Instituto Tecnológico
y de Estudios Superiores de Occidente

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018, publicado en el Diario Oficial de la Federación del 29 de noviembre de 1976.

Departamento de Electrónica, Sistemas e Informática

Maestría en Sistemas Computacionales

Ingesta y Modelado de datos de aerolíneas mediante un pipeline utilizando tecnología disponible en la nube

---

TRABAJO RECEPCIONAL para obtener el **GRADO** de
MAESTRO EN SISTEMAS COMPUTACIONALES

Presenta: JORGE LUIS CABALLERO ZUÑIGA

Director: Dr. ALBERTO DE OBESO ORENDAIN

Tlaquepaque, Jalisco. Al 24 de Diciembre 2022.

AGRADECIMIENTOS

En primer lugar, a mis padres, a mi Madre Teresa y a mi Padre Edgard, por siempre estar conmigo y acompañarme siempre por medio de Dios, por sus oraciones y el apoyo el cual siempre me han brindado, por el aliento y soporte el cual siempre sé que tendré.

A mi esposa Marisol De Anda Navarro por su apoyo incondicional, por sus ánimos y por ser un ejemplo a seguir, por el amor y comprensión que nos tenemos mutuamente.

Agradecimiento especial a mi Dir. de TOG Alberto de Obeso Orendain por su sabiduría y enseñanzas las cuales me han ayudado mucho en el proceso de la elaboración de este TOG.

Agradecimiento especial al coordinador de la Maestría el Dr. Iván Villalón por su compromiso y apoyo.

Agradezco especial a mi profesor de IDI 1 Apolo Álvarez por su guía y conocimientos.

A mis mejores amigos Carlos, Alejandro, Sergio, Pablo, Alfredo, David, Fredy, Rubén, Javier, Cristóbal, por saber que siempre puedo contar con ustedes en las buenas y en las malas.

A mis compañeros y amigos de la Maestría con los cuales he convivido durante este proceso.

Agradecimientos especiales a CONACYT por el apoyo económico brindado con el número de la beca: 503278

Agradecimiento especial al ITESO por el apoyo económico y en todos los sentidos, para mí es como mi segunda casa.

DEDICATORIA

Se dedica esta tesis a todos los científicos de datos y a aquellos usuarios de un producto o servicio, como lo son las aerolíneas, los vuelos o cualquier producto en general.

Al ser un fiel creyente del Product management y la creación de productos de calidad y que los usuarios quieran, ya que puede haber productos de calidad que no generen ningún valor para los clientes, no les hagan la vida más fácil o no resuelvan alguna necesidad primaria, creo que para un producto de tecnología debemos de llegar al siguiente nivel en experiencia de usuario (UX) y una interfaz de usuario (UI) intuitiva y fácil de usar.

Creo firmemente en el Product management y en la creación de productos de calidad que satisfagan las necesidades y deseos de los usuarios. Sin embargo, es importante recordar que incluso los productos de alta calidad pueden no generar valor para los clientes si no facilitan la vida de las personas o no resuelven sus necesidades básicas. Por eso, en el caso de productos de tecnología, debemos esforzarnos por mejorar la experiencia de usuario (UX) y ofrecer una interfaz de usuario (UI) intuitiva y fácil de usar.

Además de eso la dedicatoria es para los creadores de productos que tienen que:

- Saber identificar problemas
- Necesidades de los consumidores
- Analizar el comportamiento de los usuarios
- Optimizar procesos
- Definir una visión del producto
- Dar una idea de lo que son las marcas
- Analizar las macrotendencias
- Definición de requerimientos basado en lo que los usuarios quieren
- Idear como monetizar esas oportunidades y esos productos

Igualmente, este trabajo está dedicado a los científicos de datos los cuales día a día se dedican a analizar los ETLs generados por esos productos o servicios y los cuales sirven generar tendencias y entender cuál puede ser la mejor opción para los usuarios.

En un mundo cada vez más digital, el conocimiento que los científicos de datos aportan es invaluable. Gracias por su dedicación y su pasión por el análisis de datos, lo que nos permite entender mejor nuestro mundo y tomar decisiones informadas.

Siguen siendo pioneros en el campo de la ciencia de datos y su trabajo está ayudando a impulsar el progreso en muchas áreas, desde la medicina hasta la tecnología. Su habilidad para encontrar patrones y tendencias en los datos es fascinante y estamos muy agradecidos por su contribución a la sociedad.

Que sigan adelante con su trabajo y que sigan siendo una fuerza motriz en este campo.

RESUMEN

Este trabajo expone la creación de un pipeline de datos en la nube en la plataforma AWS (Amazon Web Services) que mediante un ETL, que por sus siglas en Ingles Extract, Transform and Load se logra resolver el problema de saber cuáles son las aerolíneas registradas en la IATA (International Air Transport Association) que tienen más demoras en la salida, y en la llegada y mediante este análisis exponer un trabajo futuro para agregarlo como una nueva funcionalidad en las aplicaciones de venta de vuelos para que así los usuarios tengan un nuevo parámetro para elegir mediante mayor información, siendo este el objetivo principal del trabajo, el cual es:

Crear un pipeline de ingesta de datos que permita recopilarlos y procesarlos de diversas fuentes y prepararlos para su análisis. Para hacerlo efectivo y útil para el análisis, es necesario definir un modelo de datos que estructure los datos de manera coherente y permita extraer información valiosa. AWS ofrece una amplia gama de herramientas que pueden ayudar a crear y gestionar un pipeline de ingesta de datos, como Amazon Glue. Además, es importante generar métricas que consideren las demoras en relación con los tipos de datos, en este caso, sobre las demoras en los vuelos. De esta manera, podremos optimizar el pipeline y garantizar un flujo de datos eficiente y de alta calidad.

De igual manera se pretende confirmar la precisión de los datos usando un modelo de machine learning de regresión lineal con la herramienta de pyspark. Este tipo de modelo es muy útil para clasificar datos y puede ayudar a mejorar la precisión de las predicciones.

Además, pyspark es una herramienta muy versátil y potente que permite realizar análisis y procesamiento de grandes conjuntos de datos de manera rápida y eficiente. Es importante tener en cuenta que la precisión de los datos es crucial para el éxito de cualquier proyecto de machine learning, por lo que es necesario realizar pruebas y validaciones para asegurarse de que los datos son confiables y están limpios.

TABLA DE CONTENIDO

MAESTRÍA EN SISTEMAS COMPUTACIONALES **1**

1. INTRODUCCIÓN **11**

1.1. ANTECEDENTES 11
1.2. JUSTIFICACIÓN 12
1.3. PROBLEMA 12
1.4. HIPÓTESIS 13
1.5. OBJETIVOS 13
1.5.1. Objetivo General: 13
1.5.2. Objetivos Específicos: 14
1.6. NOVEDAD TECNOLÓGICA O APORTACIÓN 14

2. ESTADO DEL ARTE O DE LA TÉCNICA **16**

2.1. PIPELINES DE INGESTA DE DATOS 16
2.2. QUE ES UN ETL Y POR QUÉ ES IMPORTANTE 17
2.3. LA NUBE EN GENERAL Y TIPOS DE SERVICIOS 18
2.3.1. SaaS (Software como servicio) 19
2.3.2. PaaS (Plataforma como servicio) 20
2.3.3. IaaS (Infraestructura como servicio) 20
2.1. EJEMPLO DEL USO DE PIPELINES EN LA NUBE USADO EN LA INDUSTRIA DE VIAJES Y AEROLÍNEAS 22
2.2. LA INDUSTRIA DE LA AVIACIÓN Y LAS AEROLÍNEAS 22
2.3. BIG DATA EN LA INDUSTRIA EN GENERAL 25
2.4. LA IATA (INTERNATIONAL AIR TRANSPORT ASSOCIATION) 27
2.5. MODELO DE REGRESIÓN LINEAL 28
2.6. APLICACIONES DE LA REGRESIÓN LINEAL 29

3. MARCO TEÓRICO/CONCEPTUAL **30**

3.1. CONCEPTO BÁSICO 1, LA NUBE EN GENERAL 30
3.1.1. Informática en la nube 30
3.1.2. Margen de mercado de las plataformas de la nube 31
3.1.3. AWS 32
3.1.4. Regiones de AWS 33
3.1.5. Zonas de AWS 34
3.1.6. S3 36
3.2. SERVICIOS TÉCNICOS 37
3.2.1. AWS Glue 37
3.2.2. Crawlers 39
3.2.3. Data Lake 41
3.2.4. Spark con AWS EMR 42
3.2.5. Amazon Athena 44
3.2.6. AWS Quicksight 45

4. DESARROLLO METODOLÓGICO **48**

4.1. Análisis metodológico 48
4.2. Requerimientos para el pipeline 49
4.3. Servicios usados para el pipeline 49
5. CONCLUSIONES Y RESULTADOS 54
5.1. Conclusiones generales del proyecto 54
5.2. Resultados Código de Spark con pyspark 56
5.3. Costos de los serivios de AWS 58
5.4. Conclusiones generales y objetivos completados 59
6. TRABAJO FUTURO 61
7. BIBLIOGRAFÍA 71
8. ANEXOS 75
8.1. Código 75
8.2. Pasos tomados en la consola de AWS para cada servicio usado 89
8.3. Imágenes de costos en consola de AWS 95

LISTA DE FIGURAS

Ilustración 1, Quien maneja qué? En cloud services....21
Ilustración 2, Capitalización de mercado de la nube....31
Ilustración 3, AWS grafica de liderazgo de mercado....32
Ilustración 4, Servicios de AWS....33
Ilustración 5, AWS Zonas de cobertura en América Latina....36
Ilustración 6, Funcionamiento S3 Bucket....37
Ilustración 7, Funcionamiento AWS Glue....39
Ilustración 8, Funcionamiento rastreador (Crawler)....40
Ilustración 9, Servicios de Lago de datos....42
Ilustración 10, EMR y Spark con AWS....43
Ilustración 11, Funcionamiento de AWS Athena....45
Ilustración 12, Funcionamiento AWS Quicksight....47
Ilustración 13 Arquitectura del pipeline de AWS. Diagrama Propio....52
Ilustración 14, Arquitectura de AWS Pipeline....56
Ilustración 15, Métricas de aerolíneas con demora en AWS Quicksight.....57
Ilustración 16, Demoras de vuelos por día del mes y por semana....58
Ilustración 17, Prototipo 1....62
Ilustración 18, Prototipo 2....63
Ilustración 19, Prototipo 3....65
Ilustración 20, Prototipo 4....66
Ilustración 21, Prototipo 5....68
Ilustración 22, Prototipo 6....69
Ilustración 23, selección de zonas de disponibilidad de AWS....89
Ilustración 24, creación de S3 Bucket....90
Ilustración 25, Lista de S3 Bucket....90
Ilustración 26, Creación de los rastreadores....91
Ilustración 27, Lista de rastreadores....91
Ilustración 28, AWS Glue Jobs....92
Ilustración 29, Creación de las tablas....92
Ilustración 30, Creación de los Jobs en AWS....93
Ilustración 31, Transformación y mapeo de los datos....93
Ilustración 32, Creación de los queries y vistas....94
Ilustración 33, Corriendo los Jobs....94
Ilustración 34, Costo por bucket por mes....95
Ilustración 35, Costo de S3 por GB....96
Ilustración 36, Costo AWS Glue....96
Ilustración 37, Costo AWS Athena....97
Ilustración 38, Costo AWS Quicksight....97

LISTA DE TABLAS

Tabla 1, data frame de la IATA con demoras de las aerolíneas 78
Tabla 2, data frame con columna label binaria si el delay es mayor a 15 79
Tabla 3, con columnas de features que son los vectores, la columna de prediction y de true label 83
Tabla 4, con los raw prediction y su probabilidad 86
Tabla 5, mostrando las nuevas predicciones 88

LISTA DE ACRÓNIMOS Y ABREVIATURAS

ETL	Extract,Transform and Load (Extraer, transformar y cargar)
ML	Machine Learning (Aprendizaje Maquina)
AWS	Amazon Web Services
IEEE	Institute of Electrical and Electronics Engineers
IATA	International Air Transport Association(Asociación Internacional de Transporte Aéreo
TI	Tecnologías de la Información
S3	Amazon Simple Storage Service
CSP	Cloud Service Provider
AUR	Area Under the Rock

1. INTRODUCCIÓN

En este capítulo, se presenta una introducción al objeto de estudio sobre los pipelines de ingesta de datos en la nube, como se hacían antes de la nube y se proporciona una justificación de su importancia en general y para este trabajo. Además, se definirá el problema que se está tratando y se plantearán las hipótesis que se estarán investigando durante el estudio. Es importante destacar que los antecedentes del objeto de estudio también se mencionarán brevemente para proporcionar contexto y comprensión del tema. Esta sección es fundamental para establecer el marco de referencia del estudio y proporcionar una base sólida para el trabajo que se realizará a lo largo del proyecto.

1.1. Antecedentes

Antes de la llegada de la nube, los pipelines de ingestión de datos se ejecutaban principalmente en servidores locales o en centros de datos. Esto implicaba una serie de desafíos, como la necesidad de mantener y administrar hardware costoso y complicado, y la limitación de recursos en términos de capacidad de procesamiento y almacenamiento.

Para construir un pipeline de ingestión de datos, se necesitaban herramientas de integración de datos y lenguajes de programación como SQL y Python. Estas herramientas permitían extraer, transformar y cargar datos desde fuentes externas en una base de datos local, donde podían ser procesados y analizados.

Con la llegada de la nube, muchos de estos desafíos se han mitigado ya que las plataformas en la nube ofrecen un almacenamiento y procesamiento de datos escalable y de bajo costo. Esto ha permitido a las empresas y organizaciones implementar pipelines de ingestión de datos más rápido y eficientemente, sin tener que preocuparse por la configuración y mantenimiento de servidores locales.

Además, las plataformas en la nube también ofrecen una variedad de herramientas y servicios que facilitan la creación y administración de pipelines de ingestión de datos, lo que ha llevado a una mayor adopción y utilización de estas tecnologías. En resumen, la llegada de la nube ha sido un gran avance para los pipelines de ingestión de datos, ya que ha simplificado su construcción y administración.

1.2. Justificación

La implementación de pipelines en la nube puede ofrecer una serie de ventajas competitivas a las aerolíneas, ya que permite una mayor flexibilidad y escalabilidad en la gestión de sus procesos de negocio, y en este caso, identificar las aerolíneas con más demoras en el mercado.

Uno de los principales beneficios de utilizar la nube para implementar pipelines es la capacidad de procesar grandes cantidades de datos en tiempo real. Esto permite a las aerolíneas analizar y obtener información valiosa de sus operaciones en tiempo real, lo que les permite tomar decisiones rápidas y mejorar la eficiencia en sus procesos.

Además, la nube también permite a las aerolíneas implementar sistemas de monitoreo y alerta en tiempo real, lo que les permite identificar y solucionar problemas de forma rápida y eficiente. Esto puede mejorar significativamente la calidad del servicio y la satisfacción del cliente.

En segundo lugar, la tecnología de la nube permite una colaboración en tiempo real y un acceso a los datos desde cualquier lugar, lo que facilita la toma de decisiones y la coordinación de las diferentes áreas de una aerolínea. Esto puede ayudar a mejorar la eficiencia en la gestión de los vuelos y la atención al cliente, lo que puede traducirse en una mejora en la calidad del servicio y en una ventaja competitiva para la aerolínea.

1.3. Problema

El problema de la falta de pipelines de datos eficientes es un obstáculo importante para poder aprovechar al máximo el valor de los datos y generar métricas útiles para las aerolíneas. Esto se debe a que las aerolíneas necesitan acceder rápidamente a grandes cantidades de datos para poder tomar decisiones informadas y mejorar sus operaciones. Sin una forma eficiente de transferir y procesar los datos, es difícil obtener la información necesaria para tomar

decisiones efectivas. Por lo tanto, es esencial desarrollar pipelines de datos que puedan transferir los datos de manera rápida y eficiente, lo que permitirá a las aerolíneas generar información valiosa y utilizarla para mejorar sus operaciones y aumentar su eficiencia.

1.4. Hipótesis

El uso de tecnología en la nube para los pipelines de datos es que esta tecnología es más eficiente y tiene ventajas competitivas en comparación con las tecnologías on-premise. La tecnología en la nube permite una mayor escalabilidad y flexibilidad en la gestión de los datos, lo que puede ser beneficioso para las aerolíneas que necesitan procesar grandes cantidades de datos. Además, el uso de la nube puede reducir los costos de hacer los pipelines, lo que puede ser una ventaja importante en un mercado altamente competitivo como el de las aerolíneas.

Otra hipótesis relacionada con el uso de los datos en las aerolíneas es que existe un nicho de oportunidad para generar nuevas métricas basadas en las demoras. Las aerolíneas recogen y almacenan una gran cantidad de datos sobre las demoras de los vuelos, pero aún no se han desarrollado métricas que permitan utilizar estos datos de manera eficiente.

1.5. Objetivos

1.5.1. Objetivo General:

Crear un pipeline de ingesta de datos en AWS, junto con técnicas de ciencia de datos y algoritmos de aprendizaje automático, para obtener una métrica de las aerolíneas con mejores puntuaciones en términos de demora en la llegada y en la salida. Esta información sería valiosa para las aerolíneas y para los pasajeros, ya que les permitiría conocer las opciones más confiables y eficientes para sus vuelos. Además, utilizando la nube de AWS Quicksight, es posible crear gráficos y visualizaciones de estos resultados, lo que facilitaría la interpretación de los datos y permitiría a las aerolíneas tomar decisiones informadas basadas en ellos. En resumen, el uso de un pipeline de AWS y técnicas de ciencia de datos puede ser muy útil para obtener información valiosa sobre las aerolíneas y sus servicios.

1.5.2. Objetivos Específicos*:*

- Desarrollar un pipeline de datos que utilice tecnología en la nube.
- Mejorar la eficiencia en la transferencia de datos a través del pipeline.
- Incrementar la flexibilidad y la escalabilidad del pipeline.
- Optimizar el pipeline de ingesta de datos para maximizar la eficiencia y minimizar los errores.
- Diseñar un modelo de datos que permita estructurar y procesar los datos de manera eficiente para su análisis.
- Investigar las herramientas de tecnología en la nube que son más adecuadas para la creación del pipeline de ingesta de datos.
- Desarrollar un pipeline de ingesta de datos en la nube que sea capaz de transferir y procesar grandes cantidades de datos de manera rápida y eficiente.
- Uso de algoritmo de aprendizaje máquina para validar el AUR (Area Under the Rock)
- Optimizar el uso de los datos para generar información valiosa para las aerolíneas.
- Identificar cuáles son las aerolíneas que tienen más demoras o retrasos.
- Identificar cuáles son las aerolíneas que tienen más vuelos en los tiempos identificados, es decir que respetan sus tiempos de despegue y de llegada.

1.6. Novedad tecnológica o aportación

Novedad Comercial: El presente trabajo pretende generar datos significativos por medio de un ETL(Extract, Transform, Load) e introducirlos a un pipeline de ingesta de datos en AWS(Amazon Web Services) en donde después de verificar la precisión o el accuracy de los datos por medio de un algoritmo de aprendizaje máquina, se puedan mostrar en una representación visual (Dashboard) en donde se puedan analizar de manera fácil por los usuarios.

Novedad Tecnológica: El empleo de tecnología en la nube para hacer un pipeline de datos representa una novedad tecnológica importante en el campo de la gestión de datos. La tecnología en la nube permite una mayor flexibilidad y escalabilidad en la gestión de los datos, lo que puede ser beneficioso para las aerolíneas que necesitan procesar grandes cantidades de información. Además, el uso de la nube puede reducir los costos y la complejidad del trabajo, lo que puede ser una ventaja importante en un mercado altamente competitivo como el de las aerolíneas.

Un pipeline de datos en la nube permite una mayor agilidad en el procesamiento y análisis de los datos, lo que puede ser útil para generar información valiosa para las aerolíneas. Por ejemplo, se podrían desarrollar métricas que permitan analizar las causas de las demoras de los vuelos y desarrollar estrategias para minimizarlas. Además, el uso de la nube también permite una mayor colaboración y compartición de datos entre diferentes departamentos y sistemas, lo que puede mejorar la eficiencia en la toma de decisiones.

Aportación: Efectivamente, utilizar la nube para crear pipelines de datos puede ser una forma muy beneficiosa para las empresas de aerolíneas y otras industrias. La nube ofrece una serie de ventajas, como la escalabilidad y la facilidad de acceso a grandes cantidades de datos, que pueden ayudar a las empresas a desarrollar nuevas métricas y tomar decisiones más informadas.

Con un pipeline de datos en la nube, las empresas de aerolíneas pueden recolectar y procesar grandes cantidades de datos en tiempo real, lo que les permite obtener información valiosa sobre sus operaciones y aumentar su eficiencia. Además, pueden utilizar herramientas de análisis avanzadas para crear nuevas métricas y hacer predicciones precisas sobre el rendimiento de su negocio.

Todo esto puede ayudar a las empresas de aerolíneas a tener una mejor comprensión de su negocio y a tomar decisiones estratégicas que les permitan tener una ventaja competitiva en el mercado. Por lo tanto, utilizar la nube para crear pipelines de datos puede ser una excelente forma de mejorar la eficiencia y el rendimiento de las empresas de aerolíneas.

A continuación, un breve resumen de los siguientes capítulos del trabajo:

En el capítulo 2 se presenta una revisión detallada de los trabajos relacionados en el estado del arte, este capítulo es esencial para entender el contexto en el que se enmarca el estudio y para conocer las investigaciones previas que han sido llevadas a cabo en el tema, en el capítulo 3 se aborda el marco teórico, este capítulo establece las bases conceptuales y teóricas del estudio, en el capítulo 4 se abordan los requerimientos y los servicios de AWS usados en el pipeline, el capítulo 5 se muestran los resultados y conclusiones, en donde se exponen los hallazgos principales del estudio, se presentan los resultados obtenidos en un formato claro y conciso, se discute lo más relevante de la investigación, y se establecen las conclusiones principales, en el capítulo 6 se expone el trabajo futuro, que es la aplicación del caso de uso de esta investigación, finalizando el trabajo con el capítulo 7 y 8 siendo la bibliografía y los anexos correspondientes.

2. ESTADO DEL ARTE O DE LA TÉCNICA

En este capítulo se presenta un resumen de los trabajos relacionados con el objeto de estudio, además se centra en presentar una revisión de la literatura sobre el objeto de estudio de este trabajo. Se han revisado varios artículos, libros y otras fuentes relevantes para obtener una comprensión más profunda del tema y para contextualizar el trabajo de investigación que se ha realizado. Se han identificado las principales tendencias y hallazgos en el campo y se han discutido las principales teorías y enfoques existentes. En resumen, este capítulo proporciona un marco teórico sólido para el trabajo de investigación que se presentará en los capítulos siguientes.

2.1. Pipelines de ingesta de datos

En el ámbito informático, la arquitectura en pipeline comprende un conjunto de procesos o fases secuenciales por los que va circulando un flujo de datos para ser transformado. Como consecuencia, estos procesos están conectados entre sí y, normalmente, la salida de una determinada fase es la entrada de la siguiente. En otras palabras; un pipeline, engloba una cadena de diferentes procesos que “beben unos de otros”.

Dentro de la estructura de pipeline, se pueden encontrar diferentes estados por los que debe pasar cada proceso parte de la “tubería virtual”; estos involucran desde la carga de la instrucción a ejecutar y los datos iniciales, hasta la escritura y el almacenamiento de los datos que resultan tras la ejecución de la instrucción principal.

Se puede simplificar como un esquema que interpreta un flujo constante de información, procesándola de forma secuencial. La comunicación de los procesos o datos a través de los pipelines se basa en una interacción productor/consumidor: la información necesaria para el proceso consumidor viene dada por el resultado del proceso productor anterior; es importante

resaltar que el almacenamiento de los datos intermedios entre un proceso y el siguiente es temporal.

Además, los pipelines tienen diversas características que los hacen muy comunes en sistemas operativos multitarea: capacidad de multiprogramación, empleando distintos lenguajes o llamadas a programas de distinta naturaleza; análisis en paralelo o secuencial según las necesidades y posibilidades; capacidad de ejecutar procesos independientes de manera simultánea.

La estructura general alberga diferentes estados por los que debe pasar cada proceso parte de la "tubería virtual". Los estados involucran desde la carga de la instrucción a ejecutar y los datos iniciales, hasta la escritura y el almacenaje de los datos que resultan tras la ejecución de la instrucción principal [1].

2.2. Que es un ETL y por qué es importante

ETL es un tipo de integración de datos que hace referencia a los tres pasos (extraer, transformar, cargar) que se utilizan para mezclar datos de múltiples fuentes. Se utiliza a menudo para construir un almacén de datos. Durante este proceso, los datos se toman (extraen) de un sistema de origen, se convierten (transforman) en un formato que se puede almacenar y se almacenan (cargan) en un data warehouse (almacén de datos) u otro sistema. Extraer, cargar, transformar (ELT) es un enfoque alterno pero relacionado y diseñado para canalizar el procesamiento a la base de datos para mejorar el desempeño [2].

Las empresas han confiado en el proceso ETL por muchos años para obtener una vista consolidada de los datos que dé lugar a mejores decisiones de negocios. Hoy día, este método de integración de datos de múltiples sistemas y fuentes sigue siendo un componente central de la caja de herramientas de integración de datos de una organización.

A continuación algunos puntos de por qué son importantes los ETL:

- Cuando se utiliza con un almacén de datos empresarial (datos en reposo), ETL provee profundo contenido histórico para la empresa.
- Proporcionando una vista consolidada, el ETL facilita a los usuarios de negocios analizar y generar reportes sobre datos relevantes para sus iniciativas.
- El ETL puede mejorar la productividad de los profesionales de los datos porque codifica y reutiliza procesos que mueven datos sin requerir habilidades técnicas para escribir código o scripts.

- El ETL ha evolucionado para satisfacer requisitos de integración emergentes para cosas como los datos transmitidos por streaming.
- Las organizaciones necesitan ETL y ELT para conjuntar datos, mantener la precisión y proporcionar el recurso de auditoría que suele requerirse en los almacenes, reportes y análisis de datos [2].

2.3. La nube en general y tipos de servicios

Cloud o la computación en la nube es acceso bajo demanda, a través de Internet, a recursos informáticos como aplicaciones, servidores (físicos y virtuales), almacenamiento de datos, herramientas de desarrollo, funciones de red y más, alojados en un centro de datos remoto gestionado por un proveedor de servicios en la nube (o CSP Cloud Service Provider). El CSP ofrece estos recursos en un plan de suscripción mensual o los factura según el uso.

