Universidad de Lima

Facultad de Ingeniería

Carrera de Ingeniería de Sistemas

IMPLEMENTACIÓN DE UN LAKEHOUSE EN UNA EMPRESA DEL SECTOR CONSUMO MASIVO

Trabajo de suficiencia profesional para optar el Título Profesional de Ingeniero de Sistemas

Luis Giancarlo Sanchez Huaman

Código 20161333

Asesor

Jorge Victor Miranda Pacheco

Lima – Perú

Octubre de 2024

IMPLEMENTING A LAKEHOUSE SOLUTION IN A MASS CONSUMPTION INDUSTRY COMPANY

TABLA DE CONTENIDO

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

RESUMEN

El presente informe tiene como propósito contar sobre mi experiencia profesional en una empresa de consumo masivo, detallando cuales han sido los roles y responsabilidades que he asumido a lo largo de mi puesto, destacando la creación de una solución a una necesidad de la compañía empleando una arquitectura de datos sostenible y escalable en el tiempo; mi participación como miembro activo y líder de un equipo para el desarrollo de las iniciativas comerciales de la compañía proponiendo soluciones y respuestas ante los requerimientos dados por el negocio como la disponibilización de datos o la integración con algunas aplicaciones; el aprendizaje continuo dado a la constante evolución de la tecnología y al mercado competitivo; y el seguimiento de las normas y responsabilidades profesionales adquiridas en mi aprendizaje. Siendo así, los logros obtenidos en este tiempo son la creación de un lakehouse inicialmente para el área de Transformación Digital para escalar a áreas de negocio, facilitar la convivencia de los datos de dos sistemas ERPs, disponibilización de datos para distintas áreas, optimización y estandarización de procesos, y liderar iniciativas comerciales usadas por toda la fuerza de venta a nivel nacional. Un ingeniero de Sistemas debe estar preparado para asumir el reto no solo a nivel técnico sino a nivel de gestión, ya que la oportunidad de poder liderar un equipo es una excelente manera de consolidar y reforzar lo aprendido durante la vida profesional. Para poder desempeñar mi rol he estado en constante autoaprendizaje, por medio de cursos o investigación, y recurriendo a mis pares y jefes, los cuales siempre me han apoyado con una explicación o un consejo para abordar un tema. En este informe se revisará mi experiencia en cuatro capacidades: técnica, gestión, aprendizaje continuo y conducta ética, para finalizar con las lecciones aprendidas y las recomendaciones finales.

Palabras clave: Lakehouse, Big Data, Arquitectura de datos, Empresa, Google Cloud Plataform (GCP), Datos.

ABSTRACT

The purpose of this report is to tell about my professional experience in a mass consumption company, detailing the roles and responsibilities I have assumed throughout my position, highlighting the creation of a solution to address the company's need using a sustainable and scalable data architecture; my participation as an active member and leader of a team for the development of the company's commercial initiatives proposing solutions and responses to the requirements given by the business users such as data availability or integration with some applications; continuous learning due the constant evolution of technology and the competitive market; and monitoring of the standards and professional responsibilities acquired in my learning. The achievements obtained during this time are the creation of a lakehouse initially for the Digital Transformation area to scale it to business areas, facilitating the coexistence of data from two ERP systems, data availability for different areas, optimization and standardization of processes, and leading commercial initiatives used by the entire national sales force. A Systems Engineer must be prepared to take on the challenge not only at a technical level but also at a management level, since the opportunity to lead a team is an excellent way to consolidate and reinforce what has been learned during professional life. To perform my role, I have been constantly self-learning, through courses or research, and turning to my peers and bosses, who have always supported me with an explanation or advice about a topic. This report will review my experience in four capacities: technical, management, continuous learning and ethical conduct, to end with the lessons learned and final recommendations.

Keywords: Lakehouse, Big Data, Data architecture, Company, Google Cloud Plataform (GCP), Data.

INTRODUCCIÓN

Tras culminar mis estudios de pregrado en la carrera de Ingeniería de Sistemas de la Universidad de Lima en julio de 2021 se me presentó la oportunidad de trabajar como consultor de datos en el área de Transformación Digital de una de las empresas de consumo masivo con mayor presencia en el mercado peruano.

El área de Transformación Digital era un área relativamente nueva, comenzó sus actividades en 2018 y estaba enfocada en la innovación y el desarrollo de iniciativas analíticas a través de modelos, mi rol consistía en la disponibilización de data, automatización y despliegue de procesos; donde los conocimientos adquiridos en el curso de Machine Learning y Analítica Predictiva de Datos me sirvieron para poder entender y trabajar de la mano con el equipo de Data Science.

Durante el año 2022, el área comenzó a tener mayor presencia dentro de la compañía, ya que no solo se enfocó en el desarrollo de nuevos modelos analíticos sino en la habilitación y explotación de la data que se encontraba dispersa en varios sistemas, lo cual generaba problemas si se requería un análisis cruzado. A finales del mismo año, ingresé a la empresa como Data Engineer para el equipo de Big Data con el propósito de poder sumar en la creación de la arquitectura de datos, lakehouse.

