

Proyecto de Integración y Automatización de Datos para IA para el proyecto: "Herramienta de monitoreo del consumo de recursos en AWS de la empresa Grupo ASD"

Daniela Rodriguez Fonseca¹, Diego Armando Baquero Gutiérrez²

¹⁻²Facultad de Ingeniería y Ciencias Básicas

Universidad Central

Maestría en Analítica de Datos

Integración y Automatización de Datos para IA

Bogotá, D.C., Colombia

¹drodriguez7@ucentral.edu.co, ²dbaquerog@ucentral.edu.co

November 25, 2023

Contents

1	Introducción	3
2	Características del proyecto de investigación que hace uso de Integración y Automatización de Datos para IA	3
2.1	Titulo del proyecto de investigación	3
2.2	Objetivo general	3
2.2.1	Objetivos específicos	3
2.3	Alcance	4
2.4	Pregunta de investigación	4
2.5	Hipótesis	4
3	Reflexiones sobre el origen de datos e información	5
3.1	¿Cuál es el origen de los datos e información?	5
3.2	¿Cuáles son las consideraciones legales o éticas del uso de la información?	5
3.3	¿Cuáles son los retos de la información y los datos que utilizará en Integración y Automatización de Datos para IA?	5
3.4	¿Qué espera de la utilización de Integración y Automatización de Datos para IA para su proyecto?	5

4	Diseño de integración y Automatización de Datos para IA (diagrama)	6
5	Integración de datos	7
5.1	Extracción de logs	7
5.2	Transformación de Datos	8
6	Automatización de datos	9
6.1	Automatización de datos con AWS Lambda	9
6.2	Automatización con docker	9
7	IA	9
7.1	Visualización	10
8	Próximos pasos	12
9	Lecciones aprendidas	14
10	Bibliografía	15

1 Introducción

La computación en nube se ha convertido en una solución popular para muchas empresas debido a sus beneficios de escalabilidad, flexibilidad y reducción de costos (Pragma USA Inc., s/f). Con la escalabilidad en la nube, las empresas pueden aumentar o disminuir el uso de sus recursos según sea necesario, lo que les permite ser más ágiles y eficientes en la gestión de sus operaciones. La flexibilidad es otra ventaja, ya que permite adaptarse a las fluctuaciones en la demanda de recursos informáticos de manera rápida y eficiente. Para el caso de Grupo ASD, tanto la escalabilidad como la flexibilidad son especialmente importantes ya que durante su operación se necesita ajustar los recursos de la mejor manera en función de la demanda de los proyectos.

Es importante destacar que, aunque el uso de Cloud Computing es muy beneficioso para las empresas, también puede ser complicado de controlar y monitorear, lo que a su vez puede generar costos innecesarios. AWS ofrece a las empresas la capacidad de acceder a una amplia gama de servicios y aplicaciones sin tener que preocuparse por el mantenimiento y la administración de la infraestructura (Amazon Web Services Inc., 2023).

Por lo que, desarrollar una herramienta que permita el monitoreo oportuno del consumo de recursos en la plataforma Amazon Web Services (AWS) proporcionará información detallada sobre el uso de los recursos en la nube, lo que permitirá a Grupo ASD tomar decisiones informadas sobre la asignación de recursos y ajustar el uso de estos.

2 Características del proyecto de investigación que hace uso de Integración y Automatización de Datos para IA

2.1 Título del proyecto de investigación

Desarrollo de una herramienta de análisis y monitoreo del consumo de recursos en AWS de la empresa Grupo ASD.

2.2 Objetivo general

Desarrollar una herramienta para el análisis y monitoreo de recursos de los diferentes proyectos implementados en la infraestructura en nube de Amazon Web Services (AWS) en la empresa Grupo ASD.

2.2.1 Objetivos específicos

- Realizar un proceso de recopilación y análisis de los datos de consumo de recursos de los diferentes proyectos de la empresa Grupo ASD a través de la infraestructura en nube AWS.