En comparación con los Centros de Datos privados conocidos como on-premise tradicional, y dependiendo de los servicios en la nube que elija, la computación en la nube les permite a los usuarios:

- Reducir los costos: la nube ayuda a minimizar algunos o la mayoría de los costos y el esfuerzo que implica comprar, instalar, configurar y gestionar la propia infraestructura local.
- Mejorar la agilidad y la creación de valor: con la nube, las organizaciones pueden empezar a utilizar aplicaciones empresariales en minutos, en lugar de esperar semanas o meses para que se responda a una solicitud, pueden adquirir y configurar el hardware e instalar el software. La nube también permite capacitar a ciertos usuarios, específicamente desarrolladores y científicos de datos, para acceder por sí mismos a la infraestructura de software y soporte.
- Escalar de forma más fácil y rentable: la nube proporciona flexibilidad, ya que en lugar de adquirir una cantidad excesiva de recursos que no va a utilizar durante períodos lentos, puede aumentar o disminuir la capacidad en respuesta a alzas y caídas en el tráfico. También puede aprovechar la red global de un proveedor de nube para acercar sus aplicaciones a usuarios de todo el mundo.

El término "computación en la nube" también se refiere a la tecnología que hace que la nube funcione. Esto incluye algún tipo de Centros de Datos privados virtualizados, como servidores, software de sistema operativo, redes y otra infraestructura que se abstrae

mediante software especial, de modo que las tecnologías de la información (TI) se puedan agrupar y dividir independientemente de los límites físicos del hardware. Por ejemplo, un único servidor de hardware se puede dividir en varios servidores virtuales.

La virtualización permite a los proveedores de nube aprovechar al máximo sus recursos del centro de datos. No es de extrañar que algunas empresas hayan adoptado el modelo de entrega en la nube para su infraestructura local para conseguir la máxima utilización y ahorro de costos, en comparación con la infraestructura de Centros de Datos privados, y ofrecer el mismo autoservicio y agilidad a sus usuarios finales.

Si usa un computador o un dispositivo móvil en casa o en el trabajo, es casi seguro que utiliza algún tipo de computación en la nube todos los días, ya sea una aplicación en la nube como Google Gmail o Salesforce, medios de streaming como Netflix o almacenamiento de archivos en la nube como Dropbox. Según una encuesta reciente, el 92 % de las empresas hoy en día utiliza la nube y la mayoría planea seguir usándola el próximo año [3].

IaaS (Infraestructura como servicio), PaaS (Plataforma como servicio) y SaaS (Software como servicio) son los tres modelos más comunes de servicios en la nube, y no es raro que una organización utilice los tres. Sin embargo, a menudo hay confusión entre los tres y lo que incluye cada uno:

2.3.1. SaaS (Software como servicio)

SaaS, también conocido como software basado en la nube o aplicaciones en la nube, es un software de aplicación que se aloja en la nube y al cual se accede a través de un navegador web, un cliente de desktop dedicado o una API que se integra con el sistema operativo de desktop o dispositivo móvil. En la mayoría de los casos, los usuarios de SaaS pagan un plan de suscripción mensual o anual, aunque algunos proveedores pueden ofrecer planes basados en su uso real (pago por uso).

Además de los beneficios de ahorro de costos, creación de valor y escalabilidad de la nube, SaaS ofrece lo siguiente:

- Actualizaciones automáticas**:** con SaaS, puede aprovechar los nuevos recursos tan pronto como el proveedor los añade, sin tener que actualizar de forma local.
- Protección frente a pérdidas de datos**:** debido a que los datos de su aplicación están en la nube, con la aplicación, no se pierden datos si el dispositivo se bloquea o se rompe.

SaaS es el modelo de entrega principal para la mayoría de los softwares comerciales actuales. Hay cientos de miles de soluciones SaaS disponibles, desde las aplicaciones más centradas en la industria y departamentales, hasta potentes bases de datos de software empresarial y software de IA.

2.3.2. PaaS (Plataforma como servicio)

PaaS proporciona a los desarrolladores de software una plataforma bajo demanda, que incluye hardware, colección de software completa, infraestructura e incluso herramientas de desarrollo, para ejecutar, desarrollar y gestionar aplicaciones sin el costo, la complejidad y la inflexibilidad de mantener esa plataforma en las instalaciones.

Con PaaS, el proveedor de nube aloja servidores, redes, almacenamiento, software de sistema operativo, middleware y bases de datos en su centro de datos. Los desarrolladores simplemente escogen de un menú para "iniciar" los servidores y entornos que necesitan para ejecutar, desarrollar, probar, implementar, mantener, actualizar y escalar aplicaciones.

Actualmente, PaaS se desarrolla a menudo alrededor de contenedores, un modelo de computación virtualizado que se ha eliminado de los servidores virtuales. Los contenedores virtualizan el sistema operativo, lo que permite a los desarrolladores empaquetar la aplicación utilizando solamente los servicios del sistema operativo que necesita para ejecutarse en cualquier plataforma, sin modificación y sin necesidad de middleware.

Red Hat Open Shift, por ejemplo, es una PaaS popular desarrollada alrededor de contenedores Docker y Kubernetes, una solución de orquestación de contenedores de código abierto que automatiza la implementación, el escalamiento, el equilibrio de carga y otras funciones de las aplicaciones basadas en contenedores.

2.3.3. IaaS (Infraestructura como servicio)

IaaS proporciona acceso bajo demanda a los recursos informáticos fundamentales (servidores físicos y virtuales, redes y almacenamiento) a través de Internet en una base de pago por uso. IaaS permite a los usuarios finales escalar y reducir los recursos según sea necesario, lo que elimina la necesidad de grandes inversiones iniciales o de una infraestructura local o "propia" innecesaria y además evita la compra exagerada de recursos para adaptarse a alzas periódicas de uso.

A diferencia de SaaS y PaaS (y modelos informáticos PaaS incluso más recientes como contenedores y sin servidor), IaaS proporciona a los usuarios el nivel más bajo de control de recursos informáticos en la nube.

IaaS fue el modelo de computación en la nube más popular cuando surgió a principios de la década de 2010. Aunque sigue siendo el modelo de nube para muchos tipos de cargas de trabajo, el uso de SaaS y PaaS está aumentando mucho más rápido [3].

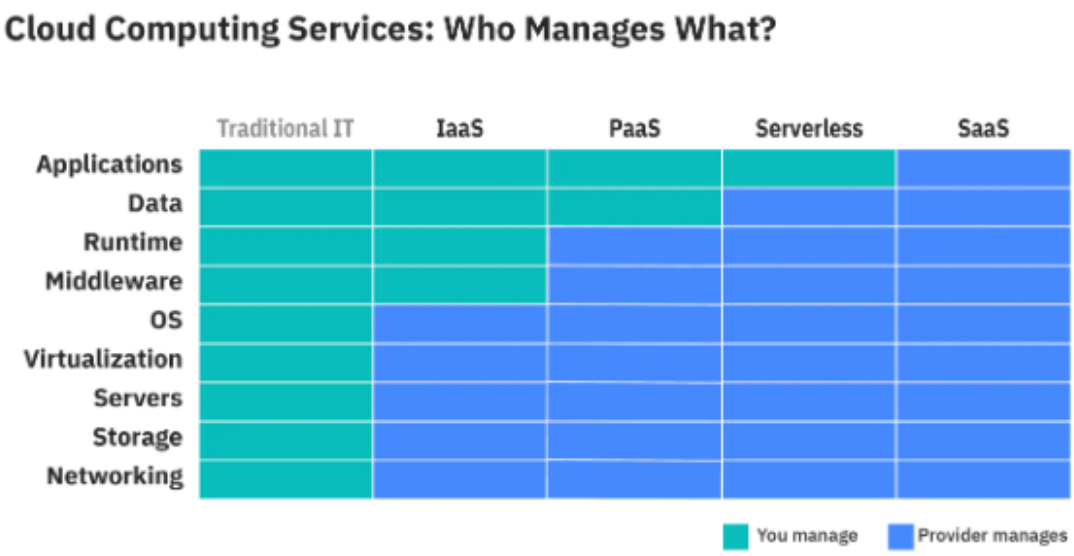

Ilustración 1, Quien maneja qué? En cloud services

"IaaS vs. PaaS vs. SaaS | IBM." https://www.ibm.com/topics/iaas-paas-saas (consultado Dic. 24, 2022).

La computación en la nube (cloud computing) se ha convertido en una pieza clave dentro de la estrategia de las compañías que buscan estar a la vanguardia y avanzar en su desarrollo.

Esta infraestructura no es precisamente algo nuevo, pero ahora es indispensable en muchos sectores, especialmente aquellos en los que la transformación digital no es una opción, pues debido a su rápido crecimiento siempre es necesario ampliar las capacidades de cómputo y almacenamiento de datos.

La consolidación de la multi-nube y nube híbrida, de acuerdo con Equinix, el 93% de las empresas expresaron interés o ya se comprometieron con la nube híbrida, lo que demuestra que hoy es realmente importante poder gozar del acceso a una conveniente combinación de los puntos más fuertes de la nube pública y la privada.

Anteriormente lo más común era elegir entre una nube pública o una privada, cada una de estas tiene ventajas, por lo que hace un tiempo se ha vuelto normal la implementación de una combinación entre ambas que permita contar con todos sus beneficios en conjunto, llegando a la llamada nube híbrida [4].

2.1. Ejemplo del uso de pipelines en la nube usado en la industria de viajes y aerolíneas

Expedia Group posee más de 20 sitios de reserva, tales como Expedia, Vrbo, Hotels.com y Orbitz, mediante los cuales los pasajeros de más de 70 países reservan alojamiento, vuelos y mucho más en más de 80 divisas. Como es un negocio de plataformas globales, los equipos necesitan atender los pagos de los clientes y los socios. En el caso del pago de los socios, Expedia interactúa con proveedores, como hoteles, líneas de cruceros y aerolíneas, cuyas reservas se distribuyen en la plataforma. En el modelo de Expedia Collect, Expedia recolecta el dinero de los clientes y luego envía los pagos a los proveedores. En 2019, Expedia procesó 7600 millones de USD en reservas comerciales, un 25 % más que el año anterior.

Sin embargo, el crecimiento de Expedia comenzó a afectar de forma negativa el segmento de conciliación de cuentas por pagar (APRecon). A medida que el volumen aumentaba, los trabajos tomaban más tiempo en ejecutarse, lo que ocasionó un efecto en cascada de demoras hasta llegar a los usuarios, estas demoras refiriéndose a otras que las estudiadas en este trabajo, son importantes de igual manera. La APRecon en SQL Server también requería una cantidad significativa de intervenciones manuales que afectó al personal operativo de la empresa. Y algunos servicios creados en .NET, una plataforma de desarrollo de código abierto, provocaron la dependencia en los proveedores, lo que limitó la implementación del servicio [5].

2.2. La industria de la aviación y las aerolíneas

En el mundo moderno, las aerolíneas juegan un papel vital para el transporte de personas y mercancías. Cualquier retraso en los horarios de estos vuelos pueden afectar negativamente el trabajo y negocios de miles de personas en cualquier momento. Prever estos retrasos es muy importante durante el proceso de planificación en las aerolíneas comerciales.

Ya se han propuesto varias técnicas para diseñar modelos para pronosticar el retraso en la hora de salida de aviones, pero debido al continuo aumento y complejidad del transporte aéreo y la cantidad de datos relacionados con ella, diseñar métodos de predicción se ha vuelto muy difícil.

Los aeropuertos son conocidos por su capacidad para aumentar las actividades comerciales cerca de ellos y por lo tanto dan como resultado el desarrollo económico.

La industria de la aviación también proporciona una gran cantidad de trabajos. Un récord de 3.700 millones de pasajeros hizo uso de aviones e instalaciones de transporte en el año 2016 y este número se espera que siga aumentando cada año. Los informes de tráfico aéreo mundial [6] publicado por el la Asociación Internacional de Transporte Aéreo (IATA) demostró que la demanda de viajes aéreos aumentó un 6,3 por ciento en el año 2016 en comparación con el año 2015. Este tipo de tráfico de volumen aéreo debe ser constantemente supervisado y controlado para evitar cualquier problema [6].

El análisis de retrasos de las aerolíneas tiene como objetivo evaluar qué factores contribuyen con mayor probabilidad a los retrasos de las aerolíneas, lo que puede ayudar a las compañías aéreas a evitar los retrasos de las aerolíneas y a planificar mejor los vuelos para minimizar su pérdida. El retraso de los vuelos se atribuye a varios factores, como las malas condiciones climáticas, fallas físicas, retrasos en la llegada de los vuelos y problemas relacionados con la tripulación. Con la ayuda de la visualización y el análisis de Big Data, se investigan varios factores que contribuyen a los retrasos de las aerolíneas para hacer sugerencias a las compañías aéreas con modelos de Big data para la industria de la aviación [7].

Con el continuo cambio de la estructura y parámetros del sistema de las aeronaves, es muy difícil establecer un modelo matemático preciso. Por lo tanto, los métodos de diagnóstico de fallas basados en el conocimiento se han desarrollado rápidamente en años recientes. Como lo sería un diagnóstico inteligente representativo típico con métodos, tales como análisis de agrupamiento, reglas de asociación, redes neuronales, máquina de vectores de soporte, etc.

Los algoritmos usan datos de muestra como entrada, por lo que el diagnóstico de fallas y los resultados están sesgados hasta cierto punto. Si todos los datos recopilados por todas las unidades a lo largo de los años se pudieran utilizar como entrada, los resultados del diagnóstico serían relativamente confiables y las estadísticas como la probabilidad de error de diagnóstico sería menor [8].

Basado en la plataforma Internet e Internet de las cosas, esa capa de adquisición de datos recopila todos los datos relacionados con el vuelo desde antes del vuelo, durante el vuelo hasta después del vuelo a través de la información obtenida, sistemas tales como sistema de parámetros de vuelo aerotransportado, sistema de tierra, sistema de decodificación de

parámetros de vuelo, grabación de voz en cabina , sistema de adquisición de vídeo y enlace de datos terrestres, incluyendo categoría multidimensional en la estructura de bucle cerrado del bucle hombre-máquina. Los datos son recogidos en el centro de datos de gestión de vuelos [8].

En resumen, big bata se refiere a la cantidad masiva y compleja de datos que se generan en el mundo digital, que son difíciles de procesar y analizar utilizando tecnologías tradicionales. La gran cantidad de datos que se generan a diario en la era digital ha llevado a la necesidad de desarrollar nuevas tecnologías y enfoques para poder manejarlos efectivamente.

Uno de los principales retos de computabilidad asociados con el Big Data es su volumen y complejidad. La cantidad de datos que se generan a diario es tan grande que es difícil de almacenar y procesar utilizando tecnologías tradicionales. Además, muchos de estos datos son altamente complejos, lo que dificulta su análisis y utilización para tomar decisiones informadas.

Otro reto importante es la velocidad a la que se generan los datos. La cantidad de datos que se generan en tiempo real es tan grande que es difícil procesarlos y analizarlos en tiempo real utilizando tecnologías tradicionales. Esto puede dificultar la toma de decisiones en tiempo real y limitar la eficiencia y el rendimiento de las empresas.

Además, el big data también plantea desafíos éticos y de privacidad, ya que puede implicar el manejo de datos personales sensibles en grandes cantidades.

En resumen, Big data y el cómputo tradicional son dos enfoques diferentes para el procesamiento y análisis de datos.

A continuación, se presentan algunas de las principales ventajas y desventajas de cada enfoque:

Ventajas del big data:

Capacidad para procesar grandes cantidades de datos: el big data se basa en el uso de sistemas de almacenamiento y procesamiento distribuidos que pueden manejar grandes volúmenes de datos de manera eficiente. Esto permite procesar datos que serían demasiado grandes o complejos para ser manejados por sistemas tradicionales.

Escalabilidad: el big data es altamente escalable, lo que significa que se puede añadir más capacidad de procesamiento y almacenamiento según sea necesario. Esto permite adaptarse a los cambios en el volumen de datos y las necesidades de procesamiento.

Análisis en tiempo real: el big data permite el análisis en tiempo real de grandes cantidades de datos, lo que significa que se pueden obtener resultados y tomar decisiones de manera más rápida.

Desventajas del big data:

- Mayor complejidad: el big data implica una mayor complejidad en la configuración y gestión de sistemas distribuidos, lo que puede ser difícil para algunos usuarios.
- Mayor costo: el big data puede ser más costoso que el cómputo tradicional debido a la necesidad de hardware y software de alto rendimiento y a la necesidad de contratar a personal especializado para gestionar los sistemas.

Ventajas del cómputo tradicional:

- Mayor sencillez: el cómputo tradicional se basa en sistemas de un solo servidor y es más sencillo de configurar y gestionar que el big data.
- Menor costo: el cómputo tradicional puede ser más asequible que el big data debido a la menor necesidad de hardware y software de alto rendimiento.

Desventajas del cómputo tradicional:

- Menor capacidad para procesar grandes cantidades de datos: el cómputo tradicional suele tener menor capacidad de procesamiento y almacenamiento que el big data, lo que puede limitar la cantidad de datos que se pueden procesar.
- Menor escalabilidad: el cómputo tradicional suele ser menos escalable que el big data, lo que significa que es más difícil adaptarse a los cambios en el volumen de datos y las necesidades de procesamiento.

2.3. Big Data en la industria en general

Con el rápido desarrollo de la tecnología de la información como Internet, computación en la nube e Internet de las Cosas, los seres humanos han entrado en una nueva era de "big data". Grandes cantidades de datos se están convirtiendo en un nuevo foco de competencia de recursos estratégicos en el mundo. La administración Obama de los Estados Unidos impulsó el proyecto de investigación de big data en marzo de 2012 e invirtió $200 millones para desarrollar la industria de big data [9].

Estados Unidos publicó el informe "desarrollo de big data: oportunidades y desafíos" en mayo de 2012, que recibió respuestas positivas de países desarrollados como Gran Bretaña, Francia, Alemania y Japón [10]; China estableció el gran comité de expertos en datos de la sociedad de comunicación de China en octubre 2012. En la aplicación del big data en el

campo del ejército, el ejército estadounidense sigue siendo el líder. hay alrededor de 800 centros de datos y más de 70000 servidores en el ejército de los EE. UU., y la cantidad de datos es absolutamente dominante en el mundo.

Por lo tanto, el ejército de EE. UU. ha establecido el proyecto de big data como el objeto clave de investigación, y a través de la investigación de big data, podemos lograr el objetivo de continuamente mejorar la capacidad militar y de toma de decisiones [11].

El análisis de Big Data se ha utilizado con éxito en muchos dominios para descubrir información para la toma de decisiones informada [12], [13]. Un ejemplo fue permitir el paso de Internet de cosas al control en tiempo real [14], [15]. Sin embargo, la aplicación del análisis de big data en la aviación es limitado. El retraso del vuelo ha sido un problema grave y generalizado que afecta tanto a los y viajes aéreos internacionales. En los últimos años, debido a los retrasos en la aviación en los Estados Unidos están aumentando, el desarrollo de nuevas tecnologías que mitiguen este aspecto.

La industria de la aviación civil se ha visto gravemente afectada debido a las restricciones de viaje y la disminución de la demanda de vuelos comerciales. De acuerdo con los datos estadísticos de retraso de las líneas aéreas, sólo el 79 por ciento de vuelos en 2019 llegaron a tiempo, lo que resulta en decenas de miles de millones de pérdida de dólares estadounidenses, incluido el costo para las aerolíneas y los pasajeros, pérdida de demanda y otros costos indirectos. Como un gran negocio en cuestión que da lugar a un importante impacto económico y reputacional pérdida para las aerolíneas, el retraso de los vuelos debe estudiarse a fondo para reducir el costo utilizando sus propias características.

Los retrasos de ciertos vuelos se propagarán a otros y la eficiencia de la operación del aeropuerto se deteriorará con el tiempo sin monitoreo oportuno e interposición. Por lo tanto, se prefiere capturar cualquier tipo de perturbaciones en el horario y la operación escenario y predecir el retraso del vuelo antes de que ocurran consecuencias inesperadas. Muchos estudios, que dependen en gran medida de la simulación y el modelado, se han realizado para simular el retraso en la llegada de aviones.

Proporcionar una base para establecer estrategias de gestión puede ser la opción para reducir estas demoras en los vuelos, sin embargo, esos métodos basados en simulación no siempre son capaces de emular comportamiento real del aeropuerto y brindar una solución en tiempo y forma.

Además, hay múltiples factores que pueden causar retrasos en los vuelos, y necesitan ser identificados y considerados al mismo tiempo, tales como malas condiciones meteorológicas, congestión del aeropuerto en hora pico, retrasos de personal y pasajeros, dificultades técnicas, y demás. Es difícil para los modelos manejar el complejo ambiente operativo que implica

múltiples factores de retardo y hacer suposiciones apropiadas. Por lo tanto, un modelo de predicción podría estar sesgado.

Se necesita alta precisión y rendimiento en tiempo real para establecer mejores estrategias de operación para abordar el retraso de los vuelos de manera eficiente. Con el desarrollo de la aviación actual y sistemas de información, cada vez hay más datos de vuelo disponibles incluidos los factores que podrían contribuir a los retrasos en los vuelos de nivel macro a nivel micro. Además, tomar en cuenta el análisis de datos de aviación utilizando técnicas de aprendizaje automático [16].

En resumen, este trabajo de obtención de grado se centra en la creación de métricas para medir los retrasos de los vuelos y las aerolíneas que más reportan demoras, en lugar de enfocarse en el análisis detallado de las causas de dichos retrasos.

2.4. La IATA (International Air Transport Association)

IATA es la asociación comercial de las aerolíneas del mundo. Apoya muchas áreas de la actividad de la aviación y ayuda a formular políticas de la industria sobre problemas críticos de aviación. Es el principal vehículo para la cooperación entre líneas aéreas en la promoción de servicios aéreos seguros, fiables, protegidos y económicos para el beneficio de los consumidores del mundo.

La IATA tiene como objetivo ser la fuerza para la creación de valor y la innovación que impulse una industria del transporte aéreo segura, protegida y rentable que conecta y enriquece al mundo. La misión de IATA es representar, liderar y servir a la industria de las aerolíneas.

De 57 miembros fundadores en 1945, IATA ahora representa a 281 aerolíneas en más de 120 países. Transportando el 83% del tráfico aéreo internacional del mundo, los miembros de la IATA incluyen a los líderes mundiales de aerolíneas de pasajeros y de carga [17].

La misión de IATA es representar, liderar y servir a la industria de las aerolíneas representando a la industria aeronáutica.

Mejorando la comprensión de la industria del transporte aéreo entre los responsables de la toma de decisiones y aumentando la conciencia de los beneficios que la aviación aporta a las economías nacionales y mundiales. Al defender los intereses de las aerolíneas de todo el mundo, desafía las reglas y los cargos irrazonables, hace que los reguladores y los gobiernos rindan cuentas y lucha por una regulación sensata.

Durante más de 70 años, la IATA ha desarrollado estándares comerciales globales sobre los cuales se construye la industria del transporte aéreo. Su objetivo es ayudar a las aerolíneas simplificando los procesos y aumentando la comodidad de los pasajeros al mismo tiempo que se reducen los costos y se mejora la eficiencia [18].

2.5. Modelo de Regresión Lineal

La regresión lineal es una técnica de modelado estadístico que se emplea para describir una variable de respuesta continua como una función de una o varias variables predictoras. Puede ayudar a comprender y predecir el comportamiento de sistemas complejos o a analizar datos experimentales, financieros y biológicos.

Las técnicas de regresión lineal permiten crear un modelo lineal. Este modelo describe la relación entre una variable dependiente y (también conocida como la respuesta) como una función de una o varias variables independientes Xi (denominadas predictores). La ecuación general correspondiente a un modelo de regresión lineal es:

$Y=\beta 0+\sum \beta iXi+\epsilon iY=\beta 0+\sum \beta iXi+\epsilon i$

donde β representa las estimaciones de parámetros lineales que se deben calcular y $\epsilon\epsilon$ representa los términos de error [19].

La capa analítica es una parte importante del pipeline de big data, ya que es donde se realizan los análisis y se extraen conclusiones a partir de los datos suministrados. Un diagrama de arquitectura típico incluiría una capa de almacenamiento de datos, que puede ser una base de datos tradicional o un sistema de almacenamiento distribuido como Hadoop, seguida por una capa de procesamiento de datos, que puede incluir herramientas como Apache Spark. La capa analítica se encuentra encima de la capa de procesamiento de datos y es donde se realizan los análisis y se producen las métricas y otros resultados a partir de los datos.

Una vez que se han producido las métricas a partir de los datos, se pueden incorporar en un modelo de datos para su posterior consulta y análisis. Por ejemplo, la regresión lineal es una técnica comúnmente utilizada en la capa analítica para producir nuevas métricas a partir de los datos. Estas métricas pueden incluir cosas como predicciones de tendencias futuras o correlaciones entre diferentes variables en los datos. Una vez que se han producido estas métricas, se pueden incorporar en el modelo de datos para su posterior consulta y análisis.

2.6. Aplicaciones de la regresión lineal

La regresión lineal cuenta con ciertas características ideales para las siguientes aplicaciones:

- Predicción o pronóstico: utiliza un modelo de regresión para crear un modelo de pronóstico para un conjunto de datos específico. A partir de la moda, puede usar la regresión para predecir valores de respuesta donde solo se conocen los predictores.
- Fuerza de la regresión: utiliza un modelo de regresión para determinar si existe una relación entre una variable y un predictor, y cuán estrecha es esta relación [19].

3. MARCO TEÓRICO/CONCEPTUAL

En este capítulo se presentan las bases teóricas y conceptuales sobre el objeto de estudio.

3.1. Concepto básico 1, la nube en general

3.1.1. Informática en la nube

La informática en la nube es la distribución de recursos de TI bajo demanda a través de Internet mediante un esquema de pago por uso. En lugar de comprar, poseer y mantener servidores y centros de datos físicos, puede acceder a servicios tecnológicos, como capacidad informática, almacenamiento y bases de datos, en función de sus necesidades a través de un proveedor de la nube como Amazon Web Services (AWS) [20].

Como síntesis se puede decir que la informática en la nube es un enfoque para el almacenamiento, el procesamiento y el uso de la informática que se basa en el acceso a recursos informáticos a través de Internet en lugar de tener un acceso directo a ellos. Esto permite a las empresas y los individuos acceder a una amplia gama de servicios y aplicaciones informáticas sin tener que adquirir y mantener hardware y software localmente.