Para poder llevar a cabo esta actividad, era necesario conocer los sistemas origen de donde extraer la información siendo la mayoría de ellos sistemas ERP. Uno de los más importantes para el área comercial es un sistema SAP NetWeaver con módulos personalizados, el cual contiene la información de la venta de sus clientes finales (Sell Out). En este escenario, los conocimientos adquiridos en el curso de Sistemas ERP me ayudaron a conocer cómo era el funcionamiento y algunas transacciones para poder visualizar la data. Asimismo, el curso de Modelación e Integración de Sistemas me permitió tener una visión general del proyecto, ya que es necesario que la data que viene de distinta fuente pueda relacionarse entre sí.

Al obtener y conocer la data es necesario comenzar a trabajarla, para ello los conocimientos adquiridos en Sistemas de Inteligencia Empresarial e Ingeniería de Datos me dieron una base para poder entender y trabajar con los datos, no solo a nivel técnico como el uso de lenguaje SQL, si no también enseñándome los diferentes enfoques que existían en el mercado. En paralelo,

resultaba fundamental diseñar una arquitectura que facilitara la implementación de esta propuesta, permitiendo orquestar los procesos a ejecutar, para lo que los cursos de Lenguaje de Programación e Ingeniería de Software me ayudaron a conocer las mejores prácticas y los estándares para desarrollar un proyecto que sea sostenible en el tiempo.

Tras tres años en los que he visto varios proyectos asociados al lakehouse, me ha tocado trabajar con distintos equipos tanto dentro como fuera del área, y para ello, el curso de Gestión de Proyectos e Ingeniería de Procesos de Negocio, me dieron las herramientas y las formas de trabajar con equipos interdisciplinarios en los cuales todos participen y tengan un rol dentro del proyecto.

En este año 2024, el área ha crecido de manera exponencial, tanto a nivel de personas como de responsabilidades, ya que el alcance cambió, ya no solo disponibilizamos datos para el equipo propio, si no que, ahora contamos con presencia dentro del negocio y surgen cada vez más iniciativas que requieren la participación activa del equipo de Big Data.

En este informe se explicará este proceso de selección, análisis e implementación de la solución en la sección de capacidad técnica, así como las diferentes iniciativas y escenarios que han surgido gracias a ello. En la sección de capacidad de gestión se explicará acerca de la práctica realizada con los equipos, la forma de trabajo y el rol desempeñado. En la siguiente sección de aprendizaje continuo, se detallará el proceso de educación realizado a través de la experiencia de las iniciativas o el aprendizaje propio. Finalmente, se presentará la sección de conducta ética donde se explicará el manejo y el buen uso de la información confidencial, así como la gestión y el trato con proveedores u otros equipos de trabajo.

1. CAPACIDAD TÉCNICA

Uno de los principales problemas que ha afrontado la compañía en este tiempo está relacionado con la centralización de datos, ya que siempre se han trabajado desarrollos en función de una necesidad especifica, sin considerar la posibilidad de escalarlos, lo que ha limitado la explotación de los datos y ha generado silos separados.

El área de Transformación Digital, por su parte, se enfocaba en aportar valor al negocio mediante la tecnología, ya sea a través de la creación de segmentos de clientes basados en su comportamiento de compra (clusters), sistemas de recomendación, entre otros. El proceso de data empleado era exclusivo para los modelos analíticos, lo cual generaba un nuevo silo ajeno al llevado por el negocio que trabaja con un sistema de reportería proporcionado por SAP.

El proceso a nivel de datos no era complejo, ya que se contaba con una única fuente que agrupaba los datos ya modelados en 3 grandes grupos: Venta, Cliente y Producto. Con esta información, el equipo de Data Science desarrollaba sus modelos, complementándolos con datos dados por el negocio o generados a través de encuestas. Esta información modelada se obtenía a partir de un desarrollo realizado por una consultora en Qliksense.

Este enfoque funcionó durante algunos años, pero un cambio en el origen provocó que los datos extraídos por este medio resulten incompletos al cabo de unos meses. Cualquier modificación implicaría un gasto adicional, además de presentar una dependencia con la consultora si en el futuro se necesitara un ajuste. Por ello, en paralelo se activaron diversas iniciativas para la obtención de estos datos de los sistemas ERPs, ya que se buscaba disminuir la dependencia de un tercero y, al mismo tiempo, explotar otros casos de uso que estaban apareciendo.

