UNIVERSIDAD EAFIT

ST0263: Tópicos Especiales en Telemática, 2025-1

Trabajo 3 – Automatización del proceso de Captura, Ingesta, Procesamiento y Salida de datos accionables para realizar la gestión de datos (Arquitectura Batch para big data)

Fecha de entrega: 02 de junio de 2025

Descripción:

Durante los diferentes temas vistos en la unidad 3, se ha podido evidenciar los retos que puede conllevar ejecutar cada una de las etapas del ciclo de vida de un proceso analítico conformado por la Captura de datos en la Fuente, la Ingesta, el Almacenamiento, el Procesamiento y los resultados.

Muchos de estos procesos, los realizamos durante el curso con procesos manuales, pocos datos y demos muy básicos que nos alejan de la realidad de los procesos de ingeniería de datos reales en las empresas.

Para esto, realizaremos un proyecto más cercano a un prototipo real de un caso de ingeniería de datos big data.

Como fuente de datos, utilizaremos los datos provistos en línea (mirar al final del documento), el acceso se puede hacer por archivos o por APIs (explorar ambos casos). Además, contaremos con una base de datos relacional real, tipo MySQL o Postgres, que contendrá datos simulados que se puedan requerir para completar el análisis de datos de los datos seleccionados. Esto nos permitirá al menos experimentar con 2 fuentes reales en las empresas (archivos en URLs o APIs y acceso a bases de datos).

A nivel de ingesta de datos, debemos crear algún proceso automático, que nos permita descargar los datos por Archivo (URL) y API, y almacenarlos en un Bucket S3 en la zona Raw. También debemos crear un proceso automático, para la extracción de los datos de la base de datos relacional mysql o postgress y almacenar los datos en un bucket S3 en la zona raw.

A nivel de procesamiento, utilizaremos un clúster de EMR con procesamiento en Spark para realizar 2 tareas:

- Automatizar procesos ETL mediante Steps en un clúster EMR con Spark. Los requerimientos de estos procesos ETL será realizar procesos de preparación de datos y unión de datos de XXX provenientes de la fuente de datos con Datos de la base de datos relacional. El resultado de los procesos ETL en Spark deberán ser

almacenados en un bucket S3 en la zona Trusted. Tanto el proceso de creación del clúster EMR como el procesamiento ETL deberán ser automáticos.

- Automatizar procesos de análisis, analítica o aprendizaje de máquina sobre los datos preparados en la zona Trusted. En principio, aplicar procesos analíticos sencillos como los descriptivos o exploratorios combinando los datos de la base de datos relacional. Los procesos de analítica de datos descriptivos deben ser implementados con dataframes pipelines y con SparkSQL. Los procesos de analítica avanzada deben utilizar dataframes pipelines con SparkML. El resultado del análisis de datos, deben ser enviados a un bucket S3 en la zona Refined. Estos resultados deben poder ser consultados de diferentes formas: Athena y API Gateways. Implementar ambos.
  Recuerde que todos los procesos deben ser automáticos y no manuales.

A nivel de aplicación, los resultados del proceso analítico deben poder ser consultados por Athena o vía una API, realizar el prototipo que corresponda para demostrar estas aplicaciones.

Algunos otros requerimientos y recomendaciones:

- La captura e ingesta de datos hacia S3 deberá ser automática sin intervención humana.
- La creación del clúster, las pruebas y el desarrollo las puede hacer manualmente, pero cuando ya esté listo el proyecto3 tanto la creación del clúster, la realización del procesamiento ETL y analítica, y los resultados programados en los 'Steps' del clúster deben ser automáticos sin intervención humana.
- La primera opción de implementación es con AWS Academy, aunque todos conocemos las limitaciones que puede tener para utilizar algunos servicios. Deberá buscar alternativas en AWS Academy sino cuenta con un servicio, sea creativo. SIN EMBARGO, si se siente con confianza y dominio de las mismas tecnologías que utilizaremos en AWS pero en otra nube como GCP o Azure, se invita y promueve su realización en estas otras nubes para implementar este mismo proyecto3. Se cuenta con créditos oficiales para GCP y Azure. La ventaja de estas últimas 2, es que podemos utilizar todos los servicios sin limitación de permisos, pero si de costo (Máx 50 USD).

