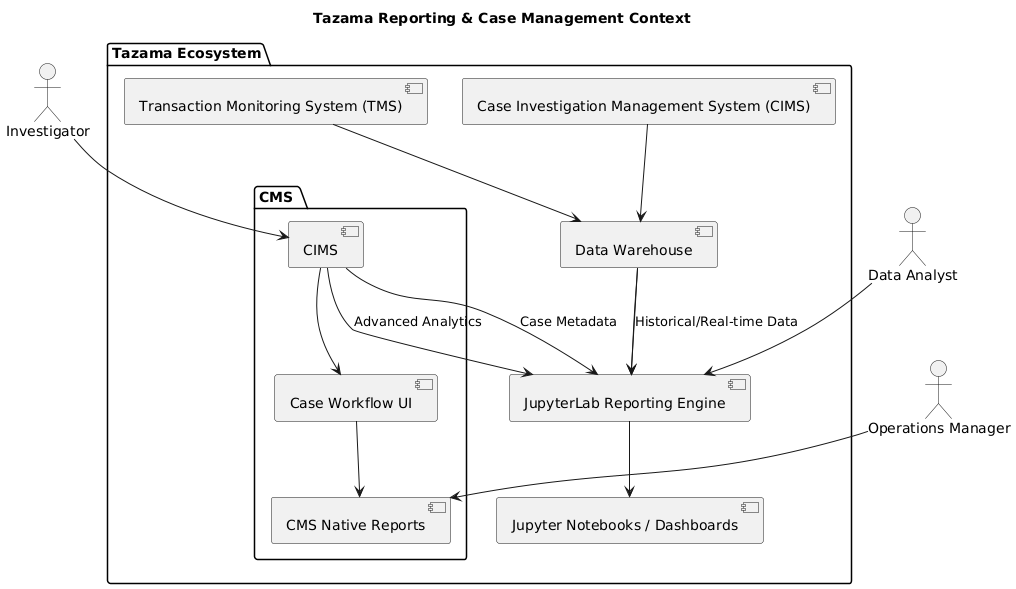
**BUSINESS REQUIREMENT SPECIFICATIONS DOCUMENT**

**BUSINESS INTELLIGENCE, ANALYTICS AND REPORTING**

| **#** | **Requirement** | |
| --- | --- | --- |
| 1. **Analytics Environment** | | |
| A.1 | Jupyter Lab will serve as the primary tool for performing data analysis and generating business reports. | |
| A.2 | Ensure flexibility in the Tazama analytics architecture by maintaining a loosely coupled design between Jupyter Lab and the broader system architecture. ( please refer to the image attached below explaining Tazama Analytics Architecture) | |
| 1. **Primary Data Source for Analytics** | | |
| **B.1** | All data streamed into the data warehouse will be used as the primary source for reporting  ***Assumption:*** *reporting will be done on data that is archived to the DWH.* | |
| **(C) Types of Data for Analytics** | | |
| **C.1** | The data type for analysis includes structured, semi-structured and unstructured data.  ***Structured Data Examples:*** Transaction data, Customer information, Rule engine logs, Case management system.  ***Semi-Structured Data Examples:*** Data stored in JSON, XML format e.g. Fraud Alerts, Email logs, Transaction Notes etc.  *Unstructured* ***Data:*** Images, screenshots, free form text e.g. screenshots submitted by the customer, email sent to the client, call center transcription (recording between customer and support staff related to suspected fraudulent activity)  ***Note:*** *Analytics will only be possible for semi-structured and unstructured data if there are any libraries available for it.* | |
| **(D) Analysis and Dashboards** | | |
| **D.1** | | Two distinct types of dashboards will be implemented within the system: a Management Dashboard for strategic oversight and decision-making, and an Operational Dashboard for monitoring and day-to-day operational management and Detection Performance Dashboards focused on measuring the effectiveness of the detection system. |
| **D.2(a)** | | The development of the following dashboards shall be within the scope of this project. These dashboards will provide authorized users with visibility into key metrics, insights, and system performance indicators relevant to their role and organizational context.  Access Control Requirements:   * The dashboard access shall be role-based, implemented via a Role-Based Access Control (RBAC) mechanism. * Keycloak shall be integrated for authentication and authorization, ensuring centralized identity management and secure access enforcement. * The system must respect multi-tenancy isolation—users (e.g., managers) associated with one tenant must not be able to access or view data belonging to another tenant under any circumstances. * Dashboard visibility and accessible metrics shall be dynamically filtered based on the user's role and tenant context.   These controls are essential to maintaining data confidentiality, regulatory compliance, and operational integrity across tenants.  Management Dashboards:**Executive Overview Dashboard**  **Functional Role Narrative:**  *A Chief Risk Officer or Executive Manager accesses the dashboard to gain real-time visibility into system-wide fraud metrics, allowing for oversight without engaging with detailed operational data.*  **Key Metrics**   * Total fraud cases detected * Fraud prevention rate (% of potential fraud stopped before completion) * Total fraud losses vs. recovered funds * Number of critical incidents escalated   **Fraud Trend Analysis Dashboard**  **Functional Role Narrative:**  *A Senior Fraud Analyst or Risk Strategy Lead accesses this dashboard to identify macro-level fraud patterns and shifts in typologies over time. These insights inform rule adjustments, typology reviews, and strategic mitigation planning.*  **Key Metrics**   * Historical fraud detection rates (quarterly or annual view) * Geographic/regional fraud distribution * Volume of fraud incidents by product, service line, or channel * Growth rate of new or recurring fraud types * Average time to detection over long intervals |