Es necesario destacar que la mayor parte de la información de la compañía se encuentran en sus sistemas ERP, los cuales reciben datos de diferentes aplicativos; como móviles, web u otras plataformas de usuario. Según Sisyukov et al., (2020) un ERP es un sistema de planificación de recursos empresariales que sirve como base para la interconexión de sistemas logísticos, sistemas de producción, gestión de clientes, entre otros; los cuales generan flujos de datos consolidados y no procesados. Estos datos son valiosos para la compañía, ya que la explotación de los mismos permite que la información no quede solo a nivel transaccional del día a día sino darle un valor agregado. Para poder obtener ello, Sisyukov et al., (2020) nos presenta la forma de extraer estos datos a través del protocolo de SAP Remote Function Call (RFC), utilizando conectores SAP

estándar, bibliotecas pyrfc/noderfc para el llamado de funciones del módulo Advanced Business Application Programming (ABAP) o el uso de ODBC/JDBC para conectarse directamente a los sistemas SAP.

Aplicando lo anterior al presente caso, y tras coordinar con el equipo de TI para evaluar las opciones planteadas, considerando los accesos y el performance, se decidió emplear la librería pyrfc en Python, utilizando un API para la transferencia a un Dataframe. Esto no solo permitió poder obtener los datos de las ventas, sino cualquier dato que se encontrara en el sistema SAP NetWeaver personalizado o SAP S/4 HANA, lo que abrió la posibilidad de trabajar con una mayor cantidad de data que anteriormente no se contaba. Esto, a su vez, permitió enriquecer los modelos analíticos. Sin embargo, esto aún no presentaba un impacto para el negocio, ya que ellos seguían consumiendo la información de su reportería.

Dentro de los objetivos planteados por la compañía para el año 2023 se encontraba el realizar una migración de sus sistemas ERP de SAP a Odoo para toda la información relacionada a la venta Sell Out. Esta migración presentó un gran problema para los sistemas de información de la compañía, ya que el cambio tendría que ser progresivo y necesitaría una convivencia entre ambos sistemas ERP para que el negocio no se vea afectado en su día a día. Esta problemática llevó a que la compañía necesitase una solución de data que permita esta convivencia y sirva como fuente de información. Dicha necesidad, se convirtió en una oportunidad para el área, ya que permitió agrandar el alcance del lakehouse, no solo proporcionando información al equipo de Data Science, sino a nivel de la compañía.

Esta gran iniciativa permitió al negocio conocer acerca de los beneficios de tener un lakehouse, lo que impulsó más iniciativas por parte del área de TI y del negocio para disponibilizar y explotar la información. Al darse cuenta de la facilidad con la que tendrían acceso y la autonomía para poder generar sus propios análisis e impulsar a la empresa a un modelo Data Driven.

Entre los proyectos trabajados se encuentra:

- Implementación de la arquitectura del lakehouse
- Convivencia de sistema ERP (SAP y Odoo)
- Integración con aplicativos finales (Power BI, Aplicación Móvil o Web)

Antes de presentar la arquitectura trabajada, es necesario revisar las opciones que se plantearon tras el cambio del alcance, ya que el público objetivo se amplió incluyendo al equipo de Data Science y el de negocio. Por ello, se revisaron diferentes propuestas que se encontraban

vigentes en el mercado y su recepción por parte de los usuarios, destacando el uso de data lake, data warehouse y lakehouse.

Una de las primeras propuestas consideradas a trabajar fue un data lake. Según Raju et al., (2018) y Nargesian et al., (2019), un data lake consiste en una colección masiva de conjuntos de datos de distintos sistemas, distintos formatos y versátil en el tiempo, ya que al no contar con una estructura definida esta puede modificarse. Además, actúa como una copia exacta de lo que se encuentra en el sistema y está orientado a usuarios de Data Science. Por otro lado, Zaharia et al., (2021) y Schneider et al., (2023) definen el data warehouse, como una propuesta más clásica empleando una base de datos relacional dónde se garantiza las propiedades de Atomicidad, Consistencia, Aislamiento y Durabilidad (ACID). Este enfoque se centra en el modelado y la gobernanza de los datos, con el objetivo de proporcionar a los líderes empresariales información analítica para la toma de decisiones y el desarrollo de la inteligencia empresarial (BI).

Los investigadores también comentaron sobre los problemas asociados a cada enfoque. En el caso del data lake al presentar una flexibilidad en su composición no proporcionaba una alta robustez y carecía de funciones de gestión; además, dependía en gran medida de la metadata para poder entender el dato a trabajar, y no aseguraba la calidad ni la gobernanza en la mayoría de los casos (Schneider et al., 2023; Zaharia et al., 2021). Por otro lado, los data warehouses empezaban a presentar problemas en la escalabilidad, ya que se necesitaba una mayor inversión por la cantidad de usuarios concurrentes trabajando con grandes volúmenes de datos. Asimismo, presentaban una limitación con los nuevos datos emergentes, como el caso de audios o videos (Zaharia et al., 2021).