- Diseñar el modelo de datos con el fin de determinar la clasificación y categorías de los datos recopilados por medio de un documento de arquitectura de datos.
- Diseñar un modelo de análisis de datos para la identificación del uso y aprovisionamiento de los diferentes recursos utilizados, mediante una herramienta de visualización de datos.

2.3 Alcance

Se espera obtener una herramienta de visualización y un modelo de Machine Learning que permita a la empresa Grupo ASD monitorear de forma centralizada el consumo de recursos desplegados en AWS de cada una de sus cuentas; lo cual permitirá una mejor toma de decisiones y optimización de costos en la gestión de infraestructuras en la nube. Además, se espera que el trabajo realizado pueda servir como base para futuros proyectos similares.

2.4 Pregunta de investigación

¿Qué plataforma se puede implementar para el desarrollo de una herramienta efectiva de monitoreo centralizado del consumo de recursos en AWS que permita a la organización optimizar sus costos y recursos de manera proactiva?

2.5 Hipótesis

Se cree que mediante la implementación de un sistema de monitoreo que integre y automatice la recopilación de datos de consumo de recursos en AWS, se podrá identificar patrones de uso, predecir picos de demanda y generar alertas, lo que resultará en una reducción de costos y una optimización de recursos para la organización.

3 Reflexiones sobre el origen de datos e información

3.1 ¿Cuál es el origen de los datos e información?

Los datos serán obtenidos de la cuenta corporativa de Amazon Web Services (AWS) de la empresa Grupo ASD del servicio de Cloudwatch configurado para cada proyecto que despliega infraestructura en esta plataforma.

3.2 ¿Cuáles son las consideraciones legales o éticas del uso de la información?

Como requerimiento de la empresa Grupo ASD los datos de los proyectos como nombres, descripción o propósito no podrán ser incluidos en el documento. Si se requiere su uso deberán ser ofuscados.

3.3 ¿Cuáles son los retos de la información y los datos que utilizará en Integración y Automatización de Datos para IA?

Los datos de consumo de recursos de cada uno de los proyectos serán extraídos a través de la API de AWS, ya sea mediante una lambda para enviar los datos desde AWS o exportados de los registros almacenados en CloudWatch. Posteriormente, los datos serán almacenados en una base de datos y procesados mediante scripts desarrollados en algún lenguaje como JavaScript, Python o Go.

El reto se encuentra en automatizar la extracción de los datos para que esta sea lo más eficiente posible y lograr que se puedan obtener los datos en periodos cortos de tiempo; así mismo, la integración de los datos, ya que estos provienen de diferentes cuentas y distintos recursos por cada cuenta.

3.4 ¿Qué espera de la utilización de Integración y Automatización de Datos para IA para su proyecto?

Se espera obtener los conocimientos necesarios en diferentes herramientas para lograr automatizar los procesos de extracción e integración de los datos desde su origen, la implementación del preprocesamiento y paso por el modelo de machine Learning y finalmente visualización de los datos.

Por otro lado, se desea realizar la automatización de la extracción de los datos desde el servicio de CloudWatch de AWS que permita recopilar datos en periodos cortos o tiempo real desde las múltiples cuentas, transformarlos y limpiarlos eficientemente implementando estos procesos de forma automatizada, para luego alimentar el algoritmo de Machine Learning para el monitoreo y análisis de consumo de recursos en AWS.

4 Diseño de integración y Automatización de Datos para IA (diagrama)

A continuación, se presenta el diagrama del diseño de la propuesta de la automatización e integración de datos para el proyecto propuesto.