| **D.2(b)** | | Operational dashboards  **Incident Management and Case Tracking Dashboard**  **Functional Role Narrative:**   *An Investigator or Case Manager accesses this dashboard to monitor open cases, track resolution times, and manage task workflows. It ensures that no fraud case is left unresolved, and that escalation procedures are triggered for high-priority incidents.*  **Key Metrics**   * Total number of open fraud cases * Average and median case resolution time * Distribution of case types (e.g., account takeover, synthetic identity fraud, payment fraud) * Case priority levels and current escalation status * Investigator workload (cases per analyst)   **Fraud Alert Resolution Dashboard**  **Functional Role Narrative:**  *An Alert Handler or Fraud Operations Supervisor accesses this dashboard to track the number of active alerts, monitor resolution timelines, and assess alert quality. It supports effective workload distribution and timely response to high-severity fraud alerts.*  **Key Metrics**   * Total number of unresolved fraud alerts * Average and maximum time to resolve alerts * Fraud alert resolution rate (%) * Number and percentage of false positives * Distribution of alerts by severity or typology   **System Performance Dashboard**      **Functional Role Narrative:**  *A System Administrator or Infrastructure Lead uses this dashboard to track key performance indicators such as transaction processing time and system throughput. This allows proactive identification of slowdowns, outages, or capacity issues that may affect fraud detection accuracy or timeliness.*  **Key Metrics**   * Transaction Processing Time (average and percentile-based) * System Throughput (transactions processed per second) * Uptime/Downtime Statistics (for key services: TMS, CMS, Analytics, API Gateway)   Error Rate (number of failed or rejected messages)**Error and Failure Tracking Dashboard**  **Functional Role Narrative:**  *A DevOps Engineer or Technical Support Lead uses this dashboard to detect and investigate recurring failures, validate system stability, and ensure critical issues are addressed before impacting users or detection workflows.*  **Key Metrics**   * Total number of system errors and failures (daily/weekly/monthly) * Number of failed transactions (e.g., due to invalid data, timeouts, or authentication issues) * Error rate per N number of transactions * Frequency and pattern of repeated exceptions * Time to acknowledge and resolve critical failures   **API Response Time Dashboard**  **Functional Role Narrative**  *An Integration Engineer or System Administrator uses this dashboard to assess API performance metrics, detect latency issues, and receive alerts on failed or degraded API connections. This ensures uninterrupted access to critical third-party data during fraud evaluation.*  **Key Metrics**   * API Response Time (average, median, Nth percentile) * API Request Volume (requests per minute/hour) * API Call Success Rate (percentage of successful responses) * API Error Rate (failed responses per N number of requests) |
| **D.2(c)** | | Performance Dashboards**False Positive Analysis Dashboard** **Functional Role Narrative**:  *As a Fraud Analyst, I want to monitor the number of false positive alerts generated by the system so that I can identify overly aggressive rules or models and optimize detection accuracy.*  **Key KPIs**:   * False Positive Rate (FPR) * Total Alerts vs. Confirmed Cases * Alerts per Typology/Rule/Model * Average Time Wasted on False Positives  **False Negative Tracking Dashboard** **Functional Role Narrative**:  *As a Risk Officer or Model Governance Lead, I want to track cases of fraud that were not detected by the system but later confirmed, so I can evaluate detection coverage gaps.*  **Key KPIs**:   * False Negative Rate (FNR) * Missed Fraud Cases (absolute count) * Detection Coverage Ratio * Time from Transaction to Discovery   *Note: False negatives can only be tracked if the missed fraud cases are later discovered (e.g., through customer complaints, chargebacks, or manual review). The dashboard will report only* ***known or confirmed missed fraud****, not all undetected cases.If exact analyst review time is not tracked, this metric will be shown as an estimated average using standard review time assumptions or benchmarks.* **Precision & Recall Performance Dashboard** **Functional Role Narrative**:  *As a Data Scientist or ML Engineer, I want to view precision, recall, and F1 scores for each rule and model so I can assess detection effectiveness and adjust model thresholds if needed.