Tras revisar las ventajas y desventajas de cada enfoque, la misma necesidad del mercado empezó a presentar un modelo híbrido, donde se buscaba converger ambas propuestas, que inicialmente parecen distantes, en un punto de equilibrio. Esto respondía a una necesidad de aprovechar ambos enfoques en paralelo. (Schneider et al., 2023). A partir de esta premisa, varios investigadores empezaron a trabajar en propuestas que se adapten a las nuevas necesidades de un mercado más competitivo y cambiante.

Un ejemplo de ello fue el presentado por Raju et al., (2018), quien propone trabajar sobre un data lake con zonas, debido a los beneficios que ofrece, en el que destaca la centralización de los datos para la eliminación de silos y un modelo de autoservicio para los usuarios. El uso de las zonas permitió ordenar el data lake definiendo actividades en cada una de ellas. La primera zona es la ingesta de datos, donde los datos de distintas fuentes llegan a un solo punto. A continuación, la zona de datos procesados, donde se filtra y organiza los datos. Finalmente, llega a la zona de

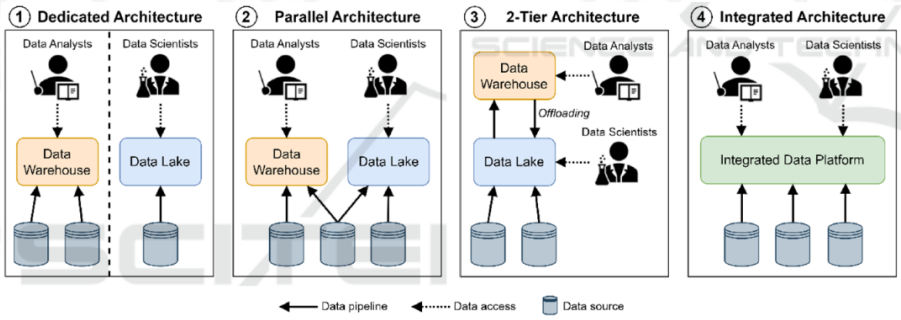
datos refinados, que es utilizada por las herramientas de análisis para la visualización, búsqueda y modelado de datos. Este último punto se acerca al concepto definido en el data warehouse, ya que presenta características que no se cuenta en un data lake convencional, demostrando que la aplicación de un modelo híbrido no es tan atípica y los conceptos pueden entrelazarse.

Tras conocer un ejemplo de un caso híbrido, se presentarán los escenarios más comunes en las compañías, con la finalidad de conocer el mercado actual, y cómo el proceso puede ir evolucionando hasta tener una propuesta híbrida que cumpla las expectativas de todos los usuarios.

En la Figura 1.1 se presentan los escenarios propuestos identificados por Schneider et al., (2023), ello no difiere de lo presentado por Zaharia et al., (2021) y también es comentada en menor medida por Oreščanin et al., (2024). El escenario 1 (Arquitectura dedicada) representa una forma de trabajo simple, donde el data lake y el data warehouse trabajan de manera independiente, sin buscar una interacción entre ambas arquitecturas. En este enfoque, el objetivo es el cumplimiento de los objetivos establecidos, sin importar la creación de silos de datos.

Figura 1.1

Arquitectura en las empresas



Nota. De "Assessing the Lakehouse: Analysis, Requirements and Definition" por J. Schneider, C. Gröger, A. Lutsch, H. Schwarz, & B. Mitschang, 2023, *Proceedings of the 25th International Conference on Enterprise Information Systems*, p. 46. (https://doi.org/10.5220/0011840500003467)

El escenario 2 (Arquitectura paralela) se caracteriza por la existencia de un origen de datos en común. Surge de la necesidad de ambos tipos de usuarios de usar la misma data, lo que genera redundancia, ya que esta se estaría encontrando en ambas plataformas. Esto puede resultar en respuestas contradictorias ante una misma pregunta y un aumento de los costos.

El escenario 3 (Arquitectura 2 niveles) es reconocido por los tres investigadores como la propuesta con mayor presencia dentro de las compañías, contando con 2 niveles (data lake + data warehouse). En este enfoque, todos los datos primero se presentan en el data lake, siendo esta la fuente de datos a utilizar por el equipo de Data Science, y en función de los requerimientos de negocio se habilita un data warehouse con lo solicitado; realizando, en la mayoría de las veces,

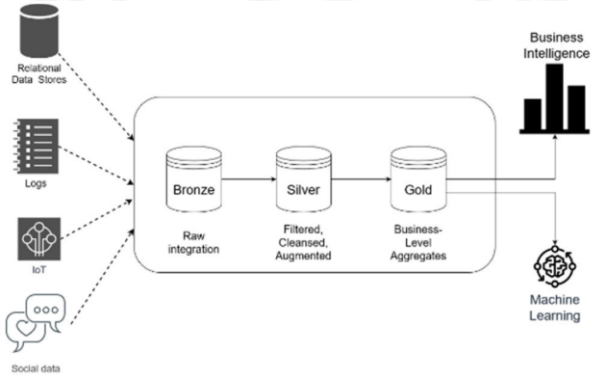
una copia del data lake, lo cual puede presentar errores en la conversión, añade complejidad, mayor opción a un fallo y presenta un aumento en los costos por el lado del data warehouse.