La informática en la nube se basa en el uso de centros de datos distribuidos y el acceso a ellos a través de Internet. Estos centros de datos alojan una gran cantidad de servidores y equipos de almacenamiento que pueden ser accedidos de forma remota y utilizados para almacenar y procesar datos, así como para ejecutar aplicaciones y servicios.

Además, tiene varias ventajas, como la capacidad de escalar rápidamente el uso de los recursos informáticos según sea necesario, la reducción de costos de hardware y software y la facilidad de acceso a aplicaciones y servicios desde cualquier lugar con una conexión a Internet. Sin embargo, también presenta algunos desafíos, como la seguridad de los datos almacenados en la nube y la dependencia de una conexión a Internet sólida para acceder a los recursos informáticos.

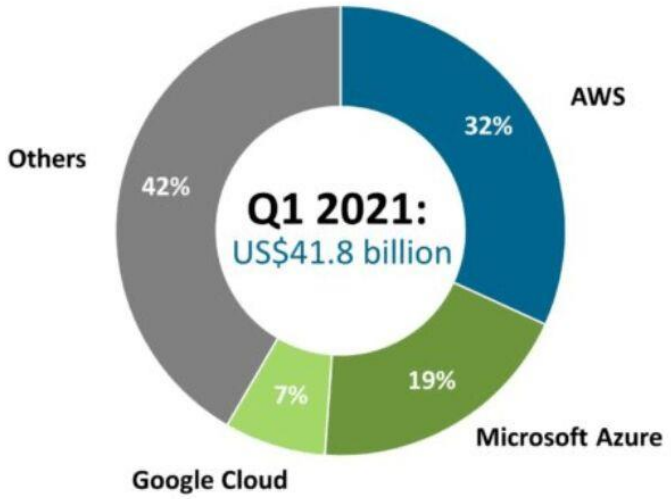

Ilustración 2, Capitalización de mercado de la nube

EES, "Public Cloud Market Share Statistics In 2022," *EES Corporation*, Sep. 13, 2021. https://www.eescorporation.com/public-cloud-market-share-statistics/ (accessed Dec. 24, 2022).

En la imagen 2 se muestra la capitalización del mercado de la nube que básicamente son las plataformas en la nube más usadas.

3.1.2. Margen de mercado de las plataformas de la nube

A continuación, como se puede ver en la ilustración 3, se presentan los márgenes de mercado de cada plataforma de la nube.

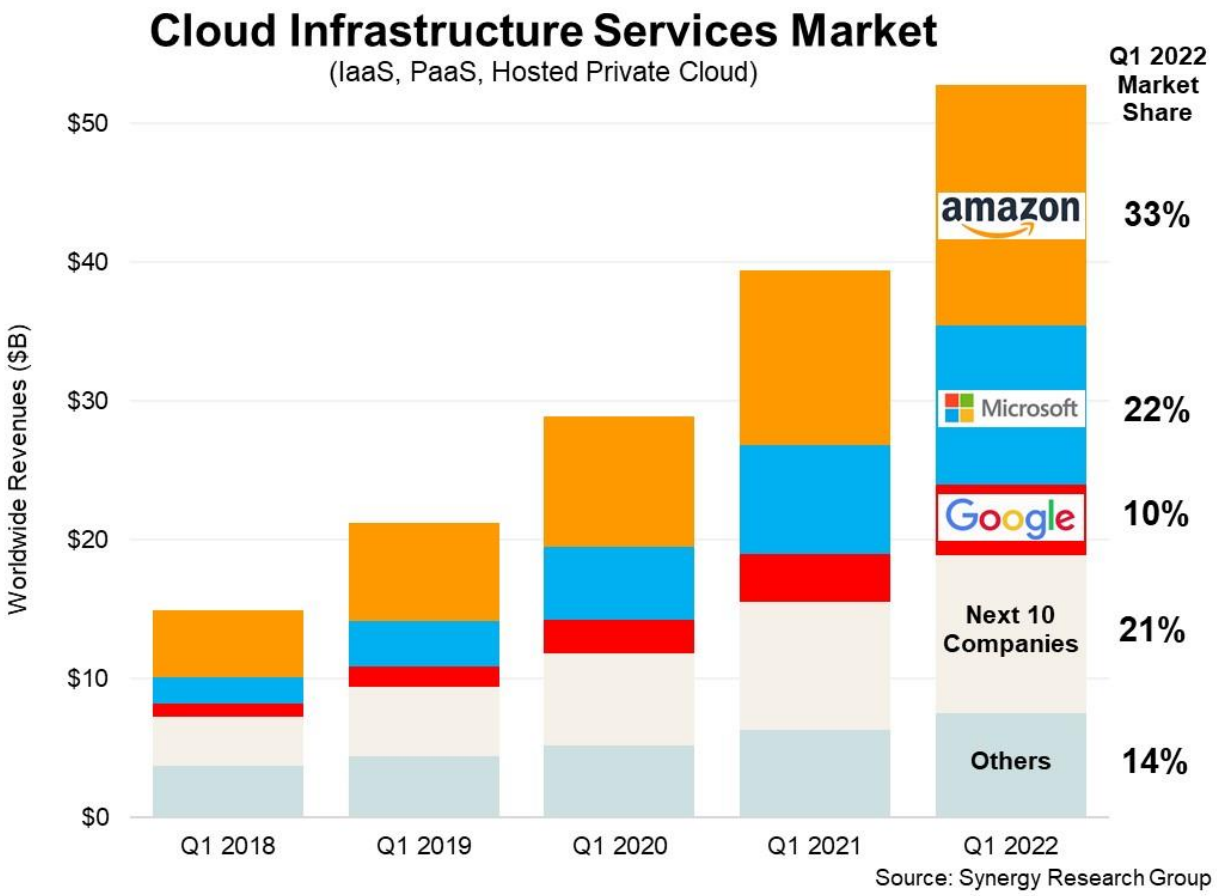

Ilustración 3, AWS grafica de liderazgo de mercado

"Huge Cloud Market Still Growing at 34% Per Year; Amazon, Microsoft & Google Now Account for 65% of the Total | Synergy Research Group." https://www.srgresearch.com/articles/huge-cloud-market-is-still-growing-at-34-per-year-amazon-microsoft-and-google-now-account-for-65-of-all-cloud-revenues (accessed Dec. 24, 2022).

3.1.3. AWS

Amazon Web Services (AWS) es la plataforma en la nube más adoptada y completa en el mundo, que ofrece más de 200 servicios integrales de centros de datos a nivel global. Millones de clientes, incluso las empresas emergentes que crecen más rápido, las compañías más grandes y los organismos gubernamentales líderes, están usando AWS para reducir los costos, aumentar su agilidad e innovar de forma más rápida [21].

Como síntesis, AWS ofrece una gran cantidad de servicios informáticos, como almacenamiento de datos, procesamiento de datos, bases de datos, servicios de redes y seguridad, herramientas de desarrollo y análisis de datos, entre otros. Estos servicios se

pueden utilizar de forma aislada o combinada para crear soluciones informáticas personalizadas para una amplia variedad de usos.

Una de las principales ventajas de AWS es su escalabilidad, que permite a las empresas aumentar o disminuir el uso de los servicios informáticos según sea necesario sin tener que adquirir hardware y software adicionales.

AWS también se destaca por su seguridad y confiabilidad. La plataforma cuenta con una serie de medidas de seguridad para proteger los datos y los servicios de los usuarios y tiene una alta disponibilidad para minimizar la interrupción de los servicios.

Ilustración 4, Servicios de AWS

"AWS Service Broker: Bridging the Gulf Between On-Premises and AWS | AWS Partner Network (APN) Blog," Nov. 29, 2017. https://aws.amazon.com/blogs/apn/aws-service-broker-bridging-the-gulf-between-on-premises-and-aws/ (accessed Dec. 24, 2022).

3.1.4. Regiones de AWS

AWS tiene el concepto de una región, que es una ubicación física en todo el mundo donde agrupamos los centros de datos. Llamamos a cada grupo de centros de datos lógicos "zona de disponibilidad". Cada región de AWS consta de varias zonas de disponibilidad aisladas y

separadas físicamente dentro de un área geográfica. A diferencia de otros proveedores de nube, que a menudo definen una región como un solo centro de datos, el diseño múltiple de zonas de disponibilidad de cada región de AWS ofrece ventajas para los clientes. Cada zona de disponibilidad tiene alimentación, refrigeración y seguridad física independientes y está conectada a través de redes redundantes de latencia ultra baja. Los clientes de AWS centrados en la alta disponibilidad pueden diseñar sus aplicaciones para que se ejecuten en múltiples zonas de disponibilidad y lograr una mayor tolerancia a errores. Las regiones de infraestructura de AWS cumplen con los niveles más altos de seguridad, cumplimiento y protección de datos.

AWS proporciona una presencia global más extensa que cualquier otro proveedor de nube, y, para respaldar su presencia global y garantizar que los clientes reciban servicios en todo el mundo, AWS abre nuevas regiones rápidamente. AWS mantiene múltiples regiones geográficas, incluidas las regiones de América del Norte, América del Sur, Europa, China, Asia-Pacífico, Sudáfrica y Medio Oriente [22].

En síntesis, las regiones de AWS son ubicaciones geográficas donde se encuentran los centros de datos de Amazon Web Services. Cada región tiene varios centros de datos y están diseñadas para ofrecer alta disponibilidad y baja latencia a los usuarios de AWS. Las regiones de AWS se utilizan para alojar y ejecutar aplicaciones y servicios en la nube, y los usuarios pueden elegir la región más adecuada en función de su ubicación geográfica, las regulaciones y otras consideraciones. En general, las regiones de AWS son una parte importante de la infraestructura de AWS y ayudan a garantizar la disponibilidad y el rendimiento de los servicios en la nube.

3.1.5. Zonas de AWS

Una zona de disponibilidad (AZ) es uno o más centros de datos discretos con alimentación, redes y conectividad redundantes en una región de AWS. Las zonas de disponibilidad permiten que los clientes operen bases de datos y aplicaciones de producción con un nivel de disponibilidad, tolerancia a errores y escalabilidad mayor que el que ofrecería un centro de datos único. Todas las zonas de disponibilidad en una región de AWS están interconectadas con redes de alto ancho de banda y baja latencia, a través de una fibra metropolitana exclusiva totalmente redundante que proporciona una red de alto rendimiento y baja latencia entre las zonas de disponibilidad. Todo el tráfico entre las AZ está cifrado. El rendimiento de la red es suficiente como para llevar a cabo la replicación sincrónica entre las zonas de disponibilidad. Las AZ facilitan la partición de las aplicaciones para una alta disponibilidad.

Si una aplicación se divide en AZ, las empresas estarán mejor aisladas y protegidas de problemas como cortes de energía, rayos, tornados, terremotos, etc. Las AZ están físicamente separadas entre sí por una distancia significativa de muchos kilómetros, aunque todas están dentro de un rango de 100 km (60 millas) de separación [27].

En síntesis, las zonas de disponibilidad de AWS son una subdivisión de las regiones de AWS que se utilizan para mejorar la disponibilidad y el rendimiento de los servicios en la nube. Cada región de AWS está dividida en varias zonas de disponibilidad, cada una con al menos un centro de datos y su propia fuente de energía y enfriamiento. Esto permite a los usuarios distribuir sus aplicaciones y servicios en la nube entre diferentes zonas de disponibilidad dentro de una región, lo que aumenta la resiliencia y reduce el riesgo de interrupciones. En general, las zonas de disponibilidad son una parte importante de la infraestructura de AWS y ayudan a garantizar la disponibilidad y el rendimiento de los servicios en la nube.

Ilustración 5, AWS Zonas de cobertura en América Latina

"AWS announces Local Zones in Latin America | AWS Public Sector Blog," Mar. 23, 2022. https://aws.amazon.com/blogs/publicsector/aws-announces-local-zones-latin-america/ (accessed Dec. 24, 2022).

3.1.6. S3

Amazon Simple Storage Service (Amazon S3) es un servicio de almacenamiento de objetos que ofrece escalabilidad, disponibilidad de datos, seguridad y rendimiento líderes en el sector. Clientes de todos los tamaños y sectores pueden almacenar y proteger cualquier

cantidad de datos para prácticamente cualquier caso de uso, como los lagos de datos, las aplicaciones nativas en la nube y las aplicaciones móviles. Gracias a las clases de almacenamiento rentables y a las características de administración fáciles de usar, es posible optimizar los costos, organizar los datos y configurar controles de acceso detallados para cumplir con requisitos empresariales, organizacionales y de conformidad específicos [23].

Ilustración 6, Funcionamiento S3 Bucket

"AWS | Almacenamiento de datos seguro en la nube (S3)," *Amazon Web Services, Inc.* https://aws.amazon.com/es/s3/ (accessed Dec. 24, 2022).

Como síntesis, S3 Buckets es un servicio de almacenamiento en la nube de Amazon Web Services que permite a los usuarios almacenar y recuperar grandes cantidades de datos en línea. Los buckets de S3 son contenedores lógicos donde se almacenan los objetos de S3, que pueden incluir cualquier tipo de datos, desde archivos binarios hasta texto simple. Los buckets de S3 se pueden utilizar para almacenar y administrar datos en aplicaciones en la nube, para realizar copias de seguridad y para distribuir contenido a través de la red de distribución de contenido de Amazon (CDN). En general, S3 Buckets es una herramienta poderosa y versátil para el almacenamiento en la nube que puede ayudar a las empresas a administrar sus datos y aumentar la disponibilidad y el rendimiento de sus aplicaciones en la nube.

3.2. Servicios técnicos

3.2.1. AWS Glue

AWS Glue es un servicio de integración de datos sin servidores que facilita la detección, preparación y combinación de datos para análisis, machine learning y desarrollo de

aplicaciones. AWS Glue proporciona todas las capacidades que se necesitan para la integración de datos, para que pueda comenzar a analizarlos y usarlos en minutos en vez de meses.

La integración de datos es el proceso de preparar y combinar datos para análisis, machine learning y desarrollo de aplicaciones. Involucra varias tareas, como descubrir y extraer datos de diversos orígenes; enriquecer, limpiar, normalizar y combinar datos; y cargar y organizar datos en bases de datos, almacenes de datos y lagos de datos. Normalmente, estas tareas las manejan diferentes tipos de usuarios y cada uno utiliza productos diferentes.

AWS Glue proporciona interfaces visuales y basadas en código para facilitar la integración de datos. Los usuarios pueden encontrar datos y acceder a ellos fácilmente con el catálogo de datos de AWS Glue. Los ingenieros de datos y los desarrolladores de ETL (extracción, trasformación y carga) pueden utilizar AWS Glue Studio para crear, ejecutar y supervisar visualmente flujos de trabajo con unos pocos clics. Los analistas y los científicos de datos pueden utilizar AWS Glue DataBrew para completar, limpiar y normalizar visualmente los datos sin escribir código.[24]

En síntesis, AWS Glue es un servicio de ETL (Extract, Transform, Load) en la nube de Amazon Web Services que permite a los usuarios extraer datos de diferentes fuentes, transformarlos y cargarlos en una base de datos o un data Waterhouse para su análisis y consulta. AWS Glue ofrece una plataforma unificada para la integración y el análisis de datos en la nube, lo que permite a las empresas simplificar y automatizar sus procesos de ETL y mejorar la eficiencia de sus análisis de datos. En general, AWS Glue es una herramienta valiosa para cualquier empresa que trabaje con grandes cantidades de datos y necesite integrar y analizar esos datos en la nube.

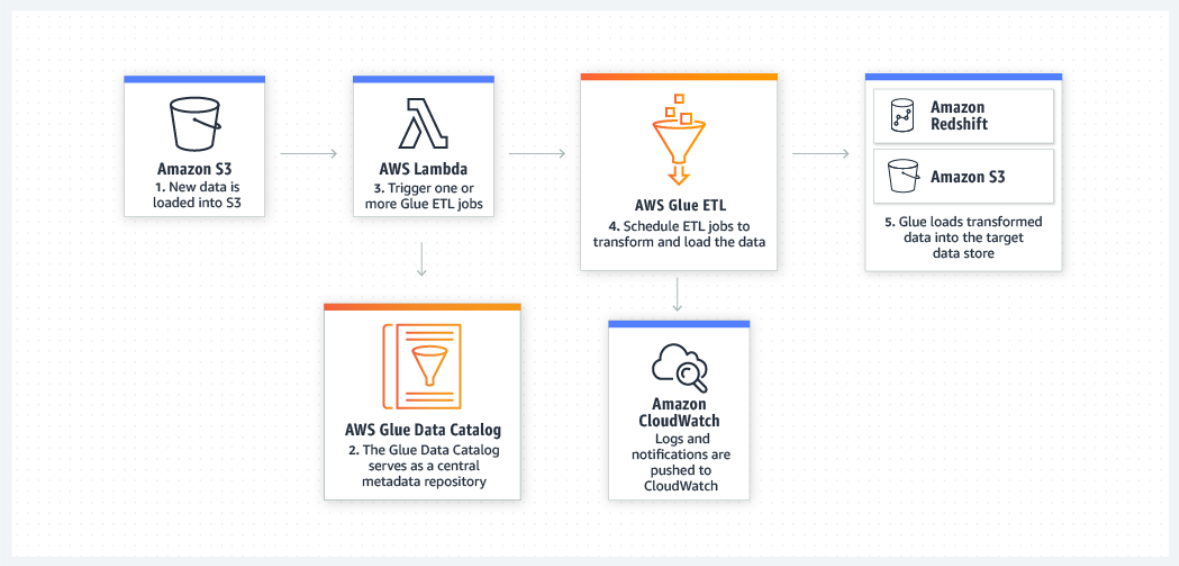

Ilustración 7, Funcionamiento AWS Glue

"Integración de datos sin servidor: AWS Glue, Amazon Web Services," *Amazon Web Services, Inc.* https://aws.amazon.com/es/glue/ (accessed Dec. 24, 2022).

3.2.2. Crawlers

Puede usar un rastreador para rellenar el AWS Glue Data Catalog con tablas. Este es el método principal usado por la mayoría de los usuarios de AWS Glue. Un rastreador puede rastrear varios almacenes de datos en una única ejecución. Cuando finaliza, el rastreador crea o actualiza una o varias tablas de su Data Catalog. Los trabajos de extracción, transformación y carga (ETL) que define en AWS Glue usan estas tablas del Data Catalog como orígenes y destinos. El trabajo de ETL lee y escribe en los almacenes de datos que se especifican en las tablas de origen y destino del Data Catalog [25].

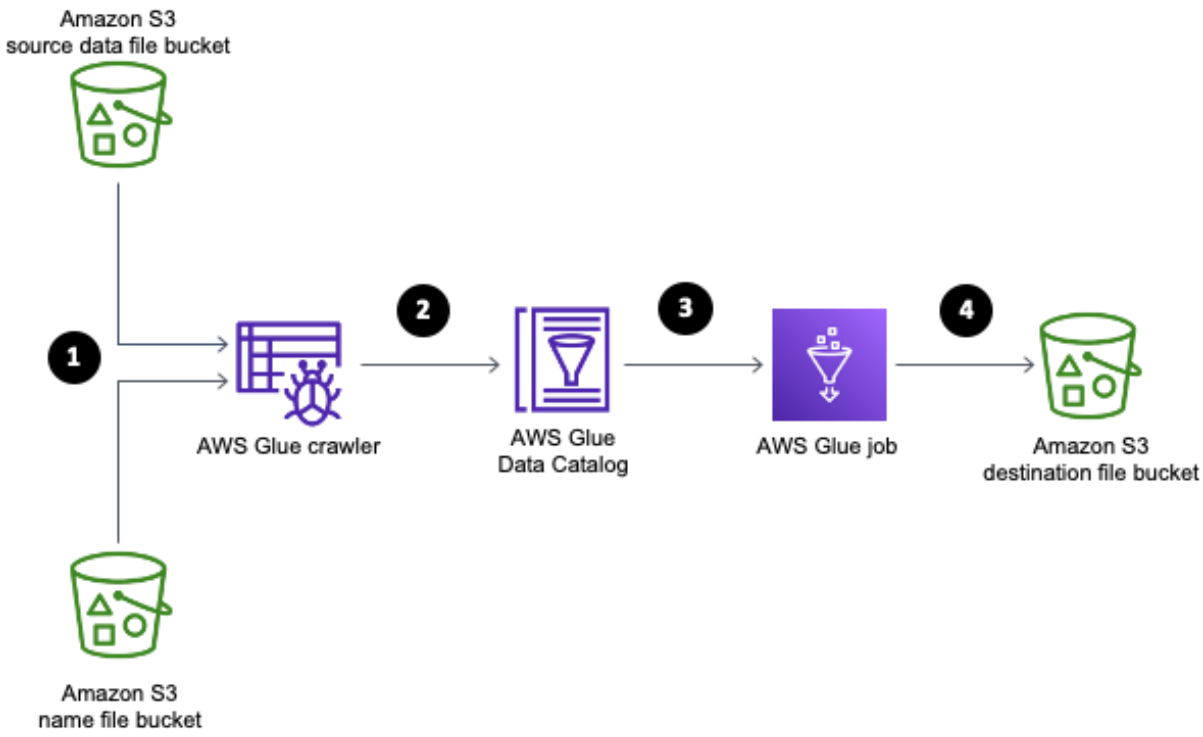

Ilustración 8, Funcionamiento rastreador (Crawler)

"Automate dynamic mapping and renaming of column names in data files using AWS Glue: Part 1 | AWS Big Data Blog," Mar. 15, 2021. https://aws.amazon.com/blogs/big-data/part-1-automating-dynamic-mapping-and-renaming-of-column-names-in-data-files-using-aws-glue/ (accessed Dec. 24, 2022).

Cuando se ejecuta un rastreador, realiza las siguientes acciones para interrogar a un almacén de datos:

- Clasifica los datos para determinar el formato, el esquema y las propiedades asociadas de los datos sin procesar: puede configurar los resultados de clasificación mediante la creación de un clasificador personalizado.
- Agrupa los datos en tablas o particiones: los datos se agrupan en función de la heurística de rastreador.
- Escribe los metadatos en el Data Catalog (Catalogo de datos): puede configurar cómo el rastreador agrega, actualiza y elimina tablas y particiones.

Al definir un rastreador, puede elegir uno o varios clasificadores que evalúen el formato de sus datos para inferir un esquema. Al ejecutarse el rastreador, el primer clasificador de su lista en reconocer correctamente su almacén de datos se usa para crear un esquema para su

tabla. Puede usar clasificadores integrados o definir los suyos propios. Puede definir sus clasificadores personalizados en una operación independiente, antes de definir los rastreadores. AWS Glue proporciona clasificadores integrados para inferir esquemas a partir de archivos comunes con formatos entre los que se incluyen JSON, CSV y Apache Avro [26].

En síntesis, los AWS Crawlers son una característica de AWS Glue que permite a los usuarios descubrir y extraer metadatos de diferentes fuentes de datos. Los crawlers de AWS se utilizan para explorar las fuentes de datos, identificar los datos relevantes y crear un catálogo de metadatos que se puede utilizar en las tareas de ETL de AWS Glue. Los crawlers de AWS son especialmente útiles en entornos en los que hay muchas fuentes de datos diferentes y es necesario integrar y analizar esos datos de manera eficiente. En general, los crawlers de AWS son una herramienta valiosa para simplificar y automatizar el proceso de integración y análisis de datos en la nube.

3.2.3. Data Lake

Muchos clientes de Amazon Web Services (AWS) requieren una solución de almacenamiento y análisis de datos que sea más ágil y flexible que los sistemas tradicionales de administración de datos. Un data lake es una modalidad nueva y cada vez más popular de almacenar y analizar datos porque permite a las empresas administrar múltiples tipos de datos de una amplia variedad de fuentes, y almacenar estos datos, estructurados y no estructurados, en un repositorio centralizado.

La nube de AWS proporciona muchos de los componentes esenciales necesarios para ayudar a los clientes a implementar un data lake seguro, flexible y rentable. Entre estos, se encuentra AWS Managed Services que permite incorporar, almacenar, buscar, procesar y analizar datos tanto estructurados como no estructurados. Con el objetivo de ayudar a los clientes durante el proceso de creación de un lago de datos, AWS ofrece Lago de datos en AWS, una implementación de referencia automatizada que implementa una arquitectura de lago de datos rentable y de alta disponibilidad en la nube de AWS junto con una consola de fácil uso para buscar y solicitar conjuntos de datos [27].

En síntesis, y como se muestra en la ilustración 9 de abajo, un data lake es un repositorio centralizado de datos en su forma más cruda y sin procesar. Los data lakes se utilizan para almacenar y gestionar grandes cantidades de datos de diferentes fuentes y tipos, y se pueden utilizar para diferentes propósitos, desde el análisis de datos hasta la ciencia de datos y la inteligencia artificial. Los data lakes ofrecen una serie de ventajas, como un almacenamiento económico y flexible, una capacidad de procesamiento de datos en paralelo y una facilidad de uso para diferentes equipos de datos. Sin embargo, también plantean desafíos, como la seguridad y la gobernanza de datos, y requieren una planificación cuidadosa y una gestión adecuada para obtener el máximo beneficio. En general, los data lakes son una herramienta valiosa para la gestión y el análisis de datos en la nube, pero requieren un enfoque cuidadoso y una planificación adecuada para aprovechar al máximo sus ventajas.

Ilustración 9, Servicios de Lago de datos

"Data Warehouse y Data Lake: ¿Qué son?," *Blog de Salesforce*. https://www.salesforce.com/mx/blog/2020/10/data-warehouse-y-data-lake.html (accessed Dec. 24, 2022).

3.2.4. Spark con AWS EMR

Amazon EMR es el mejor lugar para ejecutar Apache Spark. Puede crear rápida y fácilmente clústeres de Spark administrados con la consola de administración de AWS, la CLI de AWS o la API de Amazon EMR. Además, puede utilizar las características adicionales de Amazon EMR, que incluyen la conectividad rápida con Amazon S3 mediante el sistema de archivos de Amazon EMR (EMRFS), la integración con el mercado de spot de Amazon EC2,

el catálogo de datos de AWS Glue y el escalado administrado por EMR para añadir instancias al clúster o eliminarlas de él. AWS Lake Formation ofrece control pormenorizado del acceso, mientras que la integración con AWS Step Functions ayuda a organizar las canalizaciones de datos. EMR Studio (versión preliminar) es un entorno de desarrollo integrado (IDE, Integrated Development Environment) que facilita a los científicos e ingenieros de datos el desarrollo, la visualización y la corrección de aplicaciones de ingeniería y de ciencias de datos escritas en R, Python, Scala y PySpark. EMR Studio proporciona Jupyter Notebooks completamente administrado y herramientas como Spark UI y YARN Timeline Service para simplificar la depuración. EMR Notebooks facilita la realización de pruebas y la creación de aplicaciones con Spark. Si así lo prefiere, puede utilizar Apache Zeppelin para crear cuadernos interactivos y colaborativos a fin de analizar los datos con Spark [28].

