

**Nombre:** Carlos Pazmiño Zambrano

**Carrera:** Desarrollo de Software

## **INFORME TÉCNICO: CONSTRUCCIÓN DE MODELO SEMÁNTICO Y ARQUITECTURA LAKEHOUSE**

### **INTRODUCCIÓN**

Este proyecto consiste en la implementación de una arquitectura Lakehouse moderna para el procesamiento y análisis de datos transaccionales financieros. Originalmente planeado para Microsoft Azure, el proyecto fue migrado exitosamente a Amazon Web Services debido a restricciones de créditos en la plataforma Azure. La solución implementa el patrón Medallion Architecture (Bronze-Silver-Gold) combinando la escalabilidad de AWS S3 con las capacidades analíticas de Microsoft Fabric y Power BI, creando un modelo semántico optimizado para business intelligence y análisis avanzado.

### **OBJETIVO DEL PROYECTO**

#### **Objetivo General**

Diseñar e implementar una arquitectura de datos escalable que transforme datos transaccionales crudos en un modelo semántico de alta calidad, permitiendo análisis avanzado y reporting en tiempo real mediante Power BI.

#### **Objetivos Específicos**

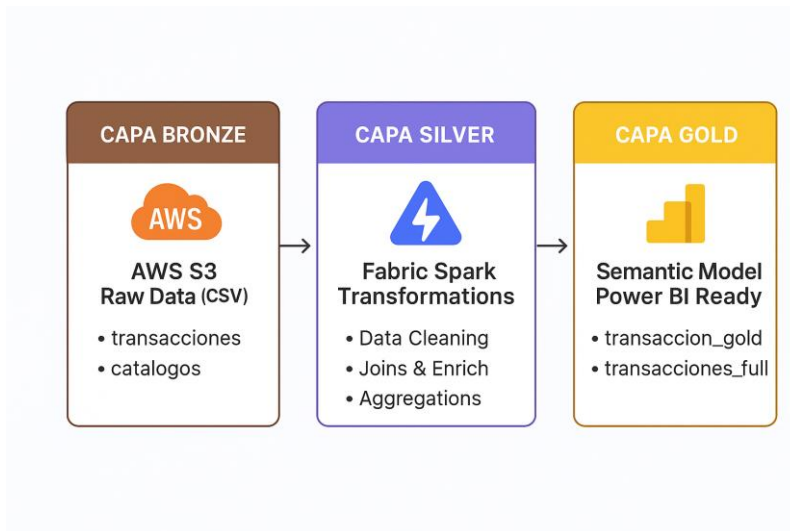
- Implementar el patrón Medallion Architecture en un entorno Lakehouse
- Realizar la transfiguración de datos crudos a modelos analíticos listos para consumo
- Crear un modelo semántico optimizado para Power BI Direct Lake
- Establecer procesos ETL/ELT robustos usando PySpark
- Habilitar capacidades de análisis geográfico y temporal sobre datos transaccionales

### **OBJETIVO PRINCIPAL**

**Transformar datos transaccionales financieros dispersos en CSV en un modelo semántico unificado y enriquecido que sirva como única fuente de verdad para el análisis de negocio, reporting ejecutivo y toma de decisiones estratégicas en la organización.**

## ARQUITECTURA DEL SISTEMA

### Diagrama de Arquitectura Lakehouse



### Componentes de la Arquitectura

#### 1. Capa Bronze (AWS S3)

- **Almacenamiento:** Amazon S3 como data lake
- **Formato:** Archivos CSV crudos
- **Tablas fuente:**
  - `transacciones.csv` - Datos transaccionales brutos
  - `catalogo_agencias.csv` - Dimensiones de agencias con geolocalización
  - `catalogo_tipo_transaccion.csv` - Catálogo de tipos de transacción

#### 2. Capa Silver (Microsoft Fabric + Spark)

- **Procesamiento:** Motor Spark en Fabric
- **Transformaciones:**
  - Limpieza y estandarización de datos
  - Joins entre tablas transaccionales y dimensionales
  - Enriquecimiento con metadatos geográficos

- Normalización de esquemas

### 3. Capa Gold (Modelo Semántico)

- **Destino:** Tablas optimizadas para Power BI
- **Modo:** Direct Lake para máxima performance
- **Productos:**
  - transaccionalidad - Vista agregada para análisis
  - transacciones\_full - Datos detallados completos

## LIBRERÍAS Y TECNOLOGÍAS UTILIZADAS

### Python & PySpark

python

# Procesamiento de Datos

pyspark.sql - Transformaciones y agregaciones Spark

pyspark.sql.functions - Funciones para manipulación de datos

# Conexión Cloud

boto3 - Cliente AWS S3 para Python

s3transfer - Manejo de transferencias S3

# Utilidades

pandas - Procesamiento adicional de datos

io.BytesIO - Manejo de streams de datos



### Cloud & Plataformas

- **AWS S3:** Almacenamiento object storage
- **Microsoft Fabric:** Plataforma de analytics unificada
- **Power BI:** Herramienta de visualización y BI

- **Direct Lake:** Modo de conexión high-performance

### **Librerías Específicas**

python

# Instalación y versión de librerías

boto3==1.40.69      # Cliente AWS para Python

botocore==1.40.69    # Librería core AWS

s3transfer==0.14.0    # Transferencias S3

jmespath==1.0.1      # Query language JSON

pyspark              # Procesamiento distribuido

### **Funciones PySpark Utilizadas**

python

from pyspark.sql.functions import (

    broadcast,      # Optimización de joins

    to\_date,        # Conversión de fechas

    col,            # Referencia a columnas

    count,         # Conteo de registros

    sum,            # Agregación suma

    count          # Conteo agregado

)

## **FLUJO DE PROCESAMIENTO IMPLEMENTADO**

### **Pipeline de Transformación**

1. **Ingesta:** Conexión a AWS S3 y carga de CSV
2. **Limpieza:** Eliminación de columnas redundantes, renombrado
3. **Enriquecimiento:** Joins con tablas maestras, agregación geográfica
4. **Modelado:** Creación de vistas semánticas para business intelligence

## 5. **Publicación:** Guardado en tablas para consumo Power BI

### **Métricas y KPIs Habilitados**

- Volumen transaccional diario por agencia
- Análisis de tendencias temporales
- Segmentación por tipo de transacción (Financiera/No Financiera)
- Distribución geográfica de operaciones
- Performance por canal transaccional

### **CONCLUSIÓN**

La implementación exitosa de esta arquitectura Lakehouse demuestra la viabilidad de soluciones híbridas que combinan lo mejor de múltiples plataformas cloud. A pesar del cambio no planificado de Azure a AWS, se logró mantener la integridad del diseño original y entregar un modelo semántico de alta calidad listo para impulsar la inteligencia de negocio en la organización.