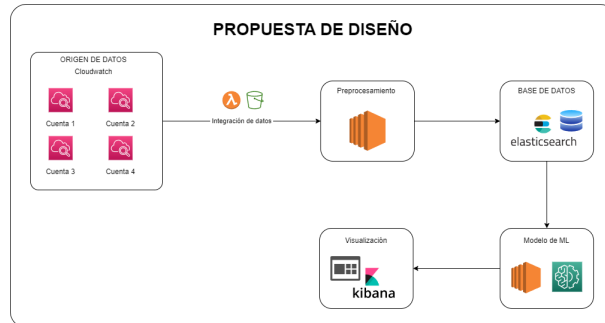


Figure 1: Propuesta diseño de la solución.

En primer lugar, se identificó el origen de los datos a partir del servicio de CloudWatch configurado en cada una de las cuentas de AWS de Grupo ASD. Este proceso se planea realizar utilizando los servicios de Amazon Lambda y S3, en donde se busca automatizar la extracción de los logs mediante una función lambda y posteriormente guardarlos en un bucket.

En segundo lugar, se realizará la integración y preprocesamiento de los datos, para después ser ingestados en la base de datos de ElasticSearch, posteriormente se implementará un modelo de Machine Learning.

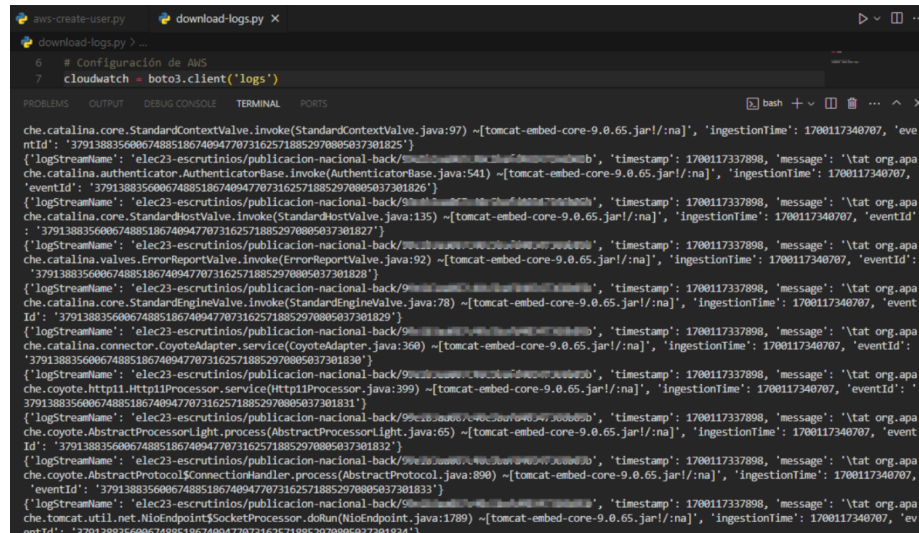
Finalmente, se realizarán los diferentes tableros para el monitoreo de los consumos con la herramienta de visualización Kibana.

5 Integración de datos

En esta fase del proyecto, se llevará a cabo la integración de datos provenientes de los logs almacenados en el servicio de CloudWatch de AWS. Se utilizará un proceso automatizado para extraer, transformar y cargar (ETL) estos datos en una base de datos centralizada. Esto permitirá centralizar los logs de múltiples cuentas y recursos en una única fuente de datos, facilitando así la generación de informes. El enfoque será garantizar la coherencia y la calidad de los datos a lo largo de este proceso. La integración se llevará a cabo de la siguiente manera:

5.1 Extracción de logs

- **Herramienta:** Se utilizará una función Lambda de AWS para automatizar la extracción de registros desde CloudWatch. La función Lambda se activará periódicamente según una programación predefinida o en respuesta a eventos específicos.
- **Frecuencia de Extracción:** La extracción se llevará a cabo en periodos definidos por el cliente, manteniendo un equilibrio entre la actualización de datos en un momento dado y la eficiencia operativa.
- **Implementación:** se usará un script en Python, utilizando el SDK de AWS para extraer los logs de CloudWatch de una cuenta.
- **Selección de logs:** se tendrán en cuenta criterios como la frecuencia de extracción y el grupo de logs y los logs streams.



```
aws-create-user.py  download-logs.py X
download-logs.py > ...
6 # Configuración de AWS
7 cloudwatch = boto3.client('logs')

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
bash + ... X

che.catalina.core.StandardContextValve.invoke(StandardContextValve.java:97) ~[tomcat-embed-core-9.0.65.jar!/:na], 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301825'}
{'logStreamName': 'elec23-escrutinios/publicacion-nacional-back/99e2b3a0667488518674094770731625718852970805037301826', 'timestamp': 1700117337898, 'message': '\tat org.apa
che.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:541) ~[tomcat-embed-core-9.0.65.jar!/:na]', 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301826'}
{'logStreamName': 'elec23-escrutinios/publicacion-nacional-back/99e2b3a0667488518674094770731625718852970805037301827', 'timestamp': 1700117337898, 'message': '\tat org.apa
che.catalina.core.StandardHostValve.invoke(StandardHostValve.java:135) ~[tomcat-embed-core-9.0.65.jar!/:na]', 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301827'}
{'logStreamName': 'elec23-escrutinios/publicacion-nacional-back/99e2b3a0667488518674094770731625718852970805037301828', 'timestamp': 1700117337898, 'message': '\tat org.apa
che.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:92) ~[tomcat-embed-core-9.0.65.jar!/:na]', 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301828'}
{'logStreamName': 'elec23-escrutinios/publicacion-nacional-back/99e2b3a0667488518674094770731625718852970805037301829', 'timestamp': 1700117337898, 'message': '\tat org.apa
che.catalina.core.StandardEngineValve.invoke(StandardEngineValve.java:78) ~[tomcat-embed-core-9.0.65.jar!/:na]', 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301829'}
{'logStreamName': 'elec23-escrutinios/publicacion-nacional-back/99e2b3a0667488518674094770731625718852970805037301830', 'timestamp': 1700117337898, 'message': '\tat org.apa
che.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:360) ~[tomcat-embed-core-9.0.65.jar!/:na]', 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301830'}
{'logStreamName': 'elec23-escrutinios/publicacion-nacional-back/99e2b3a0667488518674094770731625718852970805037301831', 'timestamp': 1700117337898, 'message': '\tat org.apa
che.coyote.http11.HttpProcessor.service(HttpProcessor.java:399) ~[tomcat-embed-core-9.0.65.jar!/:na]', 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301831'}
{'logStreamName': 'elec23-escrutinios/publicacion-nacional-back/99e2b3a0667488518674094770731625718852970805037301832', 'timestamp': 1700117337898, 'message': '\tat org.apa
che.coyote.AbstractProcessorLight.process(AbstractProcessorLight.java:65) ~[tomcat-embed-core-9.0.65.jar!/:na]', 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301832'}
{'logStreamName': 'elec23-escrutinios/publicacion-nacional-back/99e2b3a0667488518674094770731625718852970805037301833', 'timestamp': 1700117337898, 'message': '\tat org.apa
che.coyote.AbstractProtocol$ConnectionHandler.process(AbstractProtocol.java:890) ~[tomcat-embed-core-9.0.65.jar!/:na]', 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301833'}
{'logStreamName': 'elec23-escrutinios/publicacion-nacional-back/99e2b3a0667488518674094770731625718852970805037301834', 'timestamp': 1700117337898, 'message': '\tat org.apa
che.tomcat.util.net.NioEndpoint$SocketProcessor.doRun(NioEndpoint.java:1789) ~[tomcat-embed-core-9.0.65.jar!/:na]', 'ingestionTime': 1700117340707, 'eventId': '37913883560067488518674094770731625718852970805037301834'}
```

Figure 2: Logs extraídos de cuenta ESC.