En síntesis, AWS Elastic MapReduce (EMR) es un servicio de AWS que permite a los usuarios implementar y utilizar clusters de Apache Spark en la nube de forma sencilla y rápida. EMR proporciona una plataforma segura y escalable para el procesamiento de datos en paralelo utilizando Spark y otras herramientas de procesamiento de datos.

Como se puede ver en la ilustración 10 de abajo, al utilizar Spark con AWS EMR, es posible aprovechar la potencia y la eficiencia de Spark para el procesamiento de grandes volúmenes de datos sin tener que adquirir y configurar hardware y software localmente. Además, AWS EMR permite a los usuarios escalar rápidamente el uso de Spark según sea necesario y pagar solo por el uso efectivo de los recursos informáticos.

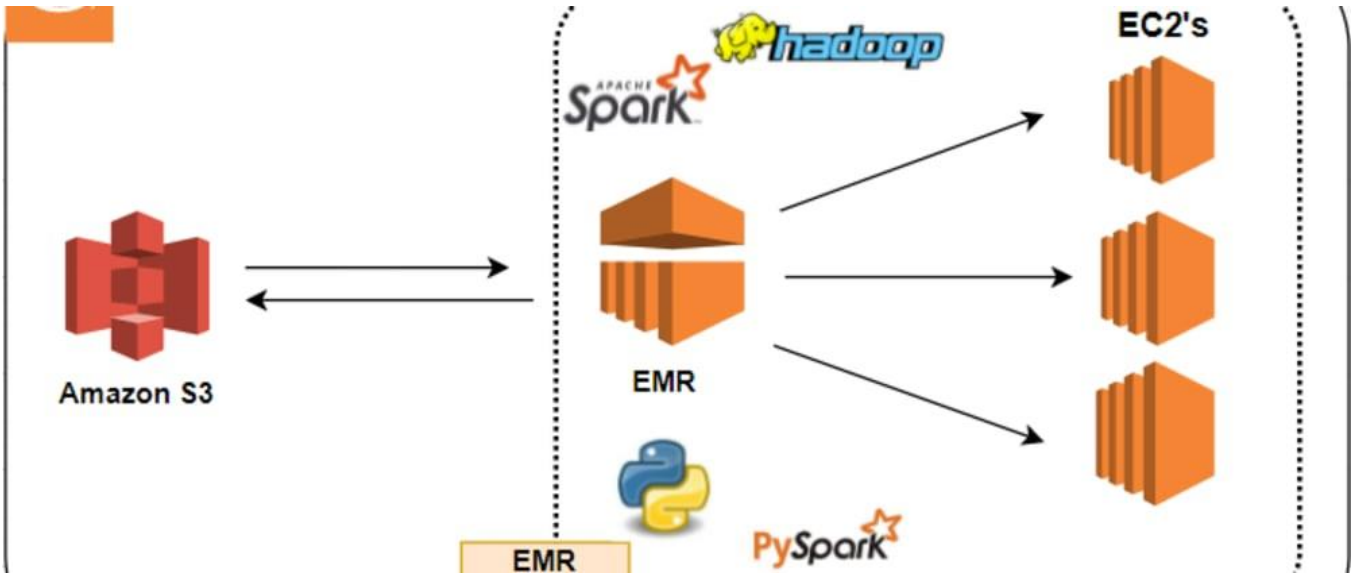

Ilustración 10, EMR y Spark con AWS

"Big Data Processing, EMR with Spark and Hadoop | Python, PySpark," *DEV Community* https://dev.to/wardaliaqat01/big-data-processing-emr-with-spark-and-hadoop-python-pyspark-4jo4 (accessed Dec. 24, 2022).

3.2.5. Amazon Athena

Amazon Athena es un servicio de consultas interactivo que facilita el análisis de datos en Amazon S3 con SQL estándar. Athena no tiene servidor, de manera que no es necesario administrar infraestructura y solo paga por las consultas que ejecuta.

Athena es sencillo de utilizar. Simplemente señale los datos en Amazon S3, defina el esquema y comience a realizar consultas con SQL estándar. La mayoría de los resultados se proporciona en cuestión de segundos. Con Athena, no es necesario realizar trabajos complejos de ETL para preparar los datos para el análisis. Por ello, cualquier persona con habilidades SQL puede analizar conjuntos de datos a gran escala de forma rápida y sencilla.

Athena se integra de serie con el catálogo de datos de AWS Glue, lo que le permite crear un repositorio de metadatos unificado en diversos servicios, rastrear orígenes de datos para descubrir esquemas y completar su catálogo con definiciones de particiones y tablas nuevas y modificadas, y mantener el control de las versiones de los esquemas [29].

En resumen, y como se muestra en la ilustración 11, AWS Athena es un servicio de consultas SQL en la nube de Amazon Web Services que permite a los usuarios ejecutar consultas SQL en datos almacenados en S3 Buckets. AWS Athena ofrece una forma rápida y sencilla de analizar y obtener insights de los datos almacenados en S3, y se puede utilizar para diferentes propósitos, desde la generación de informes y la toma de decisiones hasta la ciencia de datos y la inteligencia artificial. AWS Athena ofrece una serie de ventajas, como un alto rendimiento, una baja latencia y una facilidad de uso. Sin embargo, también plantea desafíos, como la gestión de metadatos y la optimización de consultas, y requiere una planificación cuidadosa y una gestión adecuada para obtener el máximo beneficio. En general, AWS Athena es una herramienta valiosa para el análisis de datos en la nube, pero requiere un enfoque cuidadoso y una planificación adecuada para aprovechar al máximo sus ventajas.

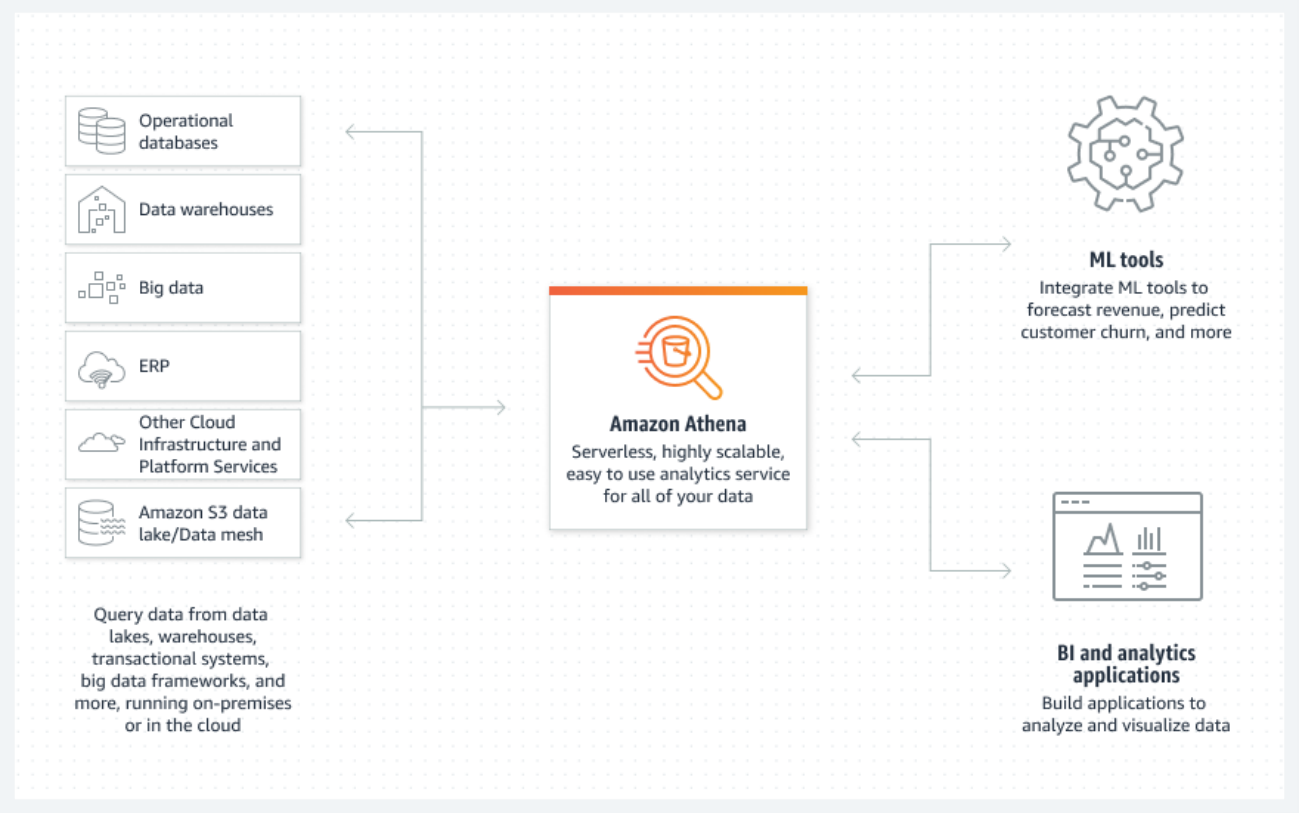

Ilustración 11, Funcionamiento de AWS Athena

"Consultas de datos al instante | Análisis de datos SQL | Amazon Athena," *Amazon Web Services, Inc.* https://aws.amazon.com/es/athena/ (accessed Dec. 24, 2022).

3.2.6. AWS Quicksight

Amazon QuickSight permite que todos los miembros de cualquier organización comprendan sus datos mediante preguntas en lenguaje natural, la exploración a través de paneles interactivos o la búsqueda automática de patrones y valores atípicos impulsada por machine learning.

Amazon QuickSight es un servicio de inteligencia empresarial impulsado por aprendizaje automático creado para la nube bajo el paraguas de Amazon Web Services . Permite a las empresas tomar decisiones más inteligentes basadas en datos [30].

La utilidad Amazon QuickSight BI permite a las empresas crear y analizar visualizaciones de datos y extraer información fácil de entender para informar la toma de decisiones comerciales. Estos tableros interactivos se pueden integrar sin problemas en muchas aplicaciones, portales y sitios web.

Amazon QuickSight es escalable, por lo que puede admitir miles de usuarios sin administración de infraestructura adicional o planificación de capacidad. También es independiente del dispositivo [30].

En síntesis, y como se muestra en la ilustración 12, AWS QuickSight es un servicio de visualización de datos en la nube de Amazon Web Services que permite a los usuarios crear y compartir dashboards y gráficos interactivos a partir de sus datos. AWS QuickSight ofrece una forma rápida y sencilla de analizar y obtener insights de los datos, y se puede utilizar para diferentes propósitos, desde la generación de informes y la toma de decisiones hasta la ciencia de datos y la inteligencia artificial. AWS QuickSight ofrece una serie de ventajas, como un rendimiento y una escalabilidad superiores, una integración con diferentes fuentes de datos y una facilidad de uso. Sin embargo, también plantea desafíos, como la seguridad y la gobernanza de datos, y requiere una planificación cuidadosa y una gestión adecuada para obtener el máximo beneficio. En general, AWS QuickSight es una herramienta valiosa para el análisis de datos en la nube, pero requiere un enfoque cuidadoso y una planificación adecuada para aprovechar al máximo sus ventajas.

Ilustración 12, Funcionamiento AWS Quicksight

"Use AnalyticsIQ with Amazon QuickSight to gain insights for your business | AWS Big Data Blog," Feb. 08, 2022. https://aws.amazon.com/blogs/big-data/use-analyticsiq-with-amazon-quicksight-to-gain-insights-for-your-business/ (accessed Dec. 24, 2022).

4. DESARROLLO METODOLÓGICO

El desarrollo metodológico de este trabajo es la parte de la investigación en la que se describe cómo se ha llevado a cabo el estudio o investigación. Incluye información sobre la metodología utilizada para analizar los datos, así como sobre el enfoque y los métodos que se han utilizado para abordar el problema de investigación.

4.1. Análisis metodológico

En este capítulo, el marco teórico incluirá una revisión bibliográfica de los principios y técnicas de diseño de pipelines de datos, que son secuencias de procesos que se utilizan para transformar y enriquecer grandes conjuntos de datos a medida que se mueven a través de ellos. También se revisarán estudios previos sobre el uso de pipelines de datos en el sector aeronáutico, es decir, en la industria de la aviación. Esta revisión se realizará con el fin de establecer un contexto y un marco de referencia para la investigación que se está llevando a cabo.

Como se menciona anteriormente en el documento, una hipótesis es que la tecnología en la nube es más eficiente y tiene ventajas competitivas en comparación con las tecnologías on-premise. La tecnología en la nube permite una mayor escalabilidad y flexibilidad en la gestión de los datos, lo que puede ser beneficioso para las aerolíneas que necesitan procesar grandes cantidades de datos. Además, el uso de la nube puede reducir los costos y la complejidad del sistema, lo que puede ser una ventaja importante en un mercado altamente competitivo como el de las aerolíneas.

Población y muestra: La población del estudio será compuesta por aerolíneas de la IATA de todo el mundo, en donde se incluirán aerolíneas de diferentes tamaños y que utilizan diferentes tecnologías de gestión de datos.

Recopilación de datos: Los datos de las aerolíneas se obtendrán de la International Air Transport Association (IATA), que es una asociación comercial que representa a más de 290 aerolíneas de todo el mundo.

Análisis de datos: Una vez que se hayan obtenido los datos de las aerolíneas, se procederá a su análisis. Para ello, se utilizará el algoritmo de Evaluación y Clasificación Binaria, que es un algoritmo que se utiliza para evaluar el rendimiento de los modelos de clasificación binaria.

Los resultados del análisis se graficarán y analizarán mediante AWS QuickSight, que es una plataforma de análisis y visualización de datos en la nube de Amazon Web Services. AWS QuickSight permite crear visualizaciones y dashboards personalizados a partir de grandes conjuntos de datos, y proporciona una amplia gama de herramientas para explorar y analizar los datos de manera sencilla y rápida.

4.2. Requerimientos para el pipeline

Para crear un pipeline de ingesta de datos en AWS que utilice S3 bucket, AWS Glue con crawlers, AWS EMR, AWS Athena y AWS QuickSight, se necesitarán los siguientes requerimientos:

1. Una cuenta de AWS con acceso a todos los servicios requeridos para el pipeline.
2. Un conjunto de datos de la IATA que se desee analizar y almacenar en un S3 bucket.
3. Un crawler de AWS Glue que se encargue de descubrir y extraer información de los datos de la IATA almacenados en el S3 bucket.
4. Un clúster de AWS EMR para procesar y analizar los datos obtenidos por el crawler de AWS Glue.
5. Una instancia de AWS Athena para realizar consultas SQL en los datos procesados por AWS EMR.
6. Una instancia de AWS QuickSight para visualizar y analizar los resultados obtenidos por AWS Athena.

4.3. Servicios usados para el pipeline

Como se muestra a continuación en el diagrama de arquitectura de AWS en la ilustración 13 de abajo, los servicios usados para este trabajo son los siguientes:

- Se hace uso de la nube AWS (Amazon Web Services)
- Se hace en un VPC que es la Virtual Private Cloud

    - Una VPC (Virtual Private Cloud) es una red virtual privada en la nube que se utiliza para implementar y ejecutar aplicaciones y servicios en la nube de manera segura y aislada. Una VPC es necesaria para un pipeline de AWS por varias razones. Primero, permite a los usuarios definir una red lógica y aislada en la nube, lo que les permite controlar el acceso y la seguridad de sus aplicaciones y servicios. Esto es especialmente importante en un pipeline de AWS, ya que implica el manejo de grandes cantidades de datos sensibles y la integración de diferentes servicios y aplicaciones. Segundo, una VPC permite a los usuarios conectarse de manera privada y segura a sus recursos en la nube, lo que mejora la confiabilidad y el rendimiento de sus aplicaciones y servicios. Por último, una VPC permite a los usuarios implementar sus aplicaciones y servicios en diferentes regiones y zonas de disponibilidad de AWS, lo que les permite aprovechar la escalabilidad y la resiliencia de la nube. En general, una VPC es una parte fundamental de un pipeline de AWS y es necesaria para garantizar la seguridad, la confiabilidad y el rendimiento de los servicios en la nube.

- Dentro del availability zone de Virginia
- Servicios como el S3 Bucket para el almacenamiento de datos
    - Los buckets de S3 se utilizan por varias razones. Primero, son una forma económica y flexible de almacenar datos en la nube, ya que ofrecen una capacidad ilimitada y un pago por uso. Segundo, son escalables y resistentes a fallos, ya que se distribuyen en varios centros de datos y zonas de disponibilidad dentro de una región de AWS. Esto permite a los usuarios confiar en la disponibilidad y el rendimiento de sus datos en la nube. Tercero, son versátiles y compatibles con diferentes tipos de datos, desde archivos

binarios hasta texto simple, lo que permite a los usuarios almacenar y gestionar una gran variedad de datos en la nube.

- Crawler
    - Los crawlers se utilizaron para examinar los datos y crear esquemas para los mismos en el catálogo de AWS Glue. Esto es útil porque proporciona una forma de acceder a los datos de una manera estructurada y poder ser utilizado por Amazon Athena para realizar consultas o analizar los datos.
    - En resumen, los crawlers de AWS Glue se utilizaron para examinar y crear un esquema para los datos, lo que proporciona una forma de acceder a ellos de manera más fácil y puede ser utilizado por el siguiente servicio en el pipeline.
- AWS Glue
    - AWS Glue es un servicio de ETL (Extract, Transform, Load) en la nube que permite a los usuarios integrar y analizar datos de diferentes fuentes de manera rápida y sencilla.
    - Ofrece una plataforma unificada para la integración y el análisis de datos en la nube, lo que simplifica y automatiza los procesos de ETL y mejora la eficiencia de los análisis de datos.
    - Es escalable y resistente a fallos, ya que se ejecuta en varios centros de datos y zonas de disponibilidad dentro de una región de AWS. Esto permite a los usuarios confiar en la disponibilidad y el rendimiento de sus tareas de ETL en la nube.
    - Es compatible con diferentes tipos de datos y fuentes, desde bases de datos relacionales hasta archivos en S3, lo que permite a los usuarios integrar y analizar una gran variedad de datos en la nube.
    - Se integra con otros servicios de AWS, como S3, Athena y Redshift, lo que permite a los usuarios construir pipelines de datos complejos y utilizar los resultados de los análisis de datos en aplicaciones en la nube. En general, AWS Glue es una herramienta valiosa para la integración y el análisis de datos en la nube que puede ayudar a las empresas a maximizar el valor de sus datos y aumentar la eficiencia de sus procesos de negocio.
- AWS EMR
    - Se inicia un cluster de EMR: Una vez que los datos estén disponibles en S3, puede iniciar un cluster de EMR utilizando la consola de AWS de EMR. Ejecuté un job de MapReduce en el cluster de EMR: Una vez que el cluster esté en ejecución, puede ejecutar un job de MapReduce utilizando el lenguaje

de pySpark. El trabajo de MapReduce procesará los datos almacenados en S3 y producirá un conjunto de resultados.

- Amazon Athena con database job
    - Después de que los datos se envíaran a un bucket de S3 para su almacenamiento, el job de AWS Glue se encarga de crear un esquema para los datos almacenados en S3 y de cargar esos datos en una tabla de Athena, la cual se puede utilizar para realizar consultas SQL sobre los datos almacenados en S3 a través de la interfaz de Athena.
    - Los resultados de las consultas se pueden utilizar para crear informes, visualizaciones de datos o para tomar decisiones en tiempo real.
- Amazon Quicksight para el modelado de estos datos
    - Se conectó QuickSight a la fuente de datos y desde la consola se empezaron a crear los Dashboards y visualizaciones. Creación del informe y dashboards: Una vez que los datos estuvieron disponibles en QuickSight, se utilizó la interfaz de usuario de arrastrar y soltar para crear informes y dashboards y visualizaciones. Se agregaron gráficos, tablas y otros elementos visuales a los informes y dashboards para hacer que los datos sean más fáciles de entender y analizar. Informes y dashboards: Una vez creados los dashboards, se pudieron interpretar los gráficos a simple vista, ya que es muy fácil identificar una tendencia o un resultado en las gráficas de barras o de pastel. Diagrama de arquitectura del pipeline de ingesta de datos en AWS

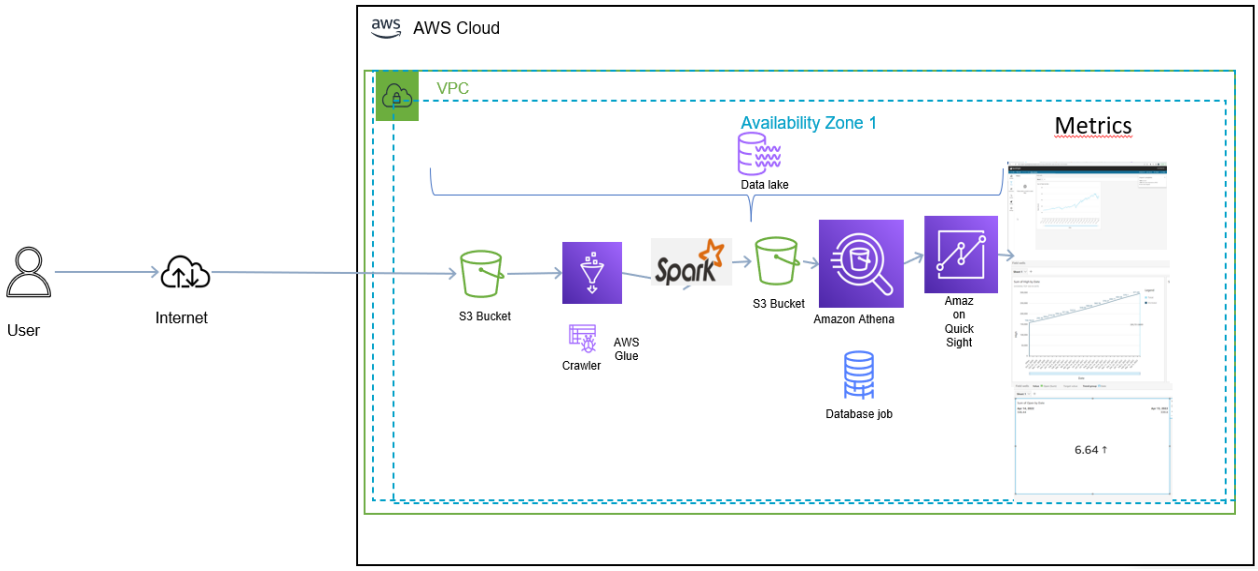

Ilustración 13 Arquitectura del pipeline de AWS. Diagrama Propio

En resumen, se siguieron los siguientes puntos en forma general para su creación, esto es a muy alto nivel, además las ilustraciones pueden ser encontradas en la sección de Anexos, en el capítulo final del documento.

- Se hace la selección de la Regio y la zona de disponibilidad
- Creación de los Buckets
- Creación del rastreador (crawler)
- Creación de los jobs necesarios
- Creación de las tablas
- Lista de los Jobs
- Tranformar y mapear los datos
- Se crean los queries para la visualización y vistas

5. CONCLUSIONES Y RESULTADOS

En este capítulo se presentan las conclusiones y trabajo futuro con relación al objeto de estudio, al igual que una breve revisión de los objetivos del estudio y cómo han sido cumplidos, discutir los hallazgos más relevantes y cómo contribuyen al conocimiento existente en el campo, señalar cualquier limitación del estudio y sugerir posibles áreas de investigación futura

5.1. Conclusiones generales del proyecto

Por medio de este trabajo se logró desarrollar un pipeline de datos con AWS con el primer paso que fue encontrar el data set de vuelos de la IATA descargados de la página oficial https://www.iata.org/, al igual se logró desarrollar la creación de todos los servicios necesarios de AWS , por lo que primero se creó un S3 bucket y se observó cómo funcionan los crawlers para rellenar el catálogo de datos de AWS Glue con tablas.

Este es el método principal utilizado por la mayoría de los usuarios de AWS Glue. Un rastreador puede rastrear múltiples almacenes de datos en una sola ejecución. Al finalizar, el rastreador crea o actualiza una o más tablas en su catálogo de datos. Los trabajos de extracción, transformación y carga (ETL) que define en AWS Glue utilizan estas tablas del catálogo de datos (Data Catalog) como orígenes y destinos. El trabajo de ETL lee y escribe en los almacenes de datos que se especifican en las tablas de catálogo de datos de origen y de destino.

Después al crear los Jobs de AWS Glue y en uno de ellos se introduce el código de Spark para hacer un algoritmo de ML, esto con el fin de hacer una predicción de si la tendencia de los vuelos va a seguir.

Después se hacen las tablas de AWS Athena para hacer los *queries* necesarios para la analítica.

A lo que después se cargan esos datos en AWS quicksight.

Posteriormente se hacen las gráficas en Quicksight para sacar las conclusiones sobre que aerolíneas son las que más demora y menos demora tienen.

Con estas gráficas presentadas en la ilustración 16 de abajo, luego de realizar el estudio a través de los datos se ha determinado que el mejor día para viajar si el retraso de llegada es un factor importante para el usuario es el sábado. Esto se debe a que, en promedio, los vuelos programados para llegar los sábados presentan menores tasas de retraso en comparación con los demás días de la semana. Por otro lado, el peor día para viajar en términos de retraso de llegada es el jueves, con una tasa de retraso significativamente mayor en comparación con los demás días.

En cuanto a los retrasos en la salida, se ha determinado que el peor día para viajar es el día 10 de cualquier mes, mientras que el mejor día es el 31. Sin embargo, es importante tener en cuenta que este resultado podría estar sesgado ya que no todos los meses tienen 31 días, por lo que es posible que haya menos vuelos programados para el 31 en comparación con otros días. En segundo lugar, el día con menores tasas de retraso en la salida es el 4 de cualquier mes. En general, estos resultados sugieren que, si el retraso en la llegada o la salida es un factor importante para el usuario, puede ser beneficioso programar su vuelo para los sábados o el último día del mes.