Esto conlleva al escenario 4 (Arquitectura integrada) en el que se cuenta con una visión de una única plataforma de datos que combina las características de ambos enfoques. Permite responder a iniciativas analíticas y consultas de negocio sin tener que apoyarse de procesos adicionales que disminuya el performance. Esta nueva forma de arquitectura se le conoce como lakehouse, el cual es definido como un sistema de gestión de datos en el medio de un data lake y un data warehouse basado en un almacenamiento de bajo costo y de acceso directo; cuenta como características la propiedad ACID, el empleo de esquemas flexibles, indexación, optimización de consultas entre otros (Schneider et al., 2023).

Oreščanin et al., (2024) presenta una propuesta que se basa en lo comentado por los anteriores investigadores, pero añade un enfoque adicional al incorporar el uso de zonas, garantizando la persistencia y los niveles de acceso en cada una de ellas. Esta nueva forma de emplear el lakehouse se basa en la arquitectura Medallion. La principal diferencia con el modelo de 2 capas es la integración del data warehouse dentro de la arquitectura, en lugar de considerarlo como una salida del mismo. De esta manera, se busca un enfoque definido que asegure la calidad y la estructura de los datos. El propósito de las capas es que estos vayan mejorando en función de su cercanía con la capa de consumo.

Figura 1.2

Arquitectura Medallion



Nota. De "Managing Personal Identifiable Information in Data Lakes" por D. Oreščanin, T. Hlupić & B. Vrdoljak, 2024, *IEEE Access*, p. 32168. (https://doi.org/10.1109/ACCESS.2024.3365042)

En la Figura 1.2 se visualiza la propuesta de la arquitectura Medallion, que consiste en tres capas. La capa Bronze es la encargada de tener los datos sin procesar y sin un esquema, tal cual se obtienen de los sistemas origen. A continuación, en la capa Silver se filtran y limpian los datos; asimismo, se define la estructura y el esquema que empleará el modelado de datos. Esta capa es

importante ya que asegura la calidad de los datos. La última capa es la Golden, la cual presenta los datos consolidados y listos para su consumo empleando un modelo estrella o data marts. En esta capa donde los usuarios de negocio y el equipo de Data Science pueden acceder los datos para sus iniciativas.

Adicional, a ello, Oreščanin et al., (2024) propone el uso de un nuevo concepto, llamado Data Mesh, que consiste en tratar al dato como un producto dentro de una plataforma de autoservicio; esto con la finalidad de que las empresas comiencen a basarse más en datos y reducir los silos por medio de la democratización del dato.

El desarrollo de un lakehouse involucra el seguimiento de buenas prácticas, ya que debe ser sostenible y escalable en el tiempo. Para ello, Mazumdar et al., (2023) presenta un conjunto de pautas para cumplirlo, como el uso del modelado, importante para estandarizar y reflejar las relaciones entre los datos y conceptos de negocio; el uso eficiente de ETLs, evitando los reprocesos y ejecutándolos en los momentos necesarios, como en la extracción en la capa Bronze; y calidad de datos, ya que permite garantizar una precisión, coherencia y fiabilidad en el lakehouse.

Finalmente, en la Tabla 1.1, se presenta un resumen de las tres arquitecturas comentadas, destacando sus características principales e información relevante en la toma de decisión de escoger una u otra.

Tabla 1.1

Cuadro comparativo entre arquitecturas

	Data lake	**Data warehouse**	**Lakehouse**
Concepto	Colección masiva de datos de diferentes sistemas y formatos obtenidos del sistema	Colección de datos empleando una base de datos relacional con una estructura definida	Sistema de gestión de datos que comparte características de un data lake y un data warehouse
Usuarios	Data Science	Usuarios de negocio	Data Science y usuarios de negocio
Uso	Investigación y desarrollo de modelos	Análisis de inteligencia empresarial	Análisis de inteligencia empresarial, investigación y aplicación de analítica avanzada
Ventajas	- Flexibilidad para incluir información de diferente estructura - Escalabilidad	- Garantiza la propiedad ACID - Fácil comprensión para la toma de decisiones de negocio	-Garantiza la propiedad ACID - Empleo de un esquema flexible - Optimización en consultas
Desventajas	- Alta dependencia en metadata - No se garantiza una calidad del dato	- Problemas de escalabilidad - Limitación con nuevos tipos de datos	Una arquitectura “compleja” con relación a las anteriores ya que se desenvuelve por capas

1.1 Implementación de la arquitectura del lakehouse

Tras decidir la arquitectura de datos a trabajar, se comenzó a evaluar las distintas alternativas en los sistemas cloud, ya que la compañía busca impulsar el uso de estos sistemas sobre los on-premise, debido a la facilidad en la administración de servicios, factibilidad del escalado y a los socios estratégicos con los que cuenta. Para ello, se evaluaron tres diferentes nubes con la que trabajan las áreas pares; Amazon Web Services (AWS), Microsoft Azure y Google Compute Engine (GCP).