A nivel de entregables:

- Repositorio github del proyecto3 donde estén todos los scripts, programas, instrucciones, documentación, etc, para replicar el proyecto.
- Un documento readme.md en el repositorio github del proyecto3.
- Una videosustentación del proyecto3
- Presentación y sustentación del proyecto3, activando todos los recursos, mostrando la automatización y exponiendo las diferentes etapas. Esta sustentación se realizará el día 03 de junio (de 8am a 12m), con turnos de 30 mins. Se enviará agenda con la citación. Debe ser presencial.

Referencias:

- Video y github de un caso sencillo de ejecución de Spark en EMR con Steps.
    - https://youtu.be/ZFns7fvBCH4?si=hu5Y34JDB9yY7bsd
    - https://github.com/airscholar/EMR-for-data-engineers/tree/main

Fecha de entrega máxima: 02 de Junio de 2025 23:59 por buzón de Interactiva virtual

Fuentes de datos:

Cada equipo de trabajo debe explorar y seleccionar los datos insumo para el proyecto 3 de las siguientes fuentes en línea y gratuitas, que le permita definir una problemática concreta que quiere analizar con alguno de las siguientes fuentes de datos (recuerden que el requisito es tener datos en línea por una API y datos históricos cargados desde bases de datos, o en última instancia desde archivos datasets):

1. Datos del tiempo en línea y datos históricos de cualquier parte del mundo

Open-Meteo

Url: https://open-meteo.com
Acceso: API gratuita (sin autenticación)
Variables: clima actual, pronóstico, históricos
Ideal para carga masiva en S3
Ej:
https://archive-api.open-meteo.com/v1/archive?latitude=6.25&longitude=-75.56&start_date=2022-01-01&end_date=2022-12-31&daily=temperature_2m_max,precipitation_sum&timezone=America/Bogota

WeatherAPI

Url: https://www.weatherapi.com
API Key requerida (gratis limitado)
Variables: clima actual, pronóstico, históricos hasta 2010
formatos: CSV/JSON/XML

Meteostat

Url: https://dev.meteostat.net/
Acceso: API gratuita (requiere token)
Datos históricos desde 1973
Variables: temperatura, viento, nubosidad, precipitaciones, presión, históricos desde 1973
Formato: JSON

2. Transporte y Movilidad

Datos Abiertos de TransMilenio

URL: https://datosabiertos.transmilenio.gov.co/
Datos de estaciones, rutas, horarios, validaciones
Formato: CSV y APIs JSON
Uso: cruzar con base de datos simulada de usuarios o incidentes

OpenTraffic (Uber Movement)

URL: https://movement.uber.com/
Datos de velocidad y tráfico por ciudad (requiere inscripción)
Uso: análisis de movilidad urbana y predicción de congestión

3. Datos financieros:

Datos de la Superintendencia Financiera de Colombia

Url: https://www.superfinanciera.gov.co
Formato: CSV, Excel, JSON (con algunas APIs disponibles)
Ejemplo: precios históricos de acciones, tasas de interés, datos de bancos
Uso: integración con bases de datos relacional simulada de clientes e inversiones

Banco Mundial - Open Data

URL: https://data.worldbank.org/
Acceso por archivo y API
Ejemplo: PIB, desempleo, acceso a internet por país
Uso: análisis comparativo internacional o regional

4. Salud publica

Datos abiertos del Ministerio de Salud Colombia

URL: https://www.datos.gov.co/Salud-y-Protecci-n-Social
Ejemplo: vacunación COVID, morbilidad, atención EPS
Formato: CSV, JSON
Uso: análisis epidemiológico por región, cruzado con base de pacientes

5. Datos sobre ecommerce

MercadoLibre Public APIs

Url: https://developers.mercadolibre.com.co
Acceso: API pública con autenticación (algunos endpoints sin token)
Datos disponibles:

- Productos por categoría o palabra clave
- Detalles de publicaciones (precio, título, vendedor)
- Comentarios y valoraciones

Uso educativo:

- Recolectar precios de productos por región o categoría
- Análisis de oferta y demanda

Fake Store API (simulación estilo Amazon)

Url: https://fakestoreapi.com
Acceso libre, sin autenticación
Datos disponibles:
Productos (título, categoría, precio, rating)
Carritos de compra, usuarios simulados
Útil para: prácticas de análisis, dashboards, simulación de compras
Ejemplo: https://fakestoreapi.com/products

Best Buy Developer API

Sitio web: https://developer.bestbuy.com/
API Key gratuita
Datos en tiempo real: catálogo de productos, precios, disponibilidad
Útil para: construir dashboards o análisis de precio por categoría