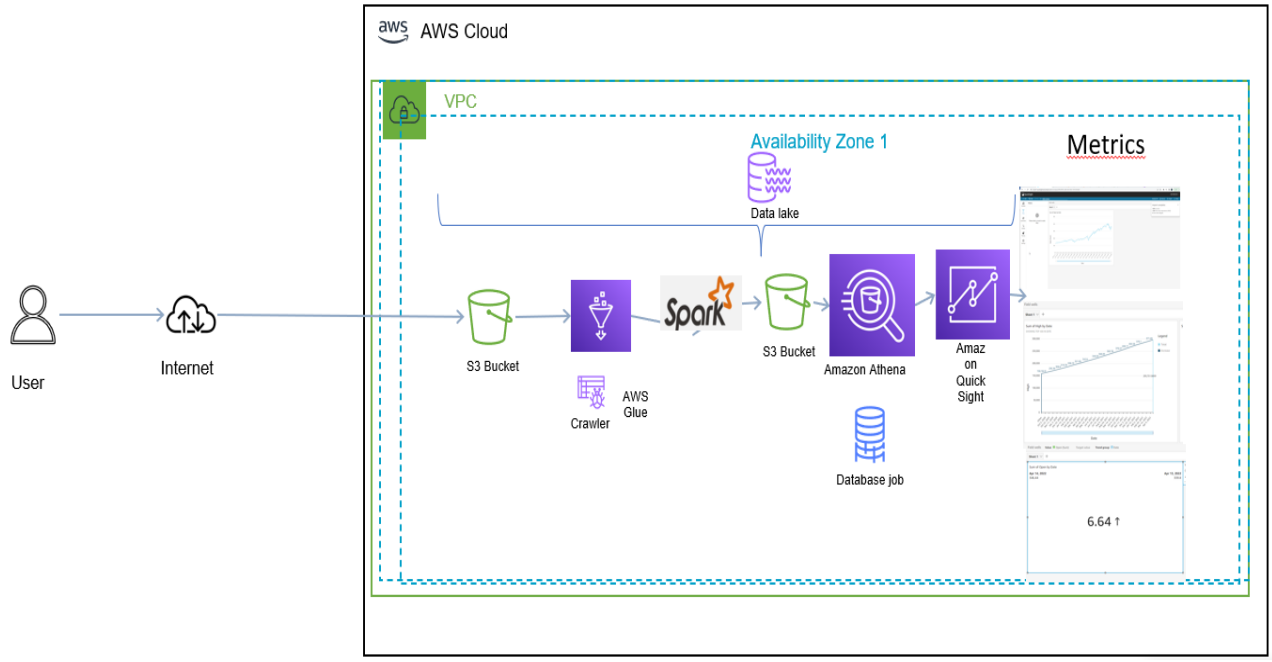

Ilustración 14, Arquitectura de AWS Pipeline.

Diagrama Propio

Diagrama de arquitectura en donde se especificaba el uso de los siguientes servicios de AWS

- AWS S3 Bucket
- AWS Glue
    - With Crawler
- Spark ML model
- AWS Athena
- AWS Quicksight

5.2. Resultados Código de Spark con pyspark

Spark ML con el modelo “BinaryClassificationEvaluator model”

Precisión mayor al .90 en el modelo presentado en la ilustración 30, esto demostrando que la tendencia continuara con los mismos resultados en el futuro

AUR2 = 0.9228699408935557

El código proporcionado había recalculado el área bajo la curva ROC (Receiver Operating Characteristic) para un conjunto de datos de predicción utilizando el BinaryClassificationEvaluator de Spark. El área bajo la curva ROC es una medida de la

calidad de un modelo de clasificación binaria y se calcula(mostrado en la sección de Anexos) a partir de la tasa de verdaderos positivos y la tasa de falsos positivos. El resultado del cálculo se imprimió en pantalla con la etiqueta "AUR2" y su valor correspondiente.

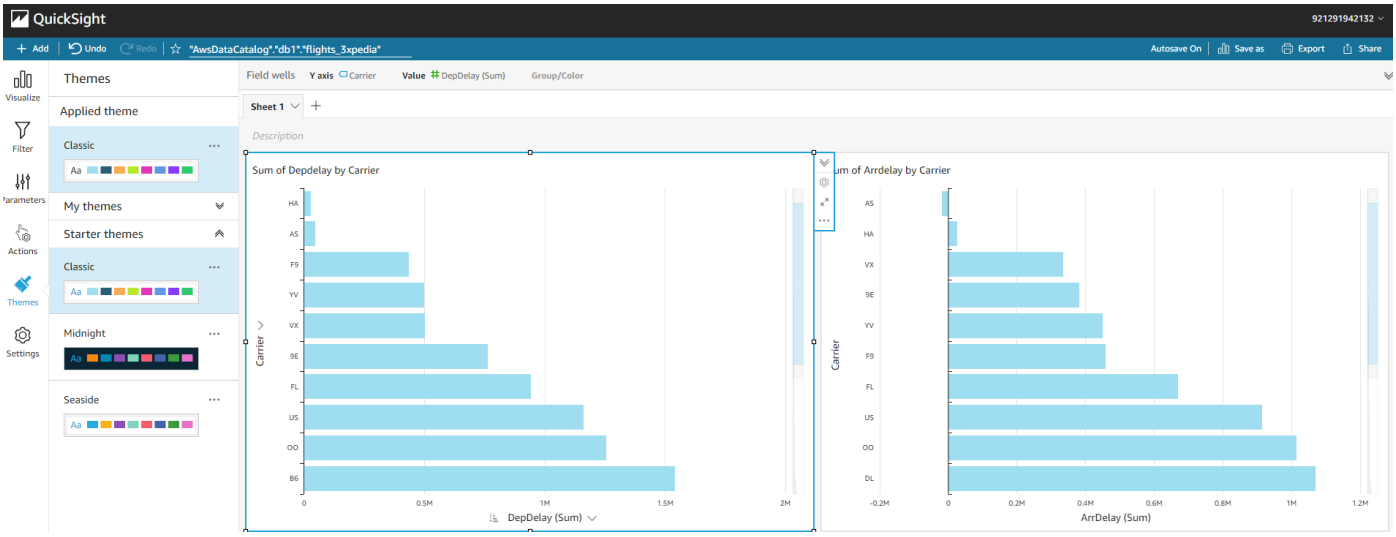

Ilustración 15, Métricas de aerolíneas con demora en AWS Quicksight.

Métricas propias creadas en AWS Quicksight

Las mejores aerolíneas para viajar si el retraso en la salida es importante para el usuario son:

1. Hawai airlines
2. Alaska airlines
3. Frontier airlines(f9)
4. Vueling airlines s.a.
5. Virgin america
6. Endeavor air
7. Airtran airways
8. United airlines
9. Skywest airlines
10. Jetblue airways corporation

Las mejores aerolíneas para viajar si el retraso en la llegada es importante para el usuario son:

1. Alaska airlines
2. Hawai airlines
3. Virgin america
4. Endeavor air
5. Vueling airlines s.a.

6. Frontier airlines(f9)
7. Airtran airways
8. United airlines
9. Skywest airlines
10. Delta air lines, inc

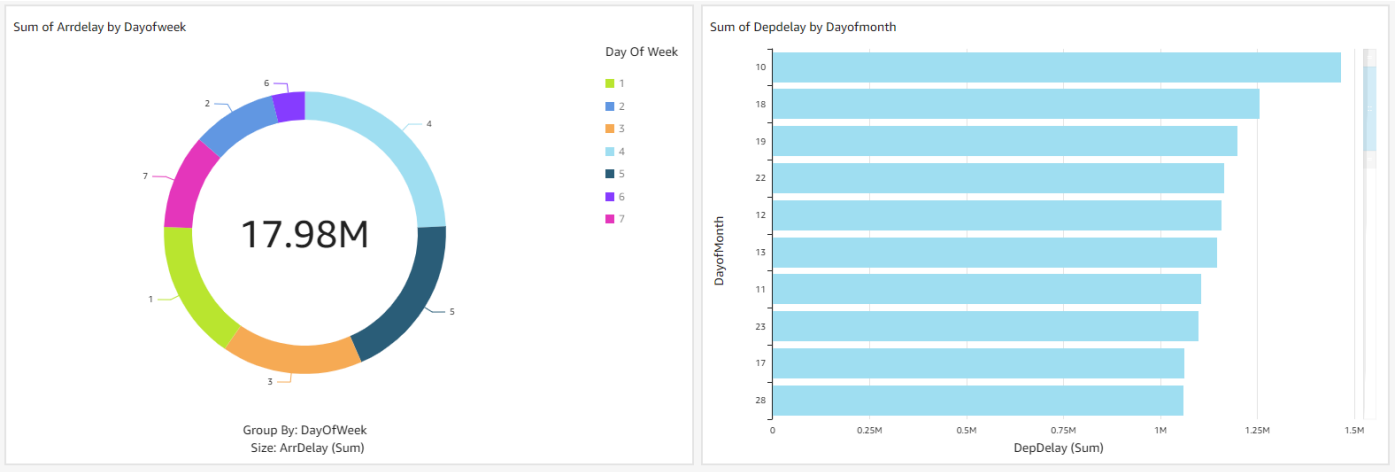

Ilustración 16, Demoras de vuelos por día del mes y por semana

Métricas propias creadas en AWS Quicksight

Según las gráficas, el mejor día para viajar si el retraso de llegada bajo es importante para el usuario es el sábado y el peor día es el jueves.

Y por igual, el peor día para viajar debido a la gran demora en la salida es el día 10 del mes, el mejor día es el 31 (sesgo) y luego el 4 del mes, el sesgo siendo porque no todos los meses tienen 31 días.

5.3. Costos de los serivios de AWS

Se espera generar costos por los siguientes conceptos

- Creación de los Buckets (.60 USD por cada S3 bucket por mes)
- S3 Bucket $0.023 por GB, 70MB por cada file cada mes
- AWS Glue < 1 $
- Athena: < 5$

- Glue Developer Endpoint < 1 $
- Quicksight Edición Standard • 26 USD por usuario por mes con suscripción anual

5.4. Conclusiones generales y objetivos completados

Parte de las conclusiones es mencionar si se cumplieron los objetivos generales y específicos, como se muestra a continuación.

Descripción de cumplimiento del objetivo general:

Para cumplir con el objetivo de crear un pipeline de ingesta de datos en AWS, primero fue necesario obtener los datos que se iban a utilizar para el análisis.

Obtención de datos: El primer paso fue obtener el dato que necesitaríamos para el análisis. Este dato pudo provenir de diversas fuentes, como bases de datos de aerolíneas, registros de vuelos, encuestas de pasajeros, etc. Es importante asegurarnos de que el dato fuera completo y preciso.

Almacenamiento de datos: Una vez que habíamos obtenido el dato, era necesario almacenarlo de manera segura y accesible. Para ello, se podía utilizar una solución de almacenamiento en la nube como AWS S3.

Procesamiento de datos: Luego, era necesario procesar el dato para prepararlo para el análisis. Esto podía incluir tareas como la limpieza de datos

Análisis de los datos: Una vez que habíamos procesado los datos, era necesario analizarlos para obtener la métrica deseada. Esto podía incluir el uso de técnicas de ciencia de datos y algoritmos de aprendizaje automático para identificar patrones y tendencias en los datos. Para realizar este análisis, se podían utilizar herramientas como AWS SageMaker o AWS EMR.

Visualizamos los resultados: Finalmente, es importante presentar los resultados de manera clara y fácilmente comprensible. Para ello, se podía utilizar una herramienta de visualización de datos como AWS Quicksight para crear gráficos y dashboards que mostraran los resultados de manera atractiva y fácil de interpretar.

Se desarrollo un pipeline de datos que utilice tecnología en la nube: Para cumplir con este objetivo, se desarrolló un pipeline de datos que utilizara tecnologías en la nube como AWS S3 y AWS Glue. Esto permitió almacenar y procesar los datos de manera segura y eficiente, utilizando las ventajas de la nube como la escalabilidad y la flexibilidad.

Mejorar la eficiencia en la transferencia de datos a través del pipeline: Una vez que se tenía el pipeline de datos en funcionamiento, se enfocó en mejorar su eficiencia. Para ello, se optimizó la transferencia de datos a través del pipeline utilizando técnicas de optimización de rendimiento y configuraciones óptimas de los componentes del pipeline.

Incrementar la flexibilidad y la escalabilidad del pipeline: Otro objetivo que se cumplió fue aumentar la flexibilidad y la escalabilidad del pipeline. Para ello, se diseñó el pipeline de manera que pudiera adaptarse fácilmente a diferentes requerimientos y cargas de trabajo, utilizando componentes de la nube que permitieran escalar de manera eficientes

Optimizamos el pipeline de ingesta de datos mediante servicios de AWS más usados en la industria, como lo son los S3 buckets o AWS Glue para maximizar la eficiencia y minimizar los errores: También nos enfocamos en optimizar el pipeline de ingesta de datos para maximizar su eficiencia y minimizar los errores. Para ello, implementamos prácticas de buen diseño de datos y utilizamos herramientas de monitoreo y depuración para detectar y corregir posibles problemas.

Diseñamos un modelo de datos mediante AWS Quicksight que permitiera estructurar y procesar los datos de manera eficiente para su análisis: Un objetivo clave fue diseñar un modelo de datos que nos permitiera estructurar y procesar los datos de manera eficiente para su análisis. Para ello, utilizamos técnicas de modelado de datos y estandarizamos la estructura de los datos para facilitar su procesamiento y análisis.

Investigamos las herramientas de tecnología en la nube que eran más adecuadas para la creación del pipeline de ingesta de datos: Antes de iniciar la creación del pipeline de ingesta de datos, investigamos cuáles eran las herramientas de tecnología en la nube más adecuadas para nuestros propósitos. Esto nos permitió seleccionar las herramientas más adecuadas para nuestro pipeline y asegurarnos de que cumplieran con nuestros requerimientos.

6. Trabajo Futuro

Los trabajos futuros que se pretenden generar a partir de este pipeline en la nube y los datos ya transformados para su visualización, sería un nuevo feature o capacidad en una aplicación mobil en el cual se podría dar un paso extra al usuario para que tenga más herramientas para seleccionar sus vuelos. A continuación, los pasos a seguir:

1. El usuario entra a la aplicación móvil de la aerolínea

Funcionalidades de usuario: El usuario abre la aplicación móvil de la aerolínea en su teléfono inteligente y, al hacerlo, se le presenta una pantalla de inicio que le permite acceder a diferentes funcionalidades y servicios. En esta pantalla, el usuario puede ver una serie de opciones, como reservar un vuelo, revisar el estado de un vuelo, acceder a su perfil de usuario y mucho más.

Una vez que el usuario ha entrado a la aplicación, puede navegar por ella de manera sencilla y cómoda, utilizando diferentes menús y opciones para acceder a la información y servicios que necesita

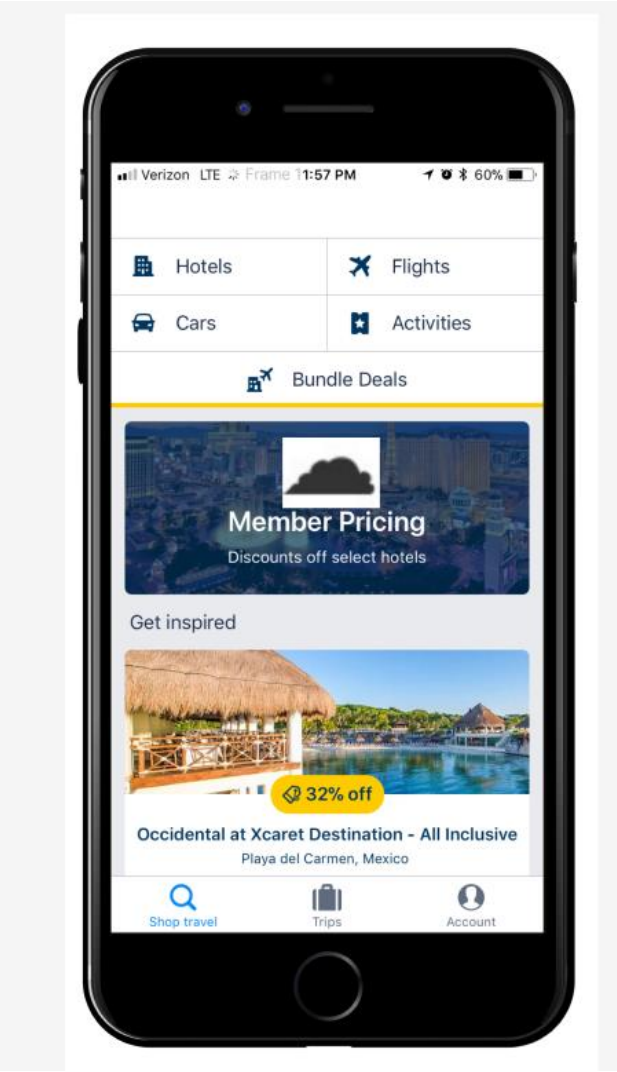

Ilustración 17, Prototipo 1.

Creación propia

2. Procede a seleccionar horarios de los vuelos y la aerolínea tentativa

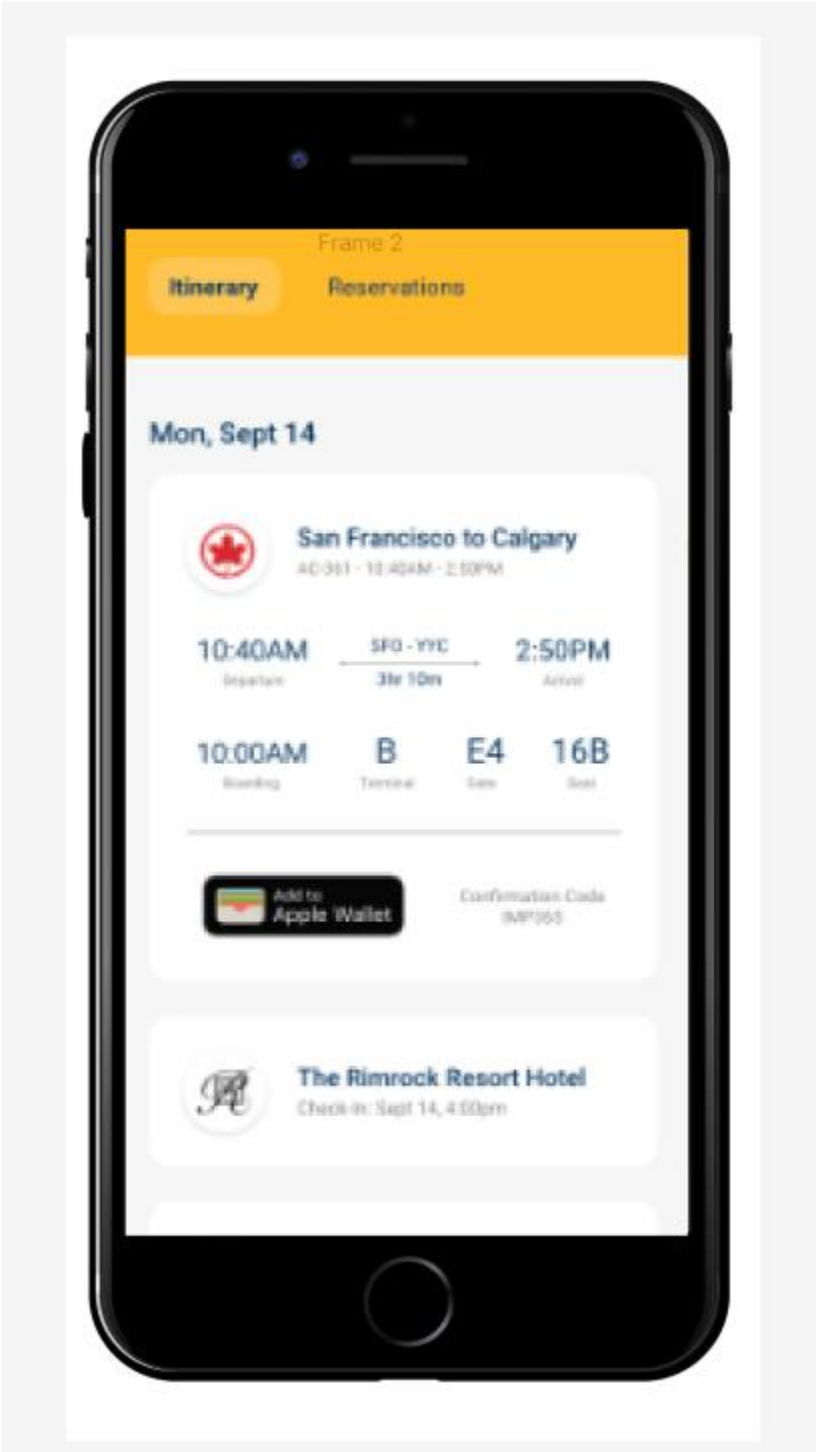

Ilustración 18, Prototipo 2

Creación propia

El usuario navega hasta el sitio web de reservas de vuelos y hace clic en el botón "Buscar vuelos".

El usuario se presenta con un formulario donde puede ingresar sus ciudades de salida y llegada, así como sus fechas de viaje preferidas.

El usuario ingresa su información de viaje deseada y hace clic en el botón "Buscar".

El sitio web muestra una lista de vuelos disponibles que cumplen con los criterios del usuario. La lista incluye información como el horario de vuelo, la aerolínea, la duración y el precio.

El usuario revisa los vuelos disponibles y selecciona el que mejor se ajusta a sus necesidades. Hace clic en el botón "Reservar" junto al vuelo elegido.

3. Al momento de hacer click en “book now”, y antes de proceder al pago

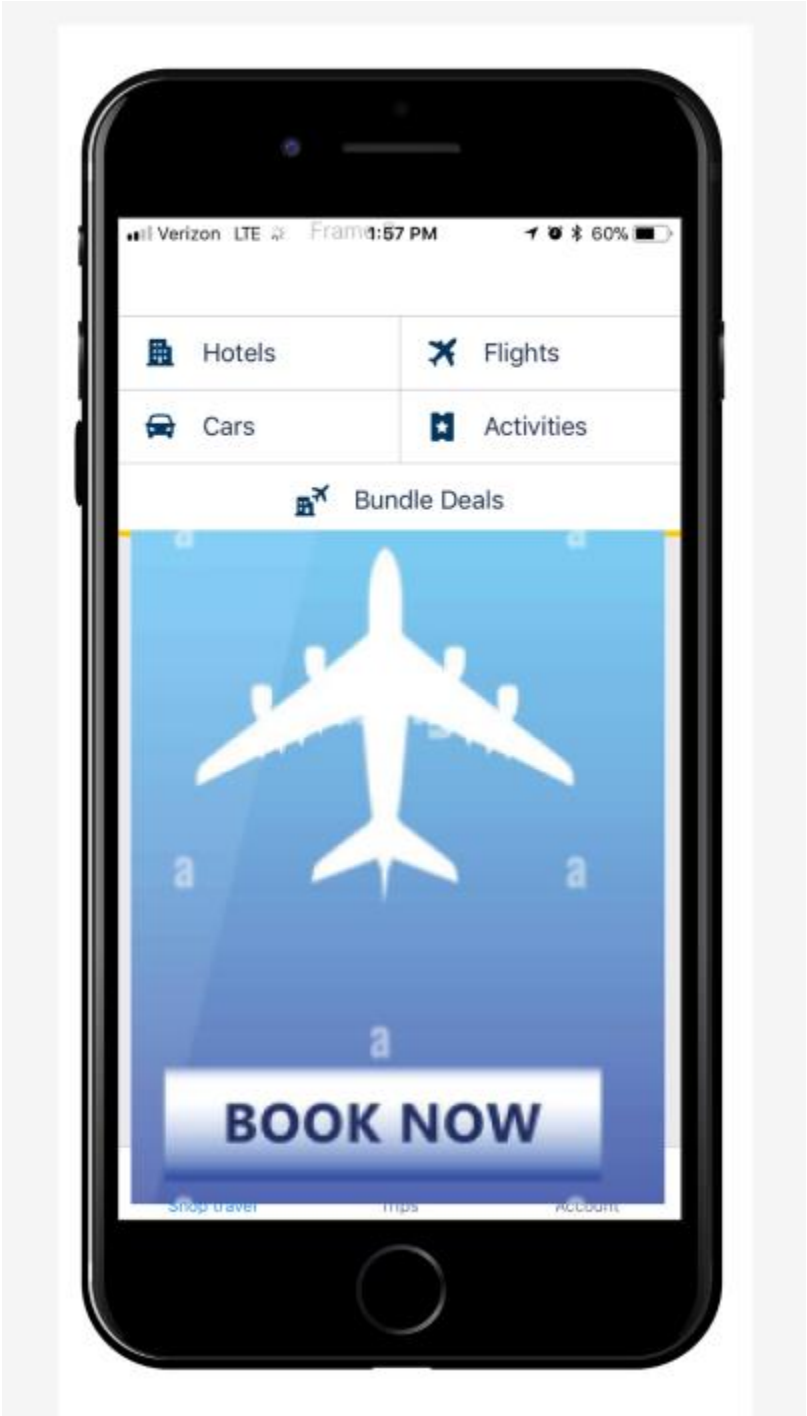

Ilustración 19, Prototipo 3

Creación propia

4. El usuario recibe una alerta de que esta aerolínea tiene posibilidad de que tenga demoras en llegadas y demoras en salidas de acuerdo a la estadística de la IATA

Ilustración 20, Prototipo 4

Creación propia

La alerta se envía al usuario para que pueda tomar medidas adecuadas en caso de que prevé viajar con esta aerolínea. Esta información es importante porque puede afectar el tiempo que el usuario dedicará a sus actividades durante el viaje, y también puede afectar su programación en general. Por ejemplo, si el usuario tenía planes de tomar un vuelo de esta aerolínea para llegar a un compromiso importante, es posible que deba cambiar sus planes si sabe de antemano que hay una alta posibilidad de retrasos. Además, si el usuario ya ha comprado un billete con esta aerolínea y recibe esta alerta, puede optar por cambiar su vuelo a otra aerolínea para evitar los posibles retrasos. En cualquier caso, es importante que el usuario esté al tanto de la posibilidad de retrasos y tome las medidas necesarias para minimizar su impacto en sus planes.