Algunos estudios similares fueron realizados por Chauhan (2020), Falah et al., (2021) y Borra (2024), los cuales buscaron definir y explicar las ventajas y desventajas de cada nube, con el fin de poder dar el mayor detalle para el uso de estas. Los tres autores coinciden en la importancia del cloud computing en las empresas, ya que permite el acceso a la información y a los datos mediante una red de Internet, así como la disponibilización de recursos informáticos de manera rápida y a demanda.

Dentro de las opciones que ofrece el mercado destaca como se brinda el servicio, los cuales son Infraestructura como Servicio (IaaS), Plataforma como Servicio (PaaS) y Software como Servicio (SaaS), aumentando la administración de la nube del primero al último. Además, los conceptos de nube privada, pública e híbrida comienzan a tener relevancia, ya que varias empresas desean tener un ecosistema privado por temas de control y seguridad. Sin embargo, esto representa una mayor inversión y gestión a diferencia de una nube pública, que ofrece gran escalabilidad y versatilidad en la administración de los recursos. Por último, existe una opción híbrida, donde se puede distribuir la carga en ambos administradores. Para poder comparar las nubes se utilizaron criterios como especificaciones tecnológicas, Big Data, Machine Learning, precio, rendimiento, interfaz de usuario y gestión de la nube.

La comparación realizada por los autores tiene diferentes niveles de detalle, ya que se presenta una comparativa a nivel servicios para un caso en específico (Falah et al., 2021), comparativa entre servicios de cada uno (Borra, 2024) y comparación a nivel de negocio (Chauhan, 2020). A continuación, se presenta un consolidado de todas estas comparaciones para tener diferentes perspectivas y enfoques en la decisión de la nube a trabajar.

Amazon Web Services (AWS) proporciona varios servicios en función de las tecnologías emergentes, servicios de Big Data y streaming de datos con la finalidad de poder trabajar con data en tiempo real. Asimismo, los autores lo definen como uno de los más costosos, pero con un rendimiento destacado y cuenta con servicios como AWS S3, Amazon EMR entre otros.

Google Cloud Platform (GCP) proporciona varios servicios integrados para el análisis de datos con una alta performance y con costos eficientes. Además, permite el uso de varios lenguajes de programación e integración con aplicaciones desarrolladas por los usuarios. Es considerada como uno de los más amigables para nuevos usuarios ya que brinda la información precisa para trabajar y entre sus servicios destacan BigQuery, Cloud Pub/Sub, Cloud Composer entre otros.

Microsoft Azure es un servicio de computación que destaca por el manejo de nubes híbridas y el uso de servicios de analítica avanzada. Además, ofrece varios servicios de implementación de Big Data para la distribución y procesamiento de los datos en entornos compartidos; como servicios tiene Azure HDInsight, Blob Storage, Azure Functions, entre otros.

Cabe resaltar que los conceptos de los servicios son compartidos por las tres nubes; por ejemplo, para el guardado de objetos AWS emplea AWS S3, GCP utiliza Cloud Storage y Azure usa Blob Storage; los tres servicios cumplen el mismo rol, pero son denominados de manera diferente, esto permite que la mayoría de los conceptos puedan abstraerse de la nube elegida. Como conclusión se destacó a GCP sobre las otras nubes en relación con la especificación tecnológica, la simplicidad para administrar los servicios en la nube, fortaleza en Big Data y los bajos costos en situaciones similares; mientras que Azure y AWS presentan mejores propuestas para las tecnologías emergentes y funcionalidades.

A continuación, en la Tabla 1.2, se presenta un cuadro con un resumen de la comparación de las tres nubes.

Tabla 1.2

Cuadro comparativo entre nubes

	Amazon Web Services (AWS)	**Google Cloud Platform (GCP)**	**Microsoft Azure**
Ventajas	- Red masiva de data centers - Variedad de servicios	- Simplicidad en la administración - Bajos costos -Interfaz amigable e intuitiva	- Manejo de nubes hibridas - Exploración en servicios de Inteligencia Artificial
Desventajas	- Altos costos a comparación de los otros	- Soporte con mayor tiempo de respuesta que los otros	- Dificultad en curva de aprendizaje
Servicios	AWS S3, Amazon EMR	BigQuery, Cloud Pub/Sub, Cloud Composer	Azure HDInsight, Blob Storage, Azure Functions