5.2 Transformación de Datos

- **ETL:** Se implementará un proceso ETL utilizando scripts de Python. Se busca garantizar la coherencia y calidad de los datos antes de cargarlos en la base de datos.
- **Normalización y Estandarización:** Los datos serán normalizados y estandarizados para asegurar la consistencia en la estructura y formato, facilitando así el análisis posterior. En este caso los datos son logs de servicios de AWS, lo que supone llevar a cabo un proceso de organización y formateo de los mismos, de tal forma que sean consistentes para facilitar su análisis y procesamiento. A continuación, algunas consideraciones específicas que se tendrán en cuenta para normalizar y estandarizar logs de servicios de AWS.
 - Identificación de campos relevantes para identificar el consumo de los diferentes recursos de AWS.
 - Extracción de información significativa.
 - Normalización del formato de la data.
 - Conversión de unidades y escalas, como las fechas.
- **Carga en la base de datos centralizada:** Se seleccionará una base de datos adecuada para almacenar los datos extraídos. Esto podría incluir opciones como Amazon RDS, Amazon Redshift o Elasticsearch.
 - Se utilizará el mismo script de Python para realizar la configuración y conexión a la base de datos de ElasticSearch, puesto que esta base almacena datos no estructurados y es ideal para almacenar y buscar logs completos.
 - Asimismo, los logs extraídos se almacenarán en un bucket de S3, el cual se configurará para permitir una gestión de datos escalable y segura.

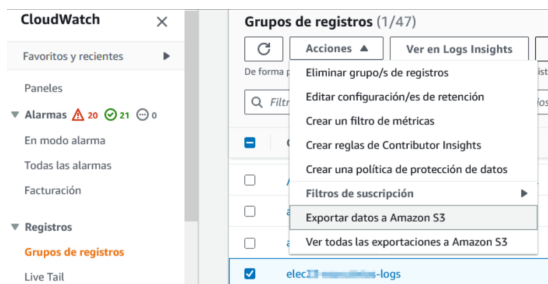


Figure 3: Exportar logs de Cloudwatch a un bucket de S3.

Se realizó una prueba para exportar los logs de cloudwatch a un almacenamiento, en este caso a un bucket de S3, como se observa en la figura anterior. Posteriormente, se automatizará este proceso.

6 Automatización de datos

La automatización de datos garantizará la eficiencia y la oportunidad en la recopilación de información desde CloudWatch. Se busca definir la frecuencia de recopilación de datos, permitiendo un monitoreo en periodos cortos de tiempo y una capacidad de reacción oportuna ante cambios en el consumo de recursos basado en los umbrales definidos por el equipo de analistas.

6.1 Automatización de datos con AWS Lambda

- **Creación de función Lambda:** En el panel de AWS Lambda, se creará una nueva función y se configurarán los permisos necesarios para acceder a los servicios de AWS relevantes, como CloudWatch.
- **Configuración del evento de activación:** Se definirán como evento de activación un horario regular para disparar la lambda.
- **Implementación lógica de extracción:** Dentro de la función Lambda se implementará el script para extraer los logs de CloudWatch de cada una de las cuentas.

6.2 Automatización con docker

- El proceso de instalación de la base de datos Elasticsearch y la herramienta de visualización Grafana se realizará por medio de Docker para asegurar la reproducibilidad y repetibilidad en distintos tipos de ambientes. En la siguiente imagen, se observa el contenedor de grafana corriendo en docker.



```
$ docker ps
CONTAINER ID   IMAGE                  COMMAND                  CREATED        STATUS        PORTS        NAMES
88b1c16d2158   grafana/grafana-oss:latest   "/run.sh"               About a minute ago   Up About a minute   3000/tcp     pensive_blackwell
```

Figure 4: Imagen de grafana en docker.

- El preprocesamiento se encapsulará dentro de un contenedor de Docker, con el fin de asegurar un entorno de ejecución consistente, además de facilitar su implementación en diferentes entornos, ya sea On-Premise o Cloud en servicios como EC2 o ECS.