5. El usuario puede ver en qué posición se encuentra esta aerolínea en comparación con las demás que ofrecen el mismo vuelo, y se da la opción de elegir otra aerolínea con mejores métricas

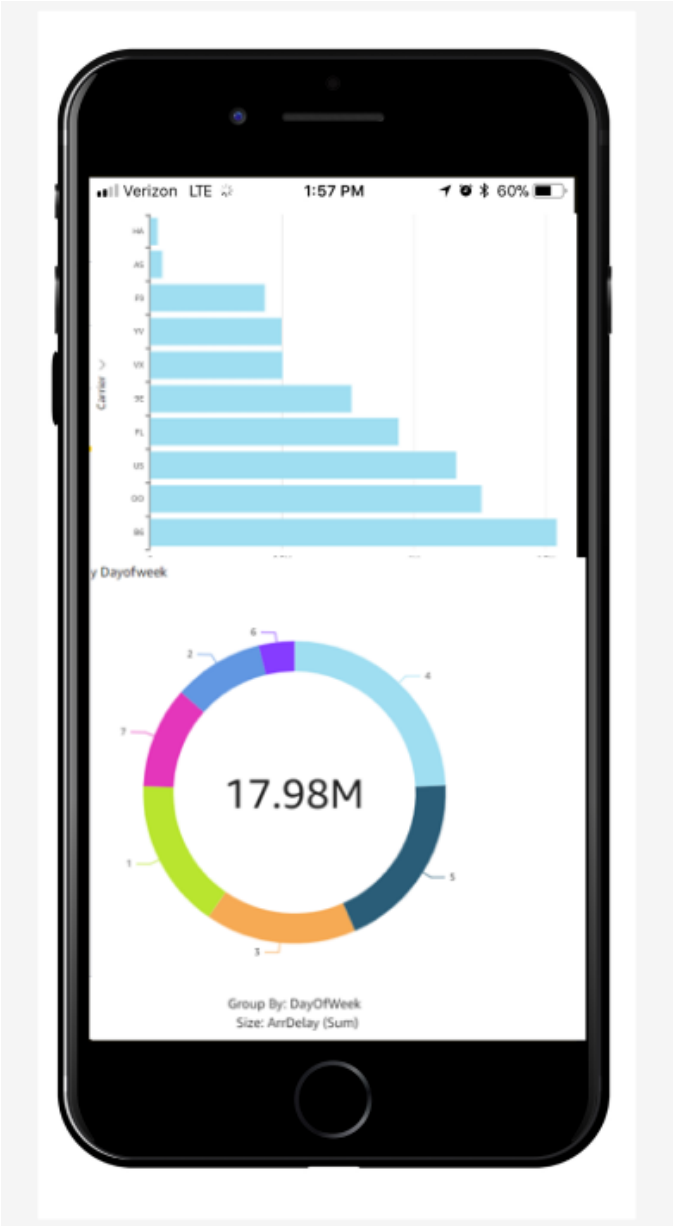

Ilustración 21, Prototipo 5

Creación propia

6. Después de esto, el usuario puede proceder al pago

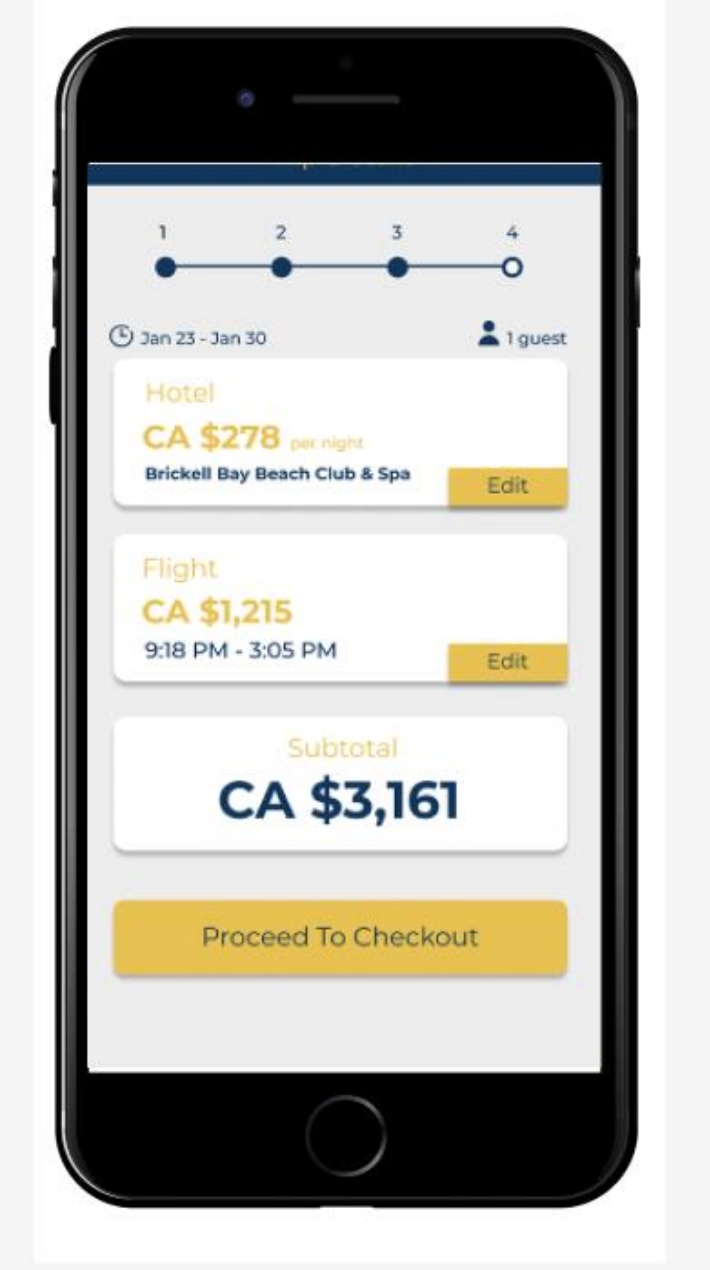

Ilustración 22, Prototipo 6

Creación propia

La aplicación muestra un formulario donde el usuario puede ingresar su información personal y de pago.

El usuario ingresa su información y hace clic en el botón "Reservar ahora".

La aplicación confirma la reserva y muestra una página de confirmación con los detalles del vuelo y un número de confirmación.

El usuario recibe una confirmación por correo electrónico con los detalles del vuelo y un enlace para ver o cancelar la reserva.

7. BIBLIOGRAFÍA

[1] A. Lomas Redondo. "Creación de un pipeline para análisis de datos metagenómicos basado en Nextflow". Doctoral dissertation, Universitat Politècnica de València, 2021.

[2] “What Is ETL?” https://www.sas.com/es_mx/insights/data-management/what-is-etl.html (accessed Nov. 02, 2022).

[3] “¿Qué es cloud?,” Mar. 17, 2022. https://www.ibm.com/mx-es/cloud/learn/cloud-computing (accessed Nov. 02, 2022).

[4] S. R. Montoya, “Así avanza la computación en la nube, un valioso recurso para las empresas,” Entrepreneur, May 10, 2022. https://www.entrepreneur.com/es/noticias/asi-avanza-la-computacion-en-la-nube-un-valioso-recurso/427238 (accessed Nov. 02, 2022).

[5]“Aceleración de las transacciones de pago de proveedores de 1 día a segundos | Caso práctico de Expedia Group | AWS,” Amazon Web Services, Inc. https://aws.amazon.com/es/solutions/case-studies/expedia-aurora-case-study/ (accessed Nov. 02, 2022).

[6] Sternberg A, Soares J, Carvalho D, Ogasawara E. A Review on Flight Delay Prediction. arXiv preprint arXiv:1703.06118. 2017 Mar 15.

[7] F. Wang, T. Xu and S. Li, "Research on cloud platform architecture of flight big data," 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA), 2021, pp. 170-174, doi: 10.1109/ICPECA51329.2021.9362529.

[8] R. Nigam and K. Govinda, "Cloud based flight delay prediction using logistic regression," *2017 International Conference on Intelligent Sustainable Systems (ICISS)*, 2017, pp. 662-667, doi: 10.1109/ISS1.2017.8389254.

[9] Big Data Research and Development Initiative [DB/OL]. [2012-03-29] . http://www. whitehouse.gov/sites/default/files/microsites/ostp/big_data_press_release_final_2.pdf .

[10] Big Data for Development: Challenges & Opportunities [DB/OL].[2012 -05 -01]. http://www. unglobalpulse.org /sites/default/files/BigData for Development - UNGlobalPulseJune2012.pdf.

[11] Big Data Across the Federal Government [EBIOI]. [2012-10-02].http://www.whitehouse.gov/sites/default/files/mierosites/ostp/big_datafact-sheet-final_1.pdf.

[12] G. Dartmann, H. Song, and A. Schmeink, Big Data Analytics for Cyber-Physical Systems: Machine Learning for the Internet of Things. Elsevier, 2019.

[13] Z. Lv, H. Song, P. Basanta-Val, A. Steed, and M. Jo, “Next-generation big data analytics: State of the art, challenges, and future research topics,” IEEE Transactions on Industrial Informatics, vol. 13, no. 4, pp. 1891–1899, 2017.

[14] Y. Sun, H. Song, A. J. Jara, and R. Bie, “Internet of things and big data analytics for smart and connected communities,” IEEE Access, vol. 4, pp. 766–773, 2016.

[15] F. Kong, J. Li, B. Jiang, T. Zhang, and H. Song, “Big data-driven machine learning-enabled traffic flow prediction,” Transactions on Emerging Telecommunications Technologies, vol. 30, no. 9, p. e3482, 2019.

[16] C. M. Ariyawansa and A. C. Aponso, "Review on state of art data mining and machine learning techniques for intelligent Airport systems," 2016 2nd International Conference on Information Management (ICIM), 2016, pp. 134-138, doi: 10.1109/INFOMAN.2016.7477547.

[17] OECD, International Regulatory Co-operation: The Role of International Organisations in Fostering Better Rules of Globalisation. OECD, 2016. doi: 10.1787/9789264244047-en.

[18] “Vision and Mission.” https://www.iata.org/en/about/mission/ (accessed Nov. 02, 2022).

[19] “¿Qué es la regresión lineal?” https://la.mathworks.com/discovery/linear-regression.html (accessed Nov. 02, 2022).

[20] “AWS | Informática en la nube. Ventajas y Beneficios,” Amazon Web Services, Inc. https://aws.amazon.com/es/what-is-cloud-computing/ (accessed Nov. 02, 2022).

[21] “¿Qué es AWS?,” Amazon Web Services, Inc. https://aws.amazon.com/es/what-is-aws/ (accessed Nov. 02, 2022).

[22] “Regiones y zonas de disponibilidad de la infraestructura global,” Amazon Web Services, Inc. https://aws.amazon.com/es/about-aws/global-infrastructure/regions_az/ (accessed Nov. 02, 2022).

[23] “AWS | Almacenamiento de datos seguro en la nube (S3),” Amazon Web Services, Inc. https://aws.amazon.com/es/s3/ (accessed Nov. 02, 2022).

[24] “AWS Glue | Servicio de integración de datos sin servidor | Amazon Web Services,” Amazon Web Services, Inc. https://aws.amazon.com/es/glue/ (accessed Nov. 02, 2022).

[25] “Definición de rastreadores en AWS Glue - AWS Glue.” https://docs.aws.amazon.com/es_es/glue/latest/dg/add-crawler.html (accessed Nov. 02, 2022).

[26] “Funcionamiento de los rastreadores - AWS Glue.” https://docs.aws.amazon.com/es_es/glue/latest/dg/crawler-running.html (accessed Nov. 02, 2022).

[27] “Lago de datos | Implementaciones | Soluciones de AWS,” Amazon Web Services, Inc. https://aws.amazon.com/es/solutions/implementations/data-lake-solution/ (accessed Nov. 02, 2022).

[28] “Apache Spark en Amazon EMR - Plataforma para big data - Amazon Web Services,” Amazon WebServices, Inc. https://aws.amazon.com/es/emr/features/spark/ (accessed Nov. 02, 2022).

[29] “Consultas de datos al instante | Análisis de datos SQL | Amazon Athena,” Amazon Web Services,
Inc. https://aws.amazon.com/es/athena/ (accessed Nov. 02, 2022).

[30] “Amazon QuickSight - Servicio de inteligencia empresarial - Amazon Web Services,” Amazon Web Services, Inc. https://aws.amazon.com/es/quicksight/ (accessed Nov. 02, 2022).

8. ANEXOS

En este capítulo se muestra los siguientes anexos: Código, pasos tomados en la consola de AWS para cada servicio usado y las imágenes de costos en consola de AWS

8.1. Código

Código y la síntesis de lo que hace línea por línea:

```
import numpy as np
import pandas as pd
```

"NumPy" es una librería de Python que proporciona funcionalidades avanzadas para el manejo de datos y el cálculo numérico. Ofrece una amplia gama de funciones matemáticas y de álgebra lineal, así como una estructura de datos de matriz eficiente para el almacenamiento y el procesamiento de datos numéricos en grandes volúmenes.

"Pandas" es una librería de Python que proporciona estructuras de datos y herramientas de análisis de datos de alto rendimiento. Permite leer y escribir datos de varios formatos (por ejemplo, CSV, Excel, SQL), así como realizar operaciones de manipulación de datos y análisis de forma sencilla y rápida.

```
pip install sparkmagic
pip install pyspark
```

"pip install sparkmagic", instala un paquete denominado "sparkmagic". Este paquete proporciona un conjunto de herramientas y librerías que permiten interactuar con un cluster de Apache Spark desde Jupyter Notebook y otras aplicaciones de código abierto.

"pip install pyspark", instala un paquete denominado "pyspark". Este paquete proporciona una interfaz de programación de aplicaciones (API) de Python para Apache Spark, un motor

de procesamiento de datos en paralelo de código abierto. Con "pyspark", es posible escribir aplicaciones Spark utilizando Python y aprovechar la escalabilidad y la eficiencia de Spark para el procesamiento de grandes volúmenes de datos.

```
from pyspark.sql.types import *
from pyspark.sql.functions import *
from pyspark.sql import SparkSession

from pyspark.ml import Pipeline
from  pyspark.ml.feature  import  VectorAssembler,  StringIndexer,
VectorIndexer, MinMaxScaler
from pyspark.ml.classification import LogisticRegression
from pyspark.ml.tuning import ParamGridBuilder, CrossValidator
from pyspark.ml.evaluation import BinaryClassificationEvaluator
```

Importa la clase StructType y StructField de la biblioteca pyspark.sql.types para crear un esquema personalizado para un DataFrame de Spark.

Importa varias funciones útiles de la biblioteca pyspark.sql.functions, como lit, col, when, count, etc.

Importa la clase SparkSession de la biblioteca pyspark.sql, que se utiliza para crear una sesión de Spark y trabajar con DataFrames.

Importa la clase Pipeline de la biblioteca pyspark.ml, que se utiliza para crear una secuencia de transformaciones y modelos de aprendizaje automático en Spark.

Importa la clase VectorAssembler de la biblioteca pyspark.ml.feature, que se utiliza para combinar varias columnas de un DataFrame en una sola columna de vectores.

Importa la clase StringIndexer de la biblioteca pyspark.ml.feature, que se utiliza para codificar valores de cadena como números únicos.

Importa la clase VectorIndexer de la biblioteca pyspark.ml.feature, que se utiliza para automáticamente identificar y codificar columnas de vectores categóricas como índices numéricos.

Importa la clase MinMaxScaler de la biblioteca pyspark.ml.feature, que se utiliza para normalizar los valores de una columna de vectores a un rango específico.

Importa la clase LogisticRegression de la biblioteca pyspark.ml.classification, que se utiliza para entrenar un modelo de regresión logística en Spark.

Importa la clase ParamGridBuilder de la biblioteca pyspark.ml.tuning, que se utiliza para construir una grilla de parámetros para la validación cruzada.

Importa la clase CrossValidator de la biblioteca pyspark.ml.tuning, que se utiliza para realizar validación cruzada en un modelo de aprendizaje automático en Spark.

Importa la clase BinaryClassificationEvaluator de la biblioteca pyspark.ml.evaluation, que se utiliza para evaluar un modelo de clasificación binaria en Spark.

```
spark = SparkSession.builder.master("local[*]").getOrCreate()
```

Crea una sesión de Spark y asigna el resultado a la variable spark. La sesión se crea con el master local y se utilizará para crear y trabajar con DataFrames y modelos de aprendizaje automático en Spark.

```
csv = spark.read.csv(r'C:\Users\Jorge_Caballero\Downloads\Spark\flights.csv', inferSchema=True, header=True)
```

Lee un archivo CSV en un DataFrame de Spark utilizando el método read.csv del objeto spark. El archivo se encuentra en la ruta especificada y se especifica que se debe inferir el esquema del archivo (esto significa que Spark intentará determinar el tipo de datos de cada columna automáticamente). También se especifica que el archivo tiene una fila de encabezado.

```
csv.show(15)
```

Muestra las primeras 15 filas del DataFrame utilizando el método show.

```
+----------+---------+-------+---------------+-------------+------
--+--------+
|DayofMonth|DayOfWeek|Carrier|OriginAirportID|DestAirportID|DepDel
ay|ArrDelay| +----------+---------+-------+---------------+--------
-----+--------+---------+ | 19| 5| DL| 11433| 13303| -3| 1| | 19| 5|
DL| 14869| 12478| 0| -8| | 19| 5| DL| 14057| 14869| -4| -15| | 19|
5| DL| 15016| 11433| 28| 24| | 19| 5| DL| 11193| 12892| -6| -11| |
19| 5| DL| 10397| 15016| -1| -19| | 19| 5| DL| 15016| 10397| 0| -1|
| 19| 5| DL| 10397| 14869| 15| 24| | 19| 5| DL| 10397| 10423| 33|
34| | 19| 5| DL| 11278| 10397| 323| 322| | 19| 5| DL| 14107| 13487|
-7| -13| | 19| 5| DL| 11433| 11298| 22| 41| | 19| 5| DL| 11298|
```

```
11433| 40| 20| | 19| 5| DL| 11433| 12892| -2| -7| | 19| 5| DL| 10397|
12451| 71| 75| +----------+---------+-------+---------------+------
-------+--------+---------+ only showing top 15 rows
```

En formato tabla

DayofMonth	DayOfWeek	Carrier	OriginAirportID	DestAirportID	DepDelay	ArrDelay
19	5	DL	11433	13303	-3	1
19	5	DL	14869	12478	0	-8
19	5	DL	14057	14869	-4	-15
19	5	DL	15016	11433	28	24
19	5	DL	11193	12892	-6	-11
19	5	DL	10397	15016	-1	-19
19	5	DL	15016	10397	0	-1
19	5	DL	10397	14869	15	24
19	5	DL	10397	10423	33	34
19	5	DL	11278	10397	323	322
19	5	DL	14107	13487	-7	-13
19	5	DL	11433	11298	22	41
19	5	DL	11298	11433	40	20
19	5	DL	11433	12892	-2	-7
19	5	DL	10397	12451	71	75

Tabla 1, data frame de la IATA con demoras de las aerolíneas

```
data = csv.select("DayofMonth", "DayOfWeek", "Carrier",
"OriginAirportID", "DestAirportID", "DepDelay", ((col("ArrDelay") >
15).cast("Int").alias("label")))
```

Selecciona ciertas columnas del DataFrame csv y las asigna a un nuevo DataFrame data. Las columnas seleccionadas son: "DayofMonth", "DayOfWeek", "Carrier", "OriginAirportID", "DestAirportID", "DepDelay". También se agrega una nueva columna llamada "label", que es una columna booleana que indica si el valor en la columna "ArrDelay" es mayor a 15. La columna booleana se convierte a un tipo de datos entero utilizando el método cast y se le da un alias utilizando el método alias.

```
data.show(10)
```

Muestra las primeras 10 filas del DataFrame utilizando el método show.

```
+---------+---------+-------+--------------+-------------+------
--+-----+
|DayofMonth|DayOfWeek|Carrier|OriginAirportID|DestAirportID|DepDel
ay|label| +---------+---------+-------+--------------+-----------
--+--------+-----+ | 19| 5| DL| 11433| 13303| -3| 0| | 19| 5| DL|
14869| 12478| 0| 0| | 19| 5| DL| 14057| 14869| -4| 0| | 19| 5| DL|
15016| 11433| 28| 1| | 19| 5| DL| 11193| 12892| -6| 0| | 19| 5| DL|
10397| 15016| -1| 0| | 19| 5| DL| 15016| 10397| 0| 0| | 19| 5| DL|
10397| 14869| 15| 1| | 19| 5| DL| 10397| 10423| 33| 1| | 19| 5| DL|
11278| 10397| 323| 1| +---------+---------+-------+--------------
+-------------+--------+-----+ only showing top 10 rows
```

En formato tabla

DayofMonth	DayOfWeek	Carrier	OriginAirportID	DestAirportID	DepDelay	label
19	5	DL	11433	13303	-3	0
19	5	DL	14869	12478	0	0
19	5	DL	14057	14869	-4	0
19	5	DL	15016	11433	28	1
19	5	DL	11193	12892	-6	0
19	5	DL	10397	15016	-1	0
19	5	DL	15016	10397	0	0
19	5	DL	10397	14869	15	1
19	5	DL	10397	10423	33	1
19	5	DL	11278	10397	323	1

Tabla 2, data frame con columna label binaria si el delay es mayor a 15

```
splits = data.randomSplit([0.7, 0.3])
```

Divide aleatoriamente el DataFrame data en dos DataFrames utilizando el método randomSplit. El primer DataFrame obtiene el 70% de las filas y el segundo DataFrame obtiene el 30% restante. Los dos DataFrames se almacenan en una lista llamada splits.

```
train = splits[0]
```

Asigna el primer DataFrame de la lista splits a la variable train. Este DataFrame se utilizará para entrenar el modelo de aprendizaje automático.

```
test = splits[1].withColumnRenamed("label", "trueLabel")
train_rows = train.count()
test_rows = test.count()
```

```
print("Training Rows:", train_rows, " Testing Rows:", test_rows)
```

Asigna el segundo DataFrame de la lista splits a la variable test. Este DataFrame se utilizará para evaluar el modelo entrenado. También se renombra la columna "label" del DataFrame a "trueLabel" utilizando el método withColumnRenamed.

Cuenta el número de filas en el DataFrame de entrenamiento y lo almacena en la variable train_rows.

Cuenta el número de filas en el DataFrame de prueba y lo almacena en la variable test_rows.

Imprime el número de filas en los DataFrames de entrenamiento y prueba utilizando la función print.

```
Training Rows: 1891409 Testing Rows: 810809
```

```
strIdx = StringIndexer(inputCol = "Carrier", outputCol = "CarrierIdx")
```

Crear un objeto "StringIndexer" que se utiliza para convertir variables categóricas con valores de cadena a variables numéricas. En este caso, la columna "Carrier" de los datos de entrada se transformará a una columna numérica "CarrierIdx" en los datos de salida.

```
catVect = VectorAssembler(inputCols = ["CarrierIdx", "DayofMonth", "DayOfWeek", "OriginAirportID", "DestAirportID"], outputCol="catFeatures")
```

Esta línea crea un objeto "VectorAssembler" que se utiliza para combinar varias columnas en una sola columna de vectores. En este caso, las columnas "CarrierIdx", "DayofMonth", "DayOfWeek", "OriginAirportID" y "DestAirportID" se combinarán en una única columna llamada "catFeatures".

```
catIdx = VectorIndexer(inputCol = catVect.getOutputCol(), outputCol = "idxCatFeatures")
```

Esta línea crea un objeto "VectorIndexer" que se utiliza para indexar los vectores en la columna "catFeatures". El resultado se guardará en la columna "idxCatFeatures".

```
numVect = VectorAssembler(inputCols = ["DepDelay"], outputCol="numFeatures")
```

Esta línea crea un objeto "VectorAssembler" que combinará la columna "DepDelay" en una única columna de vectores llamada "numFeatures".

```
minMax = MinMaxScaler(inputCol = numVect.getOutputCol(), outputCol="normFeatures")
featVect = VectorAssembler(inputCols=["idxCatFeatures", "normFeatures"], outputCol="features")
```

Luego, esta línea crea un objeto "MinMaxScaler" que se utiliza para normalizar los datos en la columna "numFeatures". El resultado se guardará en la columna "normFeatures".

La segunda línea crea un objeto "VectorAssembler" que combinará las columnas "idxCatFeatures" y "normFeatures" en una única columna de vectores llamada "features".

```
lr = LogisticRegression(labelCol="label",featuresCol="features",maxIter=10,regParam=0.3)
```

Crea un objeto "LogisticRegression" que es un modelo de regresión logística. El modelo se entrenará utilizando la columna "features" como variables predictoras y la columna "label" como variable objetivo. El parámetro "maxIter" especifica el número máximo de iteraciones que se utilizarán durante el proceso de entrenamiento, y el parámetro "regParam" especifica el parámetro de regularización que se utilizará para evitar el sobreajuste del modelo.

```
pipeline = Pipeline(stages=[strIdx, catVect, catIdx, numVect, minMax, featVect, lr])
```

Crea un objeto "Pipeline" que permite encadenar varios procesos de transformación y modelado en un flujo de trabajo. Los objetos que se han creado anteriormente (strIdx, catVect, catIdx, numVect, minMax, featVect y lr) se añaden como etapas en el pipeline, de modo que cuando se aplique el pipeline a un conjunto de datos, se ejecutarán todas las etapas en orden. Esto permite automatizar y simplificar el proceso de preprocesamiento y entrenamiento de un modelo de machine learning.

```
piplineModel = pipeline.fit(train)
```

La línea de código "piplineModel = pipeline.fit(train)" es utilizada para ajustar el pipeline a un conjunto de datos de entrenamiento. El pipeline incluye todas las etapas de preprocesamiento y modelado especificadas en la creación del objeto "Pipeline", y el método "fit" se utiliza para ajustar el pipeline a los datos de entrenamiento. El resultado es un modelo ajustado que puede ser utilizado para realizar predicciones en nuevos datos.

En concreto, el método "fit" realiza las siguientes acciones:

Aplica las etapas de preprocesamiento del pipeline a los datos de entrenamiento. Esto incluye la conversión de variables categóricas a numéricas, la combinación de varias columnas en vectores, la normalización de datos y la combinación de todas las variables en un único vector de características.

Entrena el modelo especificado en la última etapa del pipeline utilizando las características y la variable objetivo de los datos de entrenamiento.

Devuelve un modelo ajustado que puede utilizarse para realizar predicciones en nuevos datos.