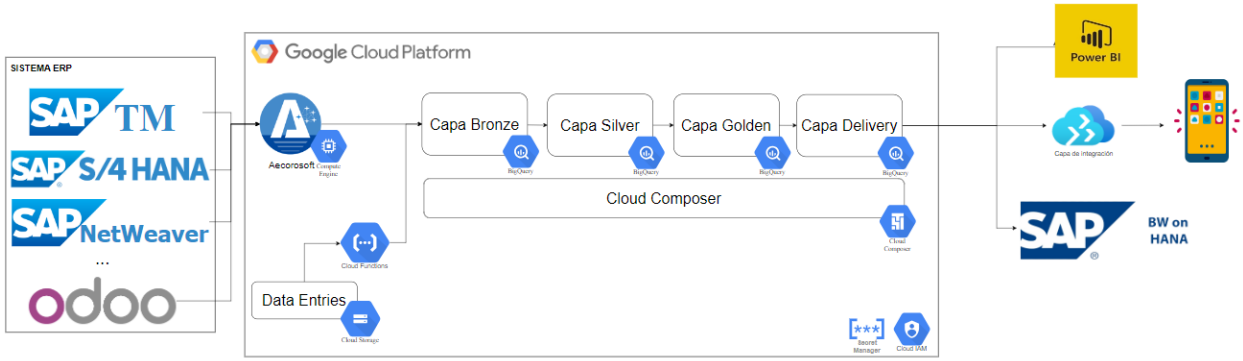
La elección de la nube fue Google Cloud Platform (GCP), debido a las conclusiones mencionadas anteriormente sumadas a la experiencia del equipo, lo cual reducía la curva de aprendizaje. Dentro de los servicios a utilizar destacan el uso de BigQuery, Cloud Composer,

Cloud Functions, Cloud Storage, Secret Manager entre otros. Con ello, se buscó crear una arquitectura robusta para poder afrontar las necesidades y cumplir las expectativas de negocio. Sin embargo, aún existía un proceso que podía optimizarse, la extracción de los sistemas ERP, ya que la solución desarrollada permitía la descarga, pero en tiempos amplios, generando que los reprocesos o el aumento de la frecuencia de actualización sean difíciles de manejar. Es por ello, que tras la revisión de varias propuestas se decidió emplear Aecorsoft, el cual es una herramienta que optimiza lo desarrollado en los inicios con pyrfc reduciendo considerablemente los tiempos y simplificando la configuración.

A continuación, en la Figura 1.3, se presenta la arquitectura trabajada, la cual cumple con los casos de uso presentados por negocio y permite la escalabilidad del mismo por si es necesario realizar algún cambio o mejora en el proceso.

Figura 1.3

Arquitectura Lakehouse



En la Figura 1.3 se tiene la arquitectura del lakehouse donde se visualiza la arquitectura Medallion, los sistemas ERP que actualmente están llegando al lakehouse por medio del servicio de extracción de data Aecorsoft para la capa Bronze, los aplicativos finales que consumen la información generada en el lakehouse y algunos servicios de GCP que complementan o dan soporte a la arquitectura presentada, como el uso de Cloud Function para la ingesta de Data Entries, Cloud Composer para la orquestación de procesos, BigQuery para las capas, Secret Manager para el guardado de las variables sensibles y el Identify and Access Management (IAM) para el tema de permisos.

Como se visualiza, la capa Bronze presenta dos formas de poblar sus tablas, a través de Aecorsoft para los sistemas ERP y Cloud Function para los archivos proporcionados por negocio (Data Entries), el uso de estos se restringe a datos que no se encuentran dentro de un sistema ERP y aportan valor a lo desarrollado en el lakehouse. Los tiempos de respuesta de ambos servicios

son buenos, ya que Aecorsoft permite la extracción de los datos de tablas pesadas en un máximo de 10 minutos, gracias a una configuración de Captura de Datos Modificados (CDC) optimizando y reduciendo los tiempos de descarga, asimismo la Cloud Function ingesta en cuestión de segundos una tabla pequeña dentro del lakehouse.

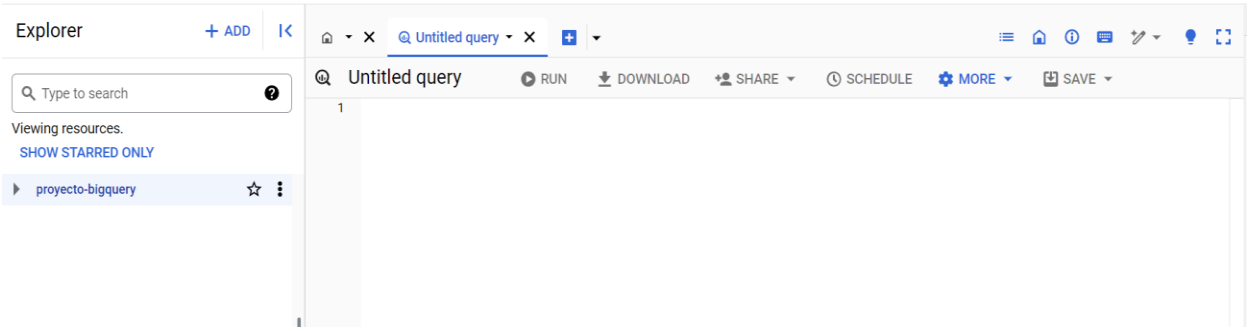
Tras ello, continúa la capa Silver en el que se da inició al proceso de modelado de datos. Este proceso consiste en ajustar los conceptos recibidos en Bronze acercándolo a términos empleados por el negocio, con la finalidad de tener un modelo propio de la empresa en un espacio estandarizado con datos limpios y de calidad. En esta parte del proceso participa el equipo de Data Governance, ya que ellos aplican las reglas de calidad según lo indicado por los usuarios, asimismo evalúan y miden métricas sobre ello. Luego se pasa a la capa Gold, donde la información se consolida para ser usada por los usuarios de negocio y los Data Scientists, en esta capa nacen las iniciativas empleando datos de calidad y gobernados. Finalmente, se cuenta con una cuarta capa, Delivery, la cual permite automatizar las tablas de las iniciativas para el consumo de los aplicativos finales, esto se trabaja así ya que no es necesario consultar toda la información de Gold sino un resumen previamente evaluado y revisado.