7 IA

En la fase de Inteligencia Artificial, se implementará un modelo de Machine Learning para analizar patrones de uso, predecir picos de demanda y generar

alertas en función de los datos integrados y procesados. Este modelo permitirá una toma de decisiones más informada y proactiva en relación con la asignación de recursos y la optimización de costos. La elección de algoritmos y la evaluación del rendimiento serán aspectos clave de esta etapa. A continuación, se listan posibles modelos a implementar:

1. **Anomalía de Comportamiento:** Modelos de Detección de Anomalías. Estos modelos pueden identificar patrones inusuales en los datos, lo que es crucial para detectar comportamientos atípicos en el consumo de recursos en la nube. Dentro de los algoritmos que se pueden implementar se encuentran Isolation Forest, One-Class SVM o métodos basados en estadísticas.
2. **Predicción de Demanda:** Modelos de Series Temporales. Estos modelos pueden ser útiles para prever la demanda futura de recursos. Dentro de los modelos que se pueden implementar se encuentran SARIMA (Seasonal AutoRegressive Integrated Moving Average).
3. **Clasificación de Eventos:** Modelos de Clasificación. Estos modelos permitirían clasificar eventos específicos o identificar patrones en los datos. Algoritmos que se podrían implementar: Random Forest, Support Vector Machines (SVM) o redes neuronales.
4. **Segmentación de Recursos:** Modelos de Clustering para agrupar recursos similares o identificar patrones de comportamiento en diferentes grupos. Modelos que se podrían implementar: K-Means.

7.1 Visualización

La fase de visualización desempeña un papel importante al proporcionar una forma visual para la interpretación y monitoreo de los datos. Para esto, se utilizó la herramienta de Grafana para el monitoreo de logs debido a su flexibilidad, capacidad de integración con diversas fuentes de datos, sus capacidades de consulta y visualización, así como configuración para generar alertas y notificaciones.

- Se realiza la configuración inicial de Grafana en ambiente local, para acceder a la herramienta e iniciar su uso.

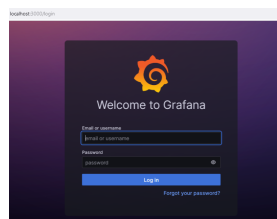


Figure 5: Inicio de Grafana.

- Se realiza la configuración de Grafana para conectarse al servicio de Cloudwatch de una cuenta en AWS, permitiendo la visualización y análisis de los datos de logs.

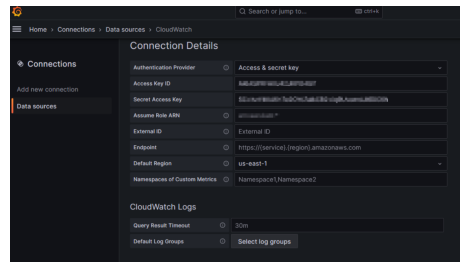


Figure 6: Grafana: Conexión con datasource de Cloudwatch.

- Así mismo, en la siguiente imagen se puede observar la creación de un dashboard inicial en donde se muestran algunas gráficas con los logs de una instancia EC2.

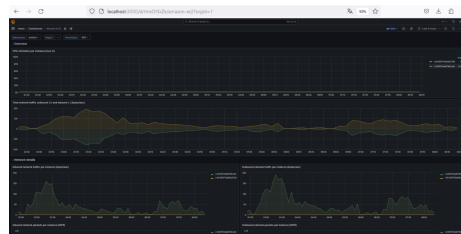


Figure 7: Grafana: Gráficas de logs de EC2.

8 Próximos pasos

En los próximos pasos, se avanzará en la implementación de los componentes restantes del proyecto. Esto incluirá la finalización de la integración de datos y preprocesamiento de estos. El entrenamiento, implementación y ajuste del modelo de Machine Learning seleccionado, y la configuración de los tableros de visualización en Grafana. Además, se llevará a cabo la validación del sistema y se preparará para su despliegue en entornos de producción.