Output

```
+-----------------------------------------------------+----------+--
-------+ |features |prediction|trueLabel| +-----------------------
---------------------------+----------+---------+
|[10.0,1.0,0.0,10397.0,12191.0,0.03115264797507788]   |0.0   |0   |
|[10.0,1.0,0.0,10397.0,12264.0,0.030114226375908618]|0.0    |0    |
|[10.0,1.0,0.0,10397.0,12264.0,0.03115264797507788]   |0.0   |0   |
|[10.0,1.0,0.0,10397.0,13851.0,0.03167185877466251]   |0.0   |0   |
|[10.0,1.0,0.0,10423.0,13487.0,0.027518172377985463]|0.0    |0    |
|[10.0,1.0,0.0,10423.0,13487.0,0.029595015576323987]|0.0    |0    |
|[10.0,1.0,0.0,10423.0,14869.0,0.027518172377985463]|0.0    |0    |
|[10.0,1.0,0.0,10529.0,11193.0,0.030114226375908618]|0.0    |0    |
|[10.0,1.0,0.0,10529.0,11433.0,0.030114226375908618]|0.0    |0    |
|[10.0,1.0,0.0,10693.0,11193.0,0.03115264797507788]   |0.0   |0   |
|[10.0,1.0,0.0,10721.0,12478.0,0.05192107995846314]   |0.0   |1   |
|[10.0,1.0,0.0,10721.0,13931.0,0.03271028037383177]   |0.0   |1   |
|[10.0,1.0,0.0,10792.0,11433.0,0.028037383177570093]|0.0    |0    |
|[10.0,1.0,0.0,10792.0,11433.0,0.03686396677050883]   |0.0   |1   |
|[10.0,1.0,0.0,10821.0,11193.0,0.03167185877466251]   |0.0   |0   |
|[10.0,1.0,0.0,10821.0,11193.0,0.05659397715472482]   |0.0   |1   |
|[10.0,1.0,0.0,10821.0,12478.0,0.027518172377985463]|0.0    |0    |
|[10.0,1.0,0.0,10821.0,13487.0,0.029075804776739357]|0.0    |0    |
|[10.0,1.0,0.0,10821.0,14492.0,0.03167185877466251]   |0.0   |0   |
|[10.0,1.0,0.0,11042.0,11433.0,0.028556593977154723]|0.0    |0    |
|[10.0,1.0,0.0,11042.0,11433.0,0.029595015576323987]|0.0    |0    |
|[10.0,1.0,0.0,11042.0,12478.0,0.030633437175493248]|0.0 |0 |

...
```

```
|[10.0,1.0,0.0,13244.0,11278.0,0.05555555555555555] |0.0 |1 | +----
-----------------------------------------------+----------+-------
--+ only showing top 100 rows
```

En formato tabla

features	prediction	trueLabel
[10.0,1.0,0.0,10397.0,12191.0,0.03115264797507788]	0	0
[10.0,1.0,0.0,10397.0,12264.0,0.030114226375908618]	0	0
[10.0,1.0,0.0,10397.0,12264.0,0.03115264797507788]	0	0
[10.0,1.0,0.0,10397.0,13851.0,0.03167185877466251]	0	0
[10.0,1.0,0.0,10423.0,13487.0,0.027518172377985463]	0	0
[10.0,1.0,0.0,10423.0,13487.0,0.029595015576323987]	0	0
[10.0,1.0,0.0,10423.0,14869.0,0.027518172377985463]	0	0
[10.0,1.0,0.0,10529.0,11193.0,0.030114226375908618]	0	0
[10.0,1.0,0.0,10529.0,11433.0,0.030114226375908618]	0	0
[10.0,1.0,0.0,10693.0,11193.0,0.03115264797507788]	0	0
[10.0,1.0,0.0,10721.0,12478.0,0.05192107995846314]	0	1
[10.0,1.0,0.0,10721.0,13931.0,0.03271028037383177]	0	1
[10.0,1.0,0.0,10792.0,11433.0,0.028037383177570093]	0	0
[10.0,1.0,0.0,10792.0,11433.0,0.03686396677050883]	0	1
[10.0,1.0,0.0,10821.0,11193.0,0.03167185877466251]	0	0
[10.0,1.0,0.0,10821.0,11193.0,0.05659397715472482]	0	1
[10.0,1.0,0.0,10821.0,12478.0,0.027518172377985463]	0	0
[10.0,1.0,0.0,10821.0,13487.0,0.029075804776739357]	0	0
[10.0,1.0,0.0,10821.0,14492.0,0.03167185877466251]	0	0
[10.0,1.0,0.0,11042.0,11433.0,0.028556593977154723]	0	0
[10.0,1.0,0.0,11042.0,11433.0,0.029595015576323987]	0	0

Tabla 3, con columnas de features que son los vectores, la columna de prediction y de true label

```
evaluator = BinaryClassificationEvaluator(labelCol="trueLabel",
rawPredictionCol="rawPrediction", metricName="areaUnderROC")
aur = evaluator.evaluate(prediction)
print ("AUR = ", aur)
```

Las siguientes 3 líneas lo que hacen es
La primera crea un objeto "BinaryClassificationEvaluator" que se utiliza para evaluar el rendimiento de un modelo de clasificación binaria. El parámetro "labelCol" especifica la columna en los datos que contiene la variable objetivo, mientras que el parámetro "rawPredictionCol" especifica la columna que contiene las predicciones del modelo. El parámetro "metricName" especifica la métrica de evaluación que se utilizará para medir el

rendimiento del modelo. En este caso, se está utilizando la "areaUnderROC", que es el área bajo la curva ROC (Receiver Operating Characteristic).

La segunda: Utiliza el objeto "evaluator" y el método "evaluate" para calcular el valor de la métrica de evaluación especificada en el objeto "evaluator" para el conjunto de predicciones proporcionado. En este caso, el valor devuelto será la "areaUnderROC" del modelo.

La tercera: Imprime el valor de la "areaUnderROC" del modelo. El área bajo la curva ROC es una métrica comúnmente utilizada para evaluar el rendimiento de un modelo de clasificación binaria, y se interpreta como la probabilidad de que un modelo clasifique correctamente a una muestra positiva como positiva y a una muestra negativa como negativa. Un valor más cercano a 1 indica un rendimiento mejor del modelo.

Output

```
AUR = 0.922869888282246
```

```
prediction.select("rawPrediction",   "probability",   "prediction",
"trueLabel").show(100, truncate=False)
```

Con esta línea imprimimos las columnas "rawPrediction", "probability", "prediction" y "trueLabel" del conjunto de datos "prediction" y las muestra en pantalla.

En resumen, esta línea de código se utiliza para visualizar las predicciones del modelo y la variable objetivo-verdadera para el conjunto de datos. Esto puede ser útil para inspeccionar el rendimiento del modelo y detectar posibles errores o patrones en las predicciones incorrectas.

```
+----------------------------------------+-----------------------
---------------+----------+---------+ |rawPrediction |probability
|prediction|trueLabel| +----------------------------------------+--
-------------------------------------+----------+---------+
|[1.5974222333219208,-
1.5974222333219208]|[0.8316577982739407,0.1683422017260593] |0.0 |0
|                                                |[1.6254215612648948,-
1.6254215612648948]|[0.8355414728404721,0.1644585271595279] |0.0 |0
|                                                |[1.5974632706356662,-
1.5974632706356662]|[0.8316635435470835,0.1683364564529165] |0.0 |0
|                                                |[1.5843762652787787,-
1.5843762652787787]|[0.8298234112390493,0.17017658876095065]|0.0 |0
|                                                |[1.6962332850497956,-
```

```
1.6962332850497956]|[0.8450421397513976,0.15495786024860236]|0.0 |0
|                                                  |[1.6403167037913389,-
1.6403167037913389]|[0.837578026750222,0.16242197324977803] |0.0 |0
|                                                  |[1.6970101832360451,-
1.6970101832360451]|[0.84514384413228,0.15485615586771995] |0.0 |0
|                                                  |[1.6259794771304519,-
1.6259794771304519]|[0.8356181227796022,0.16438187722039777]|0.0 |0
|                                                  |[1.6261143943263274,-
1.6261143943263274]|[0.8356366542306584,0.16436334576934164]|0.0 |0
|                                                  |[1.5994623785599573,-
1.5994623785599573]|[0.8319432317097484,0.16805676829025162]|0.0 |0
|                                                  |[1.0412649923090669,-
1.0412649923090669]|[0.7390940134458817,0.26090598655411834]|0.0 |1
|                                                  |[1.5593101801398204,-
1.5593101801398204]|[0.8262543460374632,0.17374565396253683]|0.0 |1
|                                                  |[1.6843421555326317,-
1.6843421555326317]|[0.8434786483459059,0.1565213516540941] |0.0 |0
|                                                  |[1.4466966851841907,-
1.4466966851841907]|[0.809489529960885,0.19051047003911503] |0.0 |1
|                                                  |[1.5866080660716722,-
1.5866080660716722]|[0.8301383464199925,0.1698616535800075] |0.0 |0
|                                                  |[0.9156090909701924,-
0.9156090909701924]|[0.7141465835802847,0.2858534164197153] |0.0 |1
|                            |[1.699163597741502,-1.699163597741502]
|[0.845425464401262,0.15457453559873802]      |0.0      |0      |
|[1.6577933761753192,-
1.6577933761753192]|[0.8399415676271719,0.16005843237282813]|0.0 |0
|                            |[1.588462615359977,-1.588462615359977]
|[0.8303996938657816,0.1696003061342184]      |0.0      |0      |
|[1.6725599493319407,-
1.6725599493319407]|[0.8419168304826739,0.15808316951732615]|0.0 |0
|                                                  |[1.6446016587027126,-
1.6446016587027126]|[0.838160113743033,0.161839886256967] |0.0 |0 |
|[1.6172308200305254,-
1.6172308200305254]|[0.8344128718992118,0.16558712810078824]|0.0 |0
|
...
|[0.9649078986651063,-
0.9649078986651063]|[0.7241033713596867,0.2758966286403133] |0.0 |1
| +----------------------------------------+----------------------
-----------------+----------+---------+ only showing top 100 rows
```

En formato tabla

rawPrediction	probability	prediction	trueLabel

[1.5974222333219208,-1.5974222333219208]	[0.8316577982739407,0.1683422017260593]	0	0
[1.6254215612648948,-1.6254215612648948]	[0.8355414728404721,0.1644585271595279]	0	0
[1.5974632706356662,-1.5974632706356662]	[0.8316635435470835,0.1683364564529165]	0	0
[1.5843762652787787,-1.5843762652787787]	[0.8298234112390493,0.17017658876095065]	0	0
[1.6962332850497956,-1.6962332850497956]	[0.8450421397513976,0.15495786024860236]	0	0
[1.6403167037913389,-1.6403167037913389]	[0.837578026750222,0.16242197324977803]	0	0
[1.6970101832360451,-1.6970101832360451]	[0.84514384413228,0.15485615586771995]	0	0
[1.6259794771304519,-1.6259794771304519]	[0.8356181227796022,0.16438187722039777]	0	0
[1.6261143943263274,-1.6261143943263274]	[0.8356366542306584,0.16436334576934164]	0	0
[1.5994623785599573,-1.5994623785599573]	[0.8319432317097484,0.16805676829025162]	0	0
[1.0412649923090669,-1.0412649923090669]	[0.7390940134458817,0.26090598655411834]	0	1
[1.5593101801398204,-1.5593101801398204]	[0.8262543460374632,0.17374565396253683]	0	1
[1.6843421555326317,-1.6843421555326317]	[0.8434786483459059,0.1565213516540941]	0	0
[1.4466966851841907,-1.4466966851841907]	[0.809489529960885,0.19051047003911503]	0	1
[1.699163597741502,-1.699163597741502]	[0.845425464401262,0.15457453559873802]	0	0
[1.6577933761753192,-1.6577933761753192]	[0.8399415676271719,0.16005843237282813]	0	0
[1.588462615359977,-1.588462615359977]	[0.8303996938657816,0.1696003061342184]	0	0
[1.6725599493319407,-1.6725599493319407]	[0.8419168304826739,0.15808316951732615]	0	0
[1.6446016587027126,-1.6446016587027126]	[0.838160113743033,0.161839886256967]	0	0
[1.6172308200305254,-1.6172308200305254]	[0.8344128718992118,0.16558712810078824]	0	0
[0.9649078986651063,-0.9649078986651063]	[0.7241033713596867,0.2758966286403133]	0	1

Tabla 4, con los raw prediction y su probabilidad

La primera línea crea una grilla de parámetros, que especifica diferentes valores para los parámetros del modelo que se van a probar durante la validación cruzada. En este caso, se están probando diferentes valores para tres parámetros: "regParam", "maxIter" y "threshold".

La segunda línea crea un objeto de validación cruzada, que se encargará de ajustar el modelo con diferentes combinaciones de los parámetros especificados en la grilla de parámetros y evaluar el rendimiento del modelo en cada caso.

La tercera línea utiliza el objeto de validación cruzada para ajustar el modelo utilizando los datos de entrenamiento. El modelo resultante será el que tenga el mejor rendimiento en la validación cruzada.

```
newPrediction = model.transform(test)
newPredicted    =    prediction.select("features",    "prediction",
"trueLabel")
newPredicted.show()
```

La primera línea tiliza el modelo de aprendizaje automático entrenado previamente para hacer predicciones sobre el conjunto de datos de prueba "test". El resultado de esta predicción se almacena en la variable "newPrediction".

La segunda línea selecciona solo algunas columnas del conjunto de datos de predicción (las columnas "features", "prediction" y "trueLabel") y almacena el resultado en una nueva variable "newPredicted".

La tercera línea muestra el contenido del conjunto de datos "newPredicted", probablemente en forma de tabla. Esta línea de código es útil para inspeccionar los resultados de la predicción y compararlos con el valor verdadero (almacenado en la columna "trueLabel").

Output

```
+--------------------+----------+---------+                        |
features|prediction|trueLabel| +--------------------+----------+---
------+ |[10.0,1.0,0.0,103...| 0.0| 0| |[10.0,1.0,0.0,103...| 0.0|
0|  |[10.0,1.0,0.0,103...|  0.0|  0|  |[10.0,1.0,0.0,103...|  0.0|  0|
|[10.0,1.0,0.0,104...|  0.0|  0|  |[10.0,1.0,0.0,104...|  0.0|  0|
|[10.0,1.0,0.0,104...|  0.0|  0|  |[10.0,1.0,0.0,105...|  0.0|  0|
|[10.0,1.0,0.0,105...|  0.0|  0|  |[10.0,1.0,0.0,106...|  0.0|  0|
|[10.0,1.0,0.0,107...|  0.0|  1|  |[10.0,1.0,0.0,107...|  0.0|  1|
|[10.0,1.0,0.0,107...|  0.0|  0|  |[10.0,1.0,0.0,107...|  0.0|  1|
|[10.0,1.0,0.0,108...|  0.0|  0|  |[10.0,1.0,0.0,108...|  0.0|  1|
|[10.0,1.0,0.0,108...|  0.0|  0|  |[10.0,1.0,0.0,108...|  0.0|  0|
|[10.0,1.0,0.0,108...| 0.0| 0| |[10.0,1.0,0.0,110...| 0.0| 0| +----
---------------+----------+---------+ only showing top 20 rows
```

En Formato tabla

features	prediction	trueLabel
[10.0,1.0,0.0,107...]	0	0
[10.0,1.0,0.0,107...]	0	1
[10.0,1.0,0.0,108...]	0	0
[10.0,1.0,0.0,108...]	0	1
[10.0,1.0,0.0,108...]	0	0
[10.0,1.0,0.0,108...]	0	0
[10.0,1.0,0.0,108...]	0	0
[10.0,1.0,0.0,110...]	0	0

Tabla 5, mostrando las nuevas predicciones

```
evaluator2 = BinaryClassificationEvaluator(labelCol="trueLabel",
rawPredictionCol="prediction", metricName="areaUnderROC")
aur2 = evaluator.evaluate(prediction)
print( "AUR2 = ", aur2)
```

La primera línea de código, "evaluator2 = BinaryClassificationEvaluator(labelCol="trueLabel", rawPredictionCol="prediction", metricName="areaUnderROC")", crea un objeto de evaluación de clasificación binaria denominado "evaluator2". Este objeto utiliza la columna "trueLabel" del conjunto de datos como la columna de etiquetas verdaderas y la columna "prediction" como la columna de predicciones del modelo. Además, utiliza el área bajo la curva ROC (Receiver Operating Characteristic) como la métrica de evaluación.

La segunda línea de código, "aur2 = evaluator.evaluate(prediction)", aplica el evaluador "evaluator2" al conjunto de datos de predicción y almacena el resultado en la variable "aur2". El resultado será un valor numérico que representa el rendimiento del modelo en términos de la métrica especificada (en este caso, el área bajo la curva ROC).

La tercera línea de código, "print("AUR2 = ", aur2)", imprime el valor de la variable "aur2" junto con un mensaje descriptivo.

Output and final result

```
AUR2 = 0.9228699408935557
```

El valor "AUR2 = 0.9228699408935557" parece ser el resultado de una evaluación de rendimiento de un modelo de aprendizaje automático utilizando el área bajo la curva ROC (Receiver Operating Characteristic) como la métrica de evaluación.

El área bajo la curva ROC es una medida comúnmente utilizada para evaluar el rendimiento de un modelo de clasificación binaria. Se calcula a partir de la tasa de verdaderos positivos (sensibilidad) y la tasa de falsos positivos (especificidad) del modelo en diferentes umbrales de decisión. Un valor de área bajo la curva ROC cercano a 1 indica un buen rendimiento del modelo, mientras que un valor cercano a 0 indica un rendimiento pobre.

En este caso, el valor "AUR2 = 0.9228699408935557" indica que el modelo de aprendizaje automático tiene un buen rendimiento en términos de clasificación binaria, con un área bajo la curva ROC cercana a 1. Sin embargo, es importante tener en cuenta que el rendimiento del modelo puede variar dependiendo del conjunto de datos utilizado y de otros factores. Por lo tanto, es importante evaluar el rendimiento del modelo de aprendizaje automático en varios conjuntos de datos y en diferentes contextos para obtener una evaluación precisa del rendimiento del modelo.

8.2. Pasos tomados en la consola de AWS para cada servicio usado

- Se hace la selección de la Regio y la zona de disponibilidad

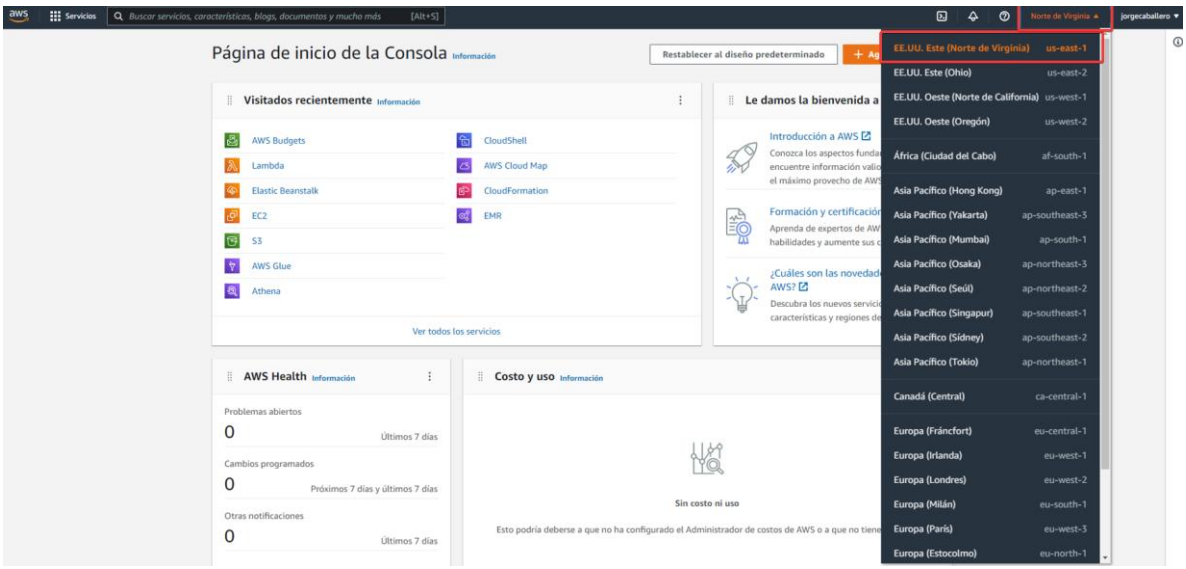

Ilustración 23, selección de zonas de disponibilidad de AWS

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

- Creación de los Buckets

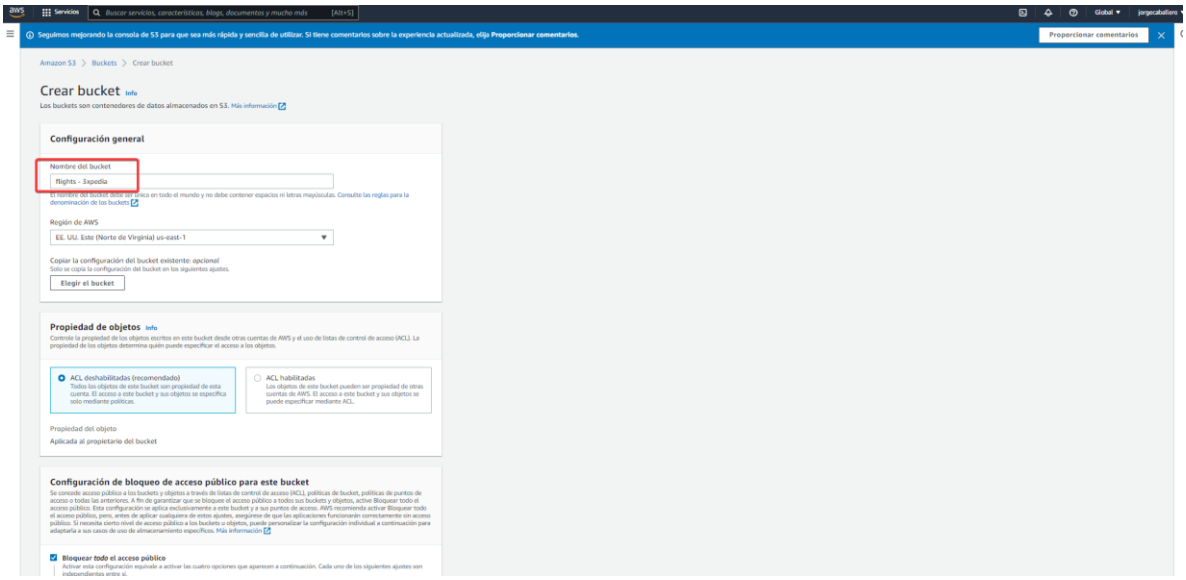

Ilustración 24, creación de S3 Bucket

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

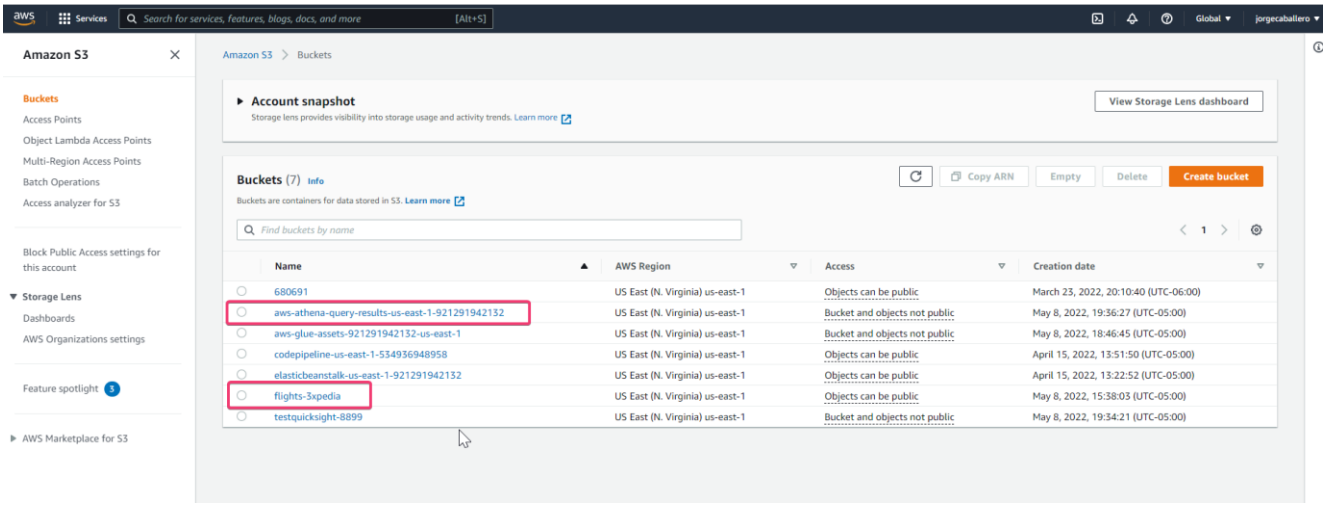

Ilustración 25, Lista de S3 Bucket

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

- Creación del rastreador (crawler)