*  **Key KPIs**:   * Precision (% of alerts that were actual fraud) * Recall (% of actual fraud that was detected) * F1 Score (harmonic mean of precision and recall) * Typology-wise model performance  **Rule/Model Effectiveness Dashboard** **Functional Role Narrative**:  *As a Detection Strategy Analyst, I want to evaluate how individual detection rules or ML models are performing, so I can deactivate underperforming rules and prioritize those with higher effectiveness.*  **Key KPIs**:   * Detection Accuracy per Rule/Model * Alert-to-Case Conversion Rate * Rule/Model Contribution to Fraud Detections * Volume of Alerts Triggered per Rule/Model |
| **(E) Artificial Intelligence with Rules and Typologies** | | |
| **E.1** | | The system shall provide an integrated AI/ML-assisted capability within the JupyterLab environment to support fraud detection and anomaly identification workflows. This functionality shall enable users to apply machine learning techniques over historical datasets in a user-friendly and guided manner.  While the final implementation approach is subject to further exploration and collaboration. |
| **E.2** | | The system shall provide the capability to identify and define new fraud detection rules and typologies that can be added to the Tazama fraud management system to enhance its detection capabilities. These rules and typologies will help capture new fraudulent activities as they emerge. |
| **E.3** | | The system will utilize an AI agent-based approach to discover and evaluate the performance of newly identified rules and typologies by continuously assessing their effectiveness in detecting fraudulent transactions. Rule discovery will be achieved through a combination of AI/ML-driven anomaly detection, leveraging historical tagged data (such as alert investigation outcomes and customer reports) to identify emerging fraud patterns. These patterns will be used to engineer new rules and typologies. The evaluation of these newly implemented rules will involve validating their performance against a baseline dataset, ensuring that the rules improve the detection accuracy by reducing False Positives and False Negatives while maintaining high True Positive and True Negative rates. As part of the ongoing process, the system will adapt to evolving fraud patterns by updating both the data and model baselines, ensuring the rules remain effective over time. The evaluation will also test the new model against both the updated data and the previous data baseline to ensure that the new model performs well for both new and existing fraud patterns. |
| **E.4** | | **Assumption**: Machine learning, model training and fine tuning will be done on the data present in the data warehouse. This assumption will be validated during the design phase |

**A.2: Tazama Analytics Architecture:**



This diagram provides a high-level architectural view of how reporting, case management, and analytics components are organized within the Tazama ecosystem, and how different user roles interact with these components.

#### Key Components:

* Transaction Monitoring System (TMS):  
   The core engine responsible for ingesting and evaluating transactional data in real time. It feeds evaluated data and detection results into the Data Warehouse.
* Case Investigation Management System (CIMS):  
   Manages the end-to-end investigation of flagged transactions. It stores case metadata and interacts with both native CMS interfaces and external analytics tools.
* Data Warehouse:  
   Acts as the central data repository for storing historical and real-time transaction data, case outcomes, and fraud detection metadata. It serves as the primary data source for reporting and analytics.
* CMS :  
   A platform used to manage case progression, investigator assignments, and generate operational reports.
  + Includes the Case Workflow UI for interactive investigation.
  + Supports CMS Native Reports for basic case tracking and metrics.
* JupyterLab Reporting Engine:  
   An embedded or integrated analytics layer that allows analysts and system stakeholders to perform advanced analytics and generate interactive dashboards. It supports:
  + Querying of historical data
  + Access to enriched case metadata
  + Development of custom Jupyter Notebook Dashboards