- Desarrollo de script para enviar logs desde todas las cuentas que se encuentren en la organización (AWS Organizations) a la base de datos centralizada.
- Creación y configuración de las funciones lambda en cada una de las cuentas. Creación de los roles y políticas necesarias para la lambda y S3.
- Prueba de concepto de servicios como AWS Glue para realizar el preprocesamiento de los datos, antes de ser enviados a la base de datos Elasticsearch y el bucket de almacenamiento de los logs.
- Definir y configurar la política de retención y almacenamiento de los logs en el bucket S3.
- **Proceso Automatizado:** La carga de datos se realizará mediante procesos automatizados para mantener la consistencia de los datos. Se implementará un preprocesamiento para garantizar la calidad y buen funcionamiento del modelo de Machine Learning a implementar.
 - **Validación de formato y estructura:** Antes de cargar los datos al modelo de Machine Learning, se verifica la estructura de cada registro cumplan con los requisitos especificados por el cliente, esto facilita la posterior consulta y análisis.
 - **Verificación de valores nulos o atípicos:** Se realizan comprobaciones para identificar y manejar valores nulos o atípicos. Esto puede incluir la detección de valores fuera del rango esperado o la identificación de campos críticos que no deben estar vacíos.
 - **Control de duplicados:** Se implementan controles para evitar la carga de registros duplicados. Esto puede incluir la comparación de registros basada en identificadores únicos o combinaciones específicas de campos.
 - **Pruebas de Integridad Lógica:** Est paso podría incluir la validación de relaciones lógicas entre los diferentes campos de los logs para asegurar que los datos sean coherentes desde un punto de vista de negocio.

La implementación de estos mecanismos de control busca asegurar que los datos integrados sean confiables y estén listos para su uso en análisis y

toma de decisiones. Además, facilita la identificación y corrección proactiva de posibles problemas en el proceso de carga.

- Desarrollo del modelo de machine Learning, entrenamiento y despliegue de este.
- Construcción del dashboard con las gráficas necesarias para realizar el monitoreo de los recursos.

9 Lecciones aprendidas

A continuación, se listan algunas de las lecciones aprendidas en la implementación de una solución en la integración y automatización de datos para la extracción y análisis de logs en AWS, y su visualización mediante Grafana.

- Importancia de la planificación y diseño de una arquitectura: La complejidad y los desafíos que presenta la integración de distintos servicios en la nube, así como On-Premise resaltan la necesidad de un diseño de la solución que permite el entendimiento del flujo de los datos y, los servicios y herramientas que interactuarán a lo largo del proceso.
- Automatización: Además de mejorar la eficiencia y tiempo de respuesta del sistema, también reduce el riesgo de errores humanos en el cargue y procesamiento de los datos.
- Flexibilidad y adaptabilidad: La tecnología y los requisitos de negocio cambian rápidamente. La solución implementada debe ser lo suficientemente flexible para adaptarse a cambios y nuevas integraciones, manteniendo la capacidad de actualizarse con mínimas interrupciones.
- Documentación de la solución implementada: La documentación facilita la comprensión y el mantenimiento continuo del sistema, así como la capacitación a los equipos que utilizarán la solución.

10 Bibliografía

References

Amazon Web Services Inc. (2023). *Información general sobre Amazon Web Services*. Documento técnico de AWS. Recuperado el 30 de abril de 2023, de: <https://aws.amazon.com/es/>

Grafana. (2023). *Amazon CloudWatch data source documentation*. Recuperado el 10 de noviembre de 2023 de: <https://grafana.com/docs/grafana/latest/datasources/aws-cloudwatch/>

Pragma USA Inc. (s/f). *Computación en la nube: todo lo que un experto debe saber*. Recuperado el 30 de abril de 2023, de: <https://www.pragma.com.co/academia/conceptos/computacion-en-la-nube>