Todo lo comentado anteriormente de las capas se desarrolla en el servicio de BigQuery, ya que proporciona costos bajo demanda según el uso y tiempos de respuesta óptimos debido al almacenamiento columnar en el que opera. Además, su uso por parte de los usuarios no presenta mucha complejidad, ya que se basa en el lenguaje de consultas SQL.

En la Figura 1.4 se presenta la interfaz donde se ejecutan las consultas. Se visualiza que es una hoja limpia donde no se cuenta con demasiada información para evitar distraer a los usuarios, y se cuenta con un panel ubicado en la parte izquierda donde se ven las tablas a las que tiene acceso por cada proyecto. En el espacio en blanco, el usuario puede colocar su consulta para ir conociendo la data.

Figura 1.4

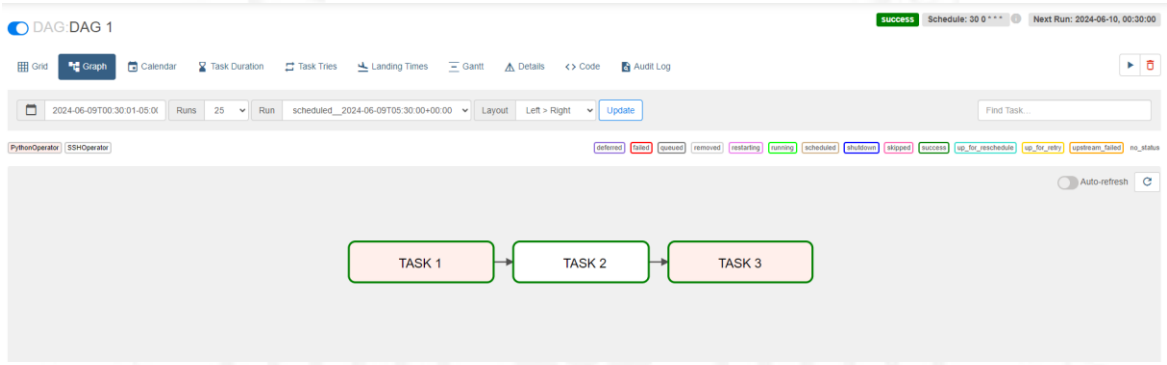
Entorno BigQuery



BigQuery es el servicio que almacena los datos, pero otro servicio que tiene gran incidencia en todo el flujo es Cloud Composer; este servicio está construido sobre Apache Airflow y tiene como principal función simplificar la orquestación y creación de pipelines (flujo de datos) para la limpieza, agregación o modificación de datos y procesos usando como lenguaje de programación Python. Uno de los elementos más usados son los Gráfico Acíclico Dirigido (DAGs) que está conformado por Tasks, los cuales son Operators que terminan ejecutando o interactuando con los servicios y los datos (Shukla, 2022). En la Figura 1.5, se muestra un DAG junto con los Task que lo definen.

Figura 1.5

Estructura de un DAG



Como se aprecia en la Figura 1.5, un DAG es una combinación de nodos que ejecuta una acción, la complejidad de la elaboración corresponde al programador que lo diseñe, quien puede ordenar los Task de manera secuencial o en paralelo. En el ejemplo presentado, el Task es secuencial, ya que para ejecutar la siguiente tarea depende de solo una actividad. Asimismo, el DAG no deja crear actividades cíclicas, debido a que esto generaría un bucle infinito que ocasionaría un alto costo dependiendo de lo que esté realizando. Por ejemplo, Shukla (2022) nos explica un caso práctico trabajando con DataProc, en el que construye, ejecuta y elimina; si esta actividad se da en un bucle se estarían creando varias instancias generando el aumento en la facturación. Los Tasks están configurados para trabajar como Operators, que son códigos ya configurados que permiten la ejecución de algunas sentencias o la interacción con algún servicio; por ejemplo, PythonOperