AIR ROUTES

Plataforma Escalable para Búsqueda de Vuelos Inmediatos

Javier González Benítez
Jorge González Benítez

Asignatura: Tecnologías de Servicios para Ciencia de Datos
Grado: Grado en Ciencia e Ingeniería de Datos
Universidad: Universidad de Las Palmas de Gran Canaria

Repositorio: github.com/javierglezbenitez/AirRoutes_TSCD

1 Descripción del Proyecto

Air Routes es una solución tecnológica diseñada para personas que necesitan encontrar vuelos rápida y eficientemente ante imprevistos. El proyecto simula un escenario real utilizando datos ficticios, pero con una arquitectura escalable, modular y orientada a la nube, basada en servicios AWS y tecnologías modernas.

Este sistema no solo procesa datos de vuelos, sino que también los organiza, los expone mediante una API y los presenta en una interfaz intuitiva. Todo esto con el objetivo de ofrecer consultas rápidas y confiables para usuarios que requieren tomar decisiones inmediatas.

2 ¿Qué hace la aplicación?

- Procesa datos de vuelos (origen, destino, duración, precio, aerolínea, etc.).
- Almacena y consulta información en una base de datos orientada a grafos (Neo4j).
- Expone una API REST para consultas dinámicas.
- Visualiza resultados mediante una interfaz web intuitiva.

3 Arquitectura del Sistema

El proyecto está compuesto por cinco módulos principales, diseñados para funcionar tanto en entornos locales como en AWS:

3.1 1. DATALAKE

El módulo Datalake es el punto de entrada para los datos crudos del sistema. Su función principal es almacenar toda la información relacionada con vuelos en su estado original, sin transformaciones. Esto permite mantener una copia íntegra de los datos para futuras consultas o reprocesamientos. Este módulo puede ejecutarse en dos modos:

- **Local:** Los datos se almacenan en el sistema de archivos del equipo, lo que facilita pruebas y desarrollo sin depender de la nube.
- **Remoto:** Utiliza Amazon S3 para almacenar los datos, garantizando escalabilidad, durabilidad y acceso distribuido.

3.2 2. DATAMART

El Datamart es responsable de transformar los datos crudos en información estructurada y lista para análisis. Cada día, este módulo elimina los datos del día anterior y carga únicamente los nuevos, asegurando que la información esté siempre actualizada. Para el almacenamiento y consulta, se utiliza Neo4j, una base de datos orientada a grafos que permite modelar relaciones complejas entre vuelos, aerolíneas y rutas. Modos de ejecución:

- **Local:** Se levanta un contenedor Docker con Neo4j, ideal para entornos de desarrollo.
- **AWS EC2:** Se crea una instancia EC2 que ejecuta Neo4j mediante Docker, ofreciendo mayor capacidad y disponibilidad.

3.3 3. ORCHESTRATOR

Este módulo actúa como el coordinador del sistema. Su función es ejecutar de manera simultánea los procesos del Datalake y el Datamart, asegurando que los datos fluyan correctamente desde su ingestión hasta su transformación. Además, gestiona la secuencia de ejecución y controla errores para mantener la consistencia del sistema.

3.4 4. API

La API es el puente entre los datos y los usuarios o aplicaciones externas. Está desarrollada en Spring Boot y se despliega en una instancia EC2 en AWS. Su función es exponer servicios REST que permiten realizar consultas sobre la información almacenada en Neo4j. La API se conecta directamente al servidor Neo4j, independientemente de si está en local o en la nube, y ofrece endpoints optimizados para búsquedas rápidas y eficientes.

3.5 5. GUI

La interfaz gráfica (GUI) es la capa de presentación del sistema. Genera páginas HTML dinámicas que permiten a los usuarios interactuar con la aplicación de forma sencilla. A través de la GUI, los usuarios pueden enviar consultas que son procesadas por la API y recibir resultados en tiempo real. Esta interfaz está diseñada para ser intuitiva y ofrecer una experiencia fluida.

Pruebas CI/CD Automáticas

Cada módulo cuenta con pruebas automáticas integradas en un flujo CI/CD. El proceso consiste en:

- Ejecutar tests unitarios y de integración en cada módulo tras cada cambio.
- Validar la correcta ejecución mediante pipelines configurados en GitHub Actions.
- Solo tras pasar todas las pruebas, se realiza el **push** al repositorio, asegurando calidad y estabilidad.

4 Tecnologías Utilizadas

- Java 17 (Spring Boot para la API)
- Neo4j (Base de datos orientada a grafos)
- Docker (Contenedores para despliegue local y en EC2)
- AWS Services:
  - S3 (Almacenamiento de datos)
  - EC2 (Instancias para API y Neo4j)
- HTML (Interfaz gráfica)
- GitHub Actions (Automatización CI/CD)

INFRAESTRUCTURA Air-Routes

Diseño de infraestructura escalable y distribuida, basada en AWS, que combina procesamiento en EC2, almacenamiento en S3 y herramientas externas para desarrollo.

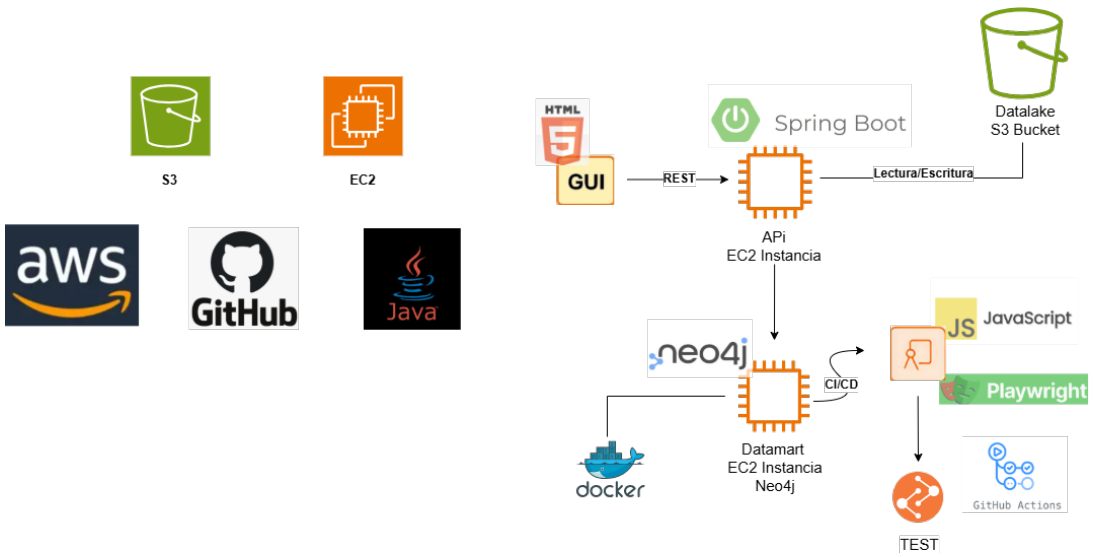

Figure 1: Infraestructura de Air-Routes.

5 Problemas Encontrados

Durante el desarrollo del proyecto se presentaron diversos retos que condicionaron la arquitectura y el flujo de trabajo:

- **Accesos denegados a servicios avanzados de AWS:** Algunas funcionalidades que podrían mejorar la arquitectura (como ECR, ECS Fargate, etc.) no pudieron ser utilizadas por restricciones de permisos.
- **Falta de fuentes de datos reales:** No se encontró un dataset real adecuado para la lógica de negocio, por lo que fue necesario simular datos de vuelos.
- **Problemas de compatibilidad y configuración:** Se presentaron errores al levantar contenedores en entornos locales y en EC2, principalmente relacionados con dependencias y versiones.

6 Próximos Objetivos

- Integración con datos reales de aerolíneas.
- Implementación de recomendaciones inteligentes.
- Optimización para dispositivos móviles.
- Mejorar la automatización CI/CD con despliegue continuo.

7 Conclusión

El desarrollo de este proyecto nos ha permitido adquirir habilidades clave para diseñar arquitecturas escalables y orientadas al negocio, integrando tecnologías modernas como AWS, Docker y Neo4j. Además, la implementación de flujos CI/CD nos ha enseñado la importancia de la automatización en entornos profesionales. Este trabajo ha sido fundamental para comprender cómo combinar infraestructura, desarrollo y procesos de calidad para ofrecer soluciones robustas y eficientes.