

## **Etapas 3 – Justificación Técnica**

### **1. Enfoque General**

La arquitectura propuesta responde a los desafíos de una institución financiera moderna que requiere **procesamiento mixto de datos (streaming y batch)**, **detección de fraude en tiempo casi real**, y **analítica avanzada de clientes**.

Se adopta un diseño **serverless y desacoplado**, aprovechando los servicios nativos de **Google Cloud Platform (GCP)** para garantizar **escalabilidad automática, alta disponibilidad, seguridad granular y control de costos**.

### **2. Justificación de Componentes**

#### **a. Ingesta de Datos**

- **Pub/Sub** Recibe eventos de transacciones de tarjetas en tiempo casi real, soportando alto throughput y baja latencia (<50 ms). Su modelo publish/subscribe desacopla productores y consumidores, garantizando resiliencia en alto tráfico.
- **Cloud Storage** actúa como landing zone para cargas batch diarias (maestros y catálogos), ofrece durabilidad y versionamiento a bajo costo.
- **Cloud Composer (Airflow)** orquesta las cargas y validaciones de integridad, permitiendo auditoría y dependencias entre tareas ETL.

#### **b. Procesamiento de Datos**

- **Dataflow (Apache Beam)** procesa flujos en streaming y batch en un entorno unificado. Se justifica por su **autoscaling, tolerancia a fallos y compatibilidad con ventanas temporales** necesarias para analizar patrones de fraude.
- Los pipelines incluyen **validaciones de esquema y reglas de calidad**, así como **enriquecimiento** con datos de clientes y comercios desde BigQuery.
- Los errores se canalizan a un **DLQ (Dead Letter Queue)** en Pub/Sub, cumpliendo el principio de líneas de defensa en calidad de datos.

#### **c. Almacenamiento y Analítica**

- **BigQuery** centraliza los datos en capas raw, staging, curated y bi, ofreciendo almacenamiento columnar con separación entre cómputo y datos.
- **BI Engine y Looker Studio** permiten consultas interactivas sobre dashboards ejecutivos de KPIs de fraude, operaciones y segmentación de clientes.
- La arquitectura soporta partitioning por fecha y clustering por customer\_id, optimizando desempeño y costo de consulta.

#### **d. Machine Learning (ML)**

- **Vertex AI** gestiona el ciclo del modelo: entrenamiento, registro y despliegue.
  - **Workbench / AutoML** acelera la experimentación con datasets en BigQuery.
  - **Model Registry** garantiza trazabilidad y versionamiento.
  - **Endpoints online** ofrecen inferencia con latencia P95 < 200 ms, habilitando alertas en tiempo casi real.

- Esta integración asegura coherencia entre training y serving mediante el uso de un Feature Store (opcional).

#### e. Integración y Consumo de Resultados

- **Cloud Run** implementa un microservicio stateless que orquesta solicitudes al endpoint de Vertex AI y publica resultados o alertas de fraude en Pub/Sub.
- Este diseño **minimiza la latencia de red**, facilita despliegues continuos y expone un servicio REST seguro mediante IAM y políticas de red (VPC Service Controls).

#### f. Gobierno, Seguridad y Operación

- **Cloud IAM** aplica el principio de mínimo privilegio segmentando dominios (Fraude, BI, ML).
- **Cloud KMS** encripta datos en tránsito y reposo.
- **Data Catalog** documenta linaje y metadatos, fortaleciendo trazabilidad y gobierno.
- **Cloud Logging / Monitoring** genera las métricas y alertas proactivas.
- **Budgets & Alerts** permiten controlar y prevenir el gasto operativo.

### 3. Flujo de Datos y Latencias

- Flujo Streaming (fraude):  
*Pub/Sub → Dataflow → Vertex AI Endpoint → BigQuery (fraud\_scoring) → Pub/Sub (alertas)*
  - Latencia end-to-end:  $\leq 500$  ms (P95).
- Flujo Batch (analítica):  
*Cloud Storage → Composer → BigQuery → Looker Studio*
  - Procesamiento diario con SLA D+1.
- Cada paso incluye validaciones automáticas, control de esquema y métricas.

### 4. Riesgos y Mitigaciones

Riesgo	Impacto	Mitigación
Altas de transacciones	Aumento de latencia o pérdida de eventos	Autoscaling en Pub/Sub y Dataflow; DLQ para reprocesos
Cambios de esquema en fuentes	Fallas en ingesta	Validación en Dataflow y versionamiento en BigQuery
Data drift del modelo de fraude	Degradación de precisión	Retraining periódico en Vertex AI; monitoreo de drift
Exceso de costos por consultas BI	Costo no controlado	Particionamiento, alertas de presupuesto y caching con BI Engine
Accesos no autorizados	Riesgo de fuga de datos	IAM granular, KMS, auditorías y VPC Service Controls

### 5. Beneficios Clave

- **Latencia controlada:** detección de fraude en menos de 500 ms.
- **Elasticidad y resiliencia:** servicios serverless escalan automáticamente.
- **Gobierno robusto:** trazabilidad, auditoría y control de accesos.
- **Eficiencia operativa:** automatización completa de pipelines.
- **Costo optimizado:** pago por uso y control presupuestario granular.