As a secure system, I want authentication and encryption, so data stays protected.
- US11: As an analyst, I want real-time dashboard metrics so I can monitor trends.
- US12: As an analyst, I want dashboard filters so I can find specific alerts.
- US13: As an admin, I want to register partners so they can access the system.
- US14: As a developer, I want to deploy the app so partners can use it.
- US15: As an admin, I want to configure settings, so detection adjusts to policies.
- US16: As the system, I want to use shared intelligence to improve accuracy.
- US17: As an admin, I want to upload new ML models, so detection improves.

Sprint Schedule

Sprint	Features
1	US1
2	US2 US3
3	US4 US5
4	US6, US7
5	US8
6	US9 US10
7	US11 US12
8	US13
9	US14 US15
10	US16 US17

Section 8 – Client/Faculty Sign-off