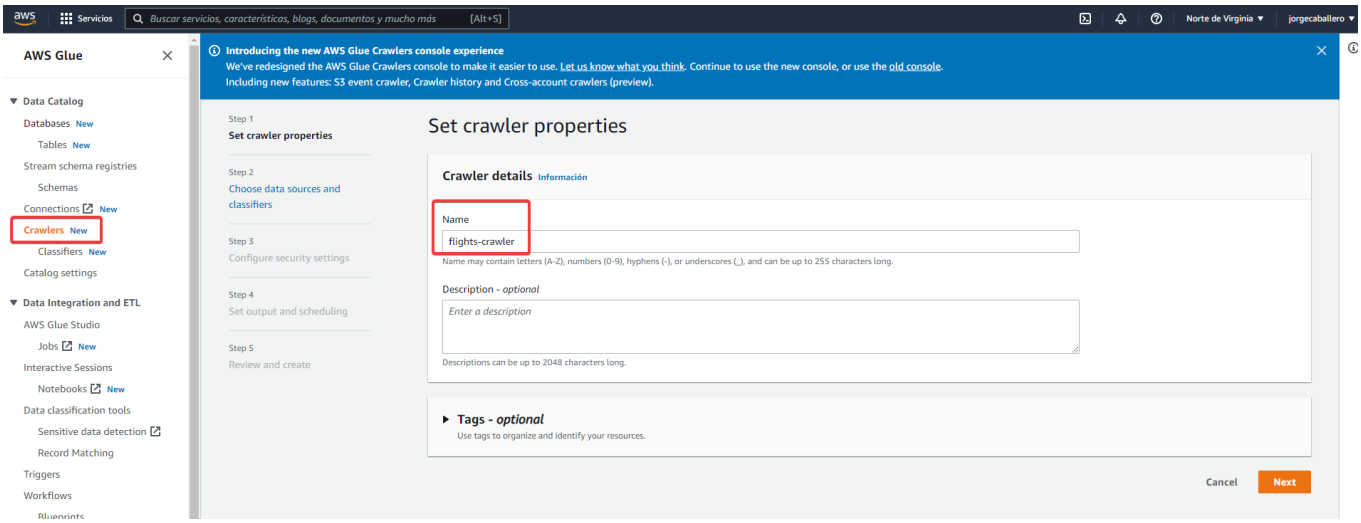

Ilustración 26, Creación de los rastreadores

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

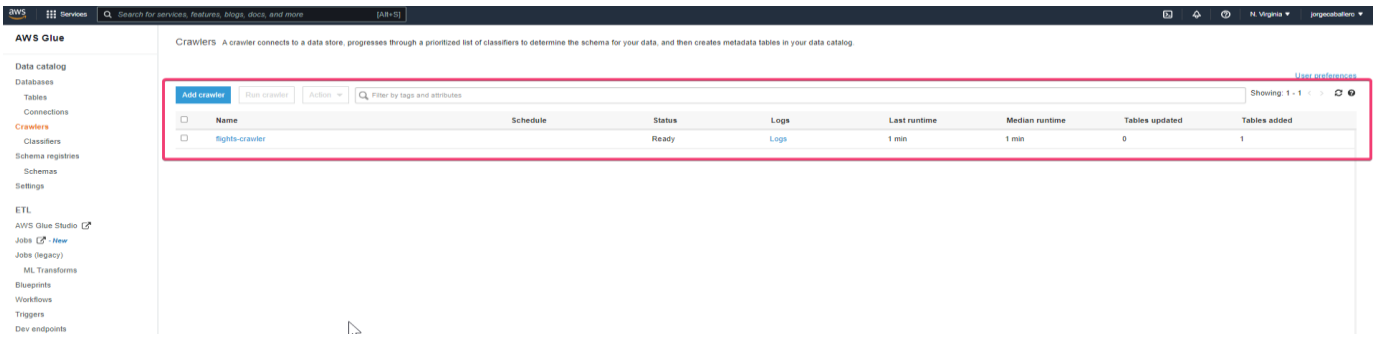

Ilustración 27, Lista de rastreadores

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

- Creación de los jobs necesarios

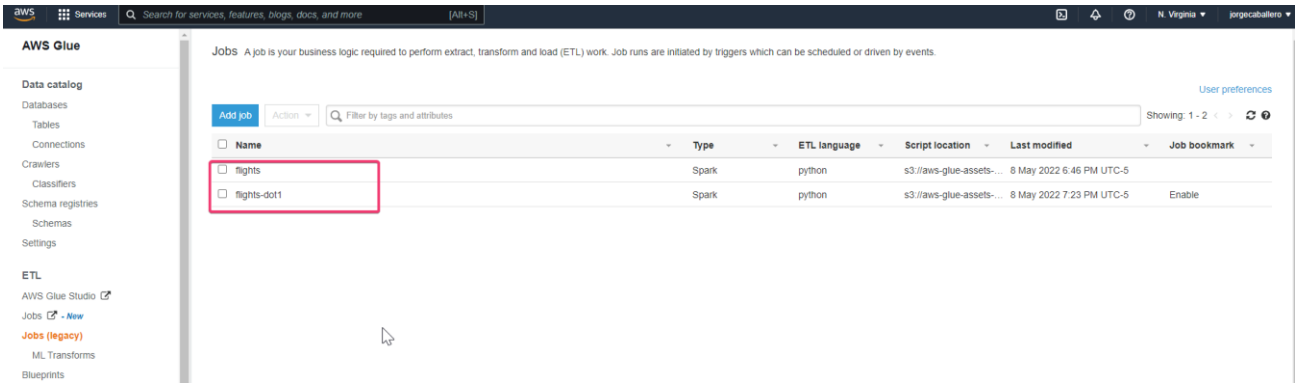

Ilustración 28, AWS Glue Jobs

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

- Creación de las tablas

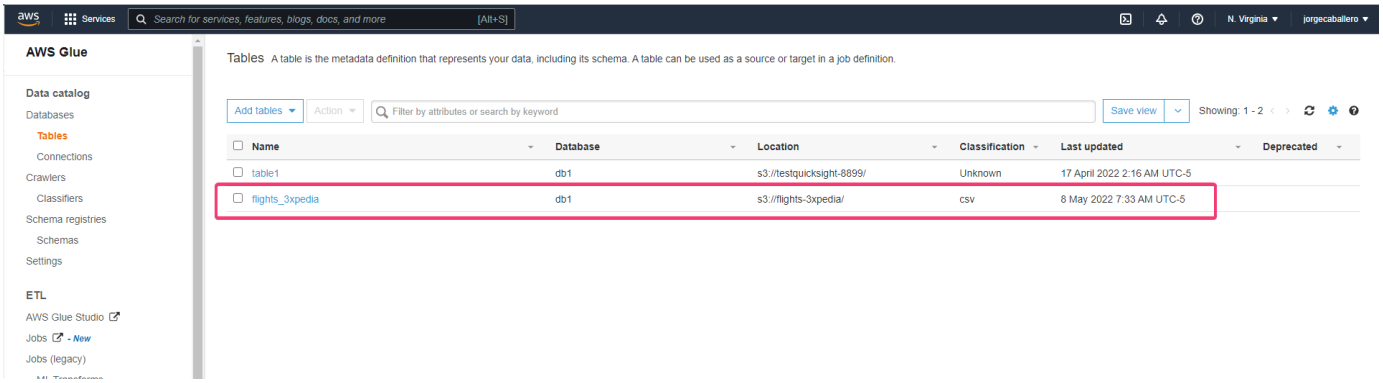

Ilustración 29, Creación de las tablas

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

- Lista de los Jobs

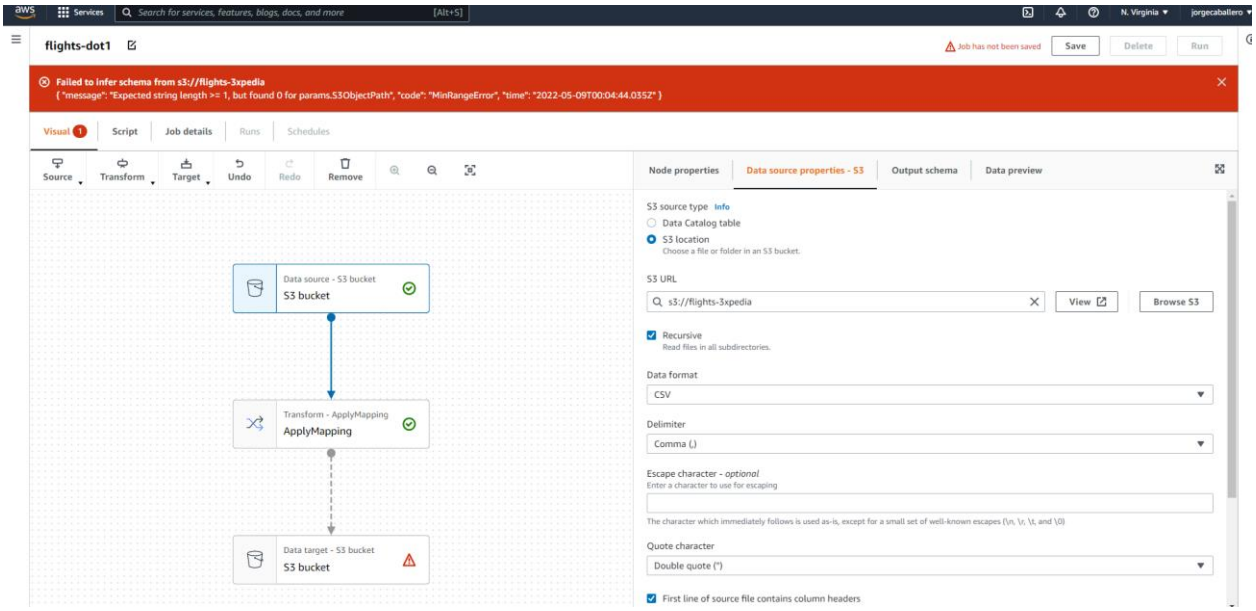

Ilustración 30, Creación de los Jobs en AWS

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

- Tranformar y mapear los datos

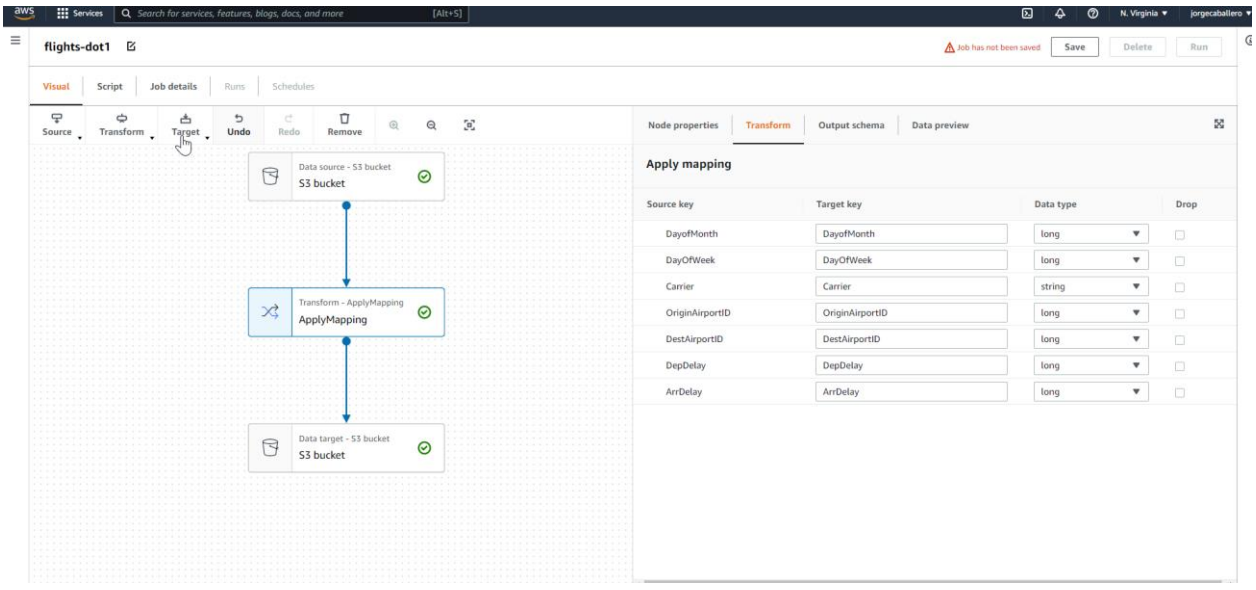

Ilustración 31, Transformación y mapeo de los datos

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

- Se crean los queries para la visualización y vistas

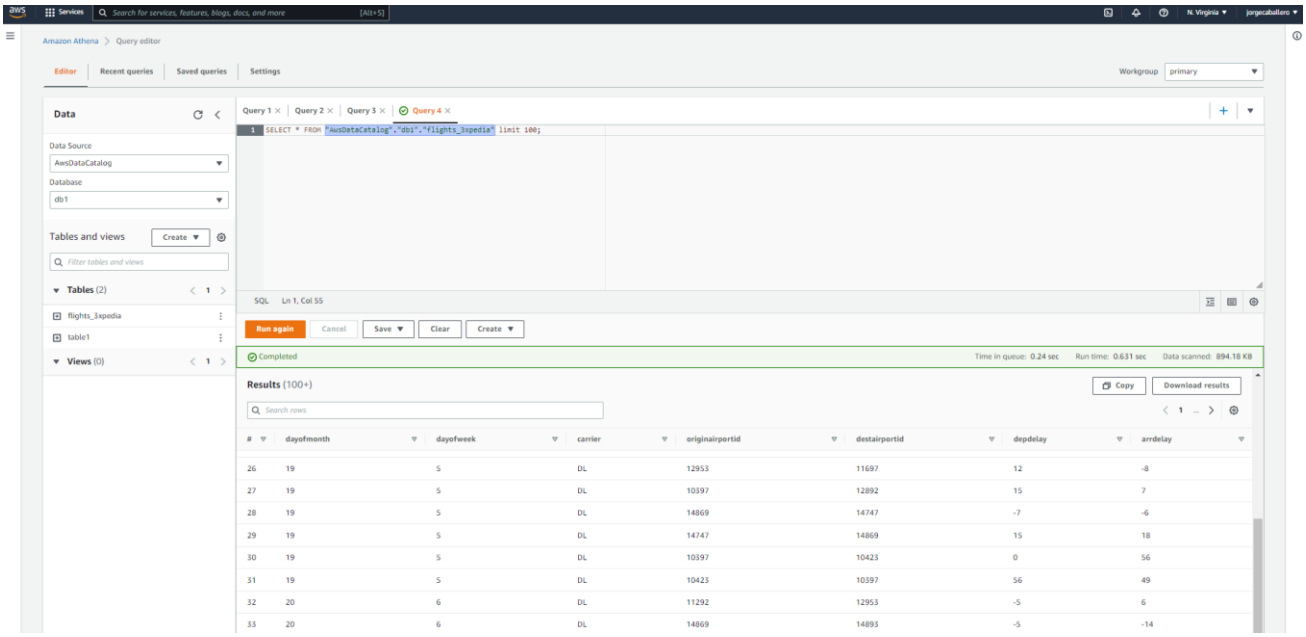

Ilustración 32, Creación de los queries y vistas

Imagen tomada de mi propia consola de AWS

"Interfaz de usuario | Consola de administración | AWS," Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

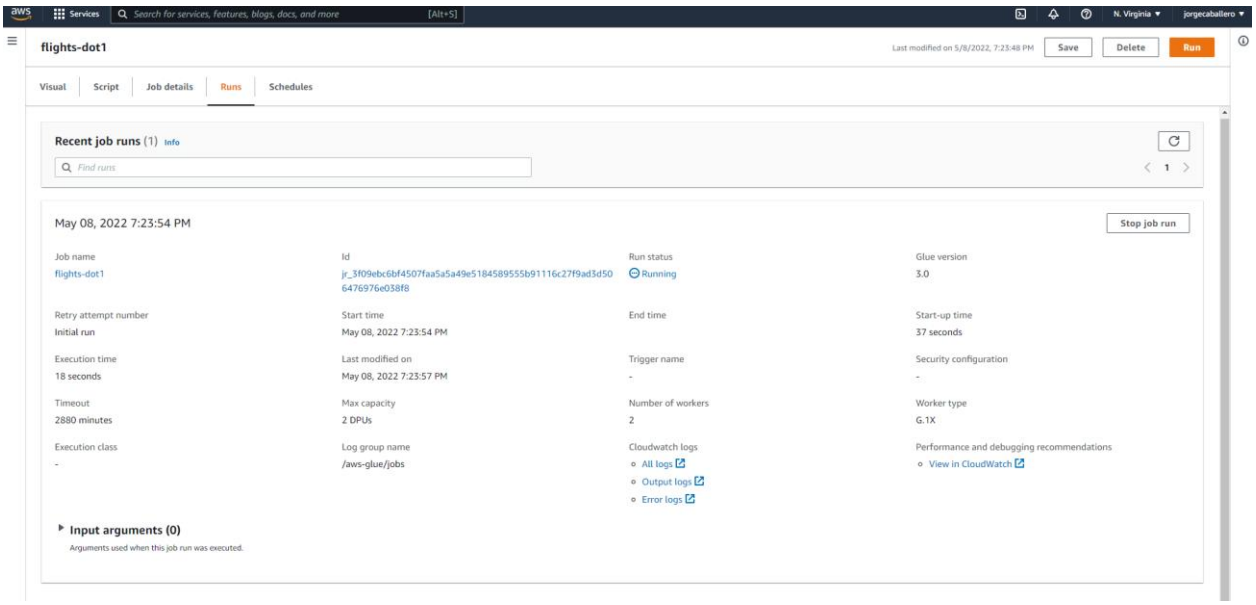

Ilustración 33, Corriendo los Jobs

Imagen tomada de mi propia consola de AWS

“Interfaz de usuario | Consola de administración | AWS,” Amazon Web Services, Inc. https://aws.amazon.com/es/console/ (accessed Dec. 24, 2022).

8.3. Imágenes de costos en consola de AWS

Se espera generar costos por los siguientes conceptos

- Creación de los Buckets

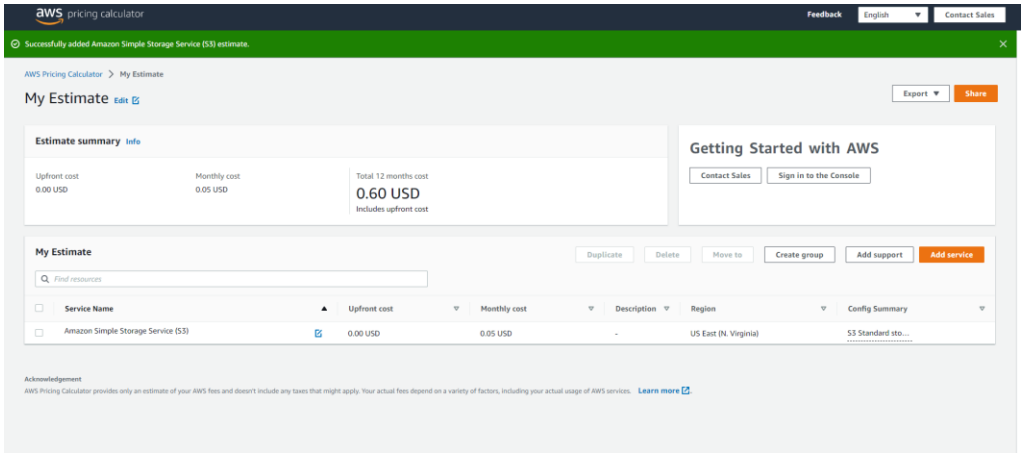

Ilustración 34, Costo por bucket por mes

“AWS Pricing Calculator.” https://calculator.aws/#/addService (accessed Dec. 24, 2022).

  - .60 USD por cada S3 bucket por mes

- S3 Bucket $0.023 por GB, 70MB por cada file cada mes

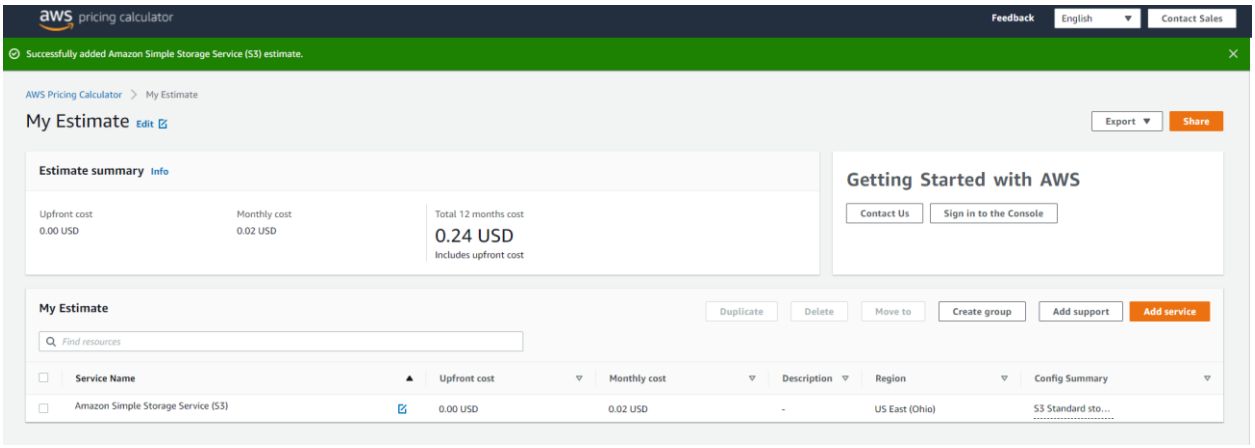

Ilustración 35, Costo de S3 por GB

"AWS Pricing Calculator." https://calculator.aws/#/addService (accessed Dec. 24, 2022).

- AWS Glue < 1 $

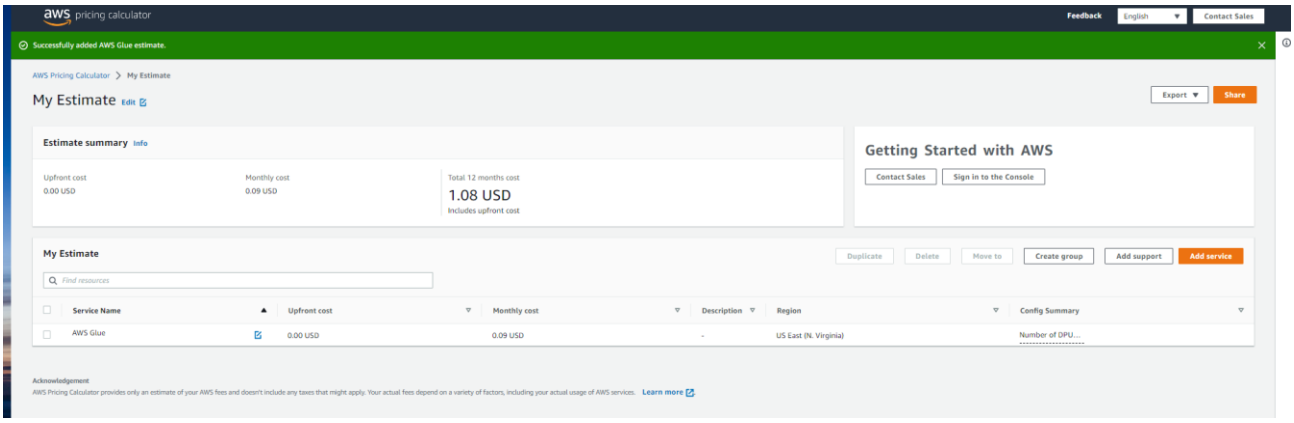

Ilustración 36, Costo AWS Glue

"AWS Pricing Calculator." https://calculator.aws/#/addService (accessed Dec. 24, 2022).

- Athena: < 5$

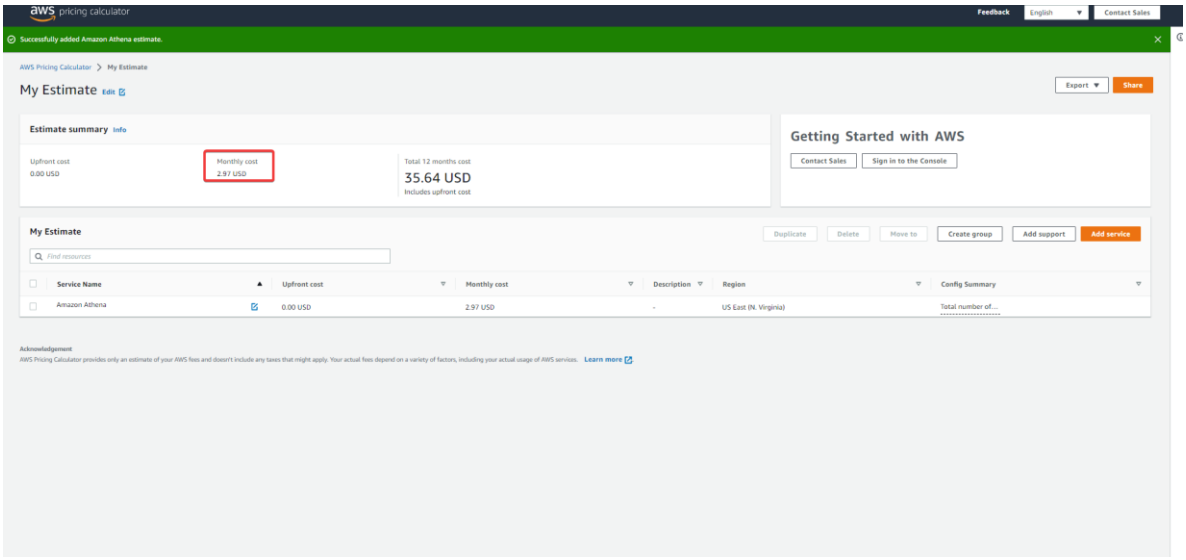

Ilustración 37, Costo AWS Athena

"AWS Pricing Calculator." https://calculator.aws/#/addService (accessed Dec. 24, 2022).

- Glue Developer Endpoint < 1 $
- Quicksight Edición Standard • 26 USD • por usuario por mes • con suscripción anual

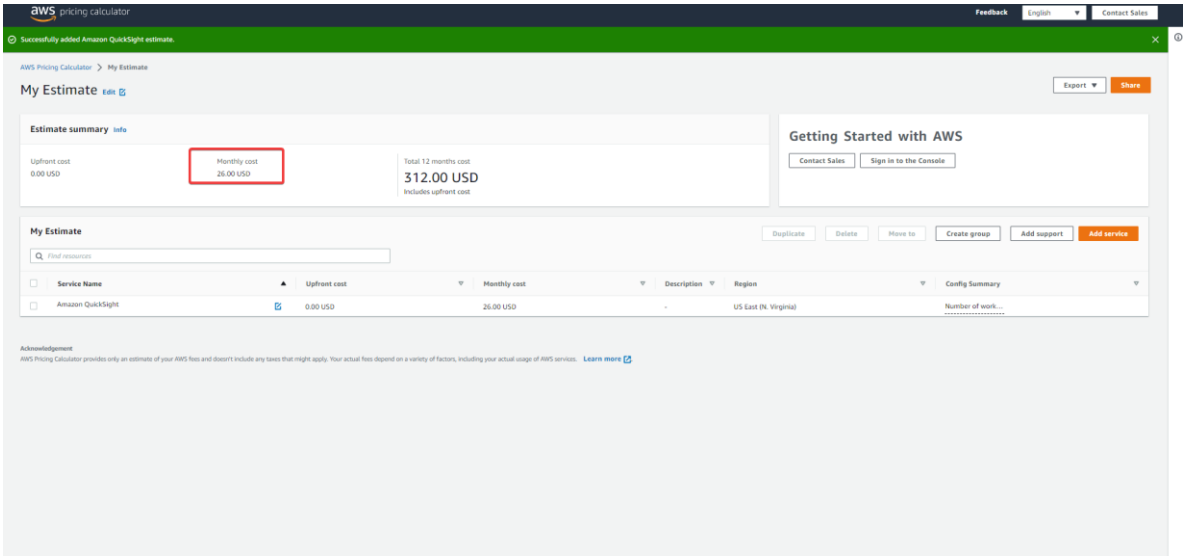

Ilustración 38, Costo AWS Quicksight

"AWS Pricing Calculator." https://calculator.aws/#/addService (accessed Dec. 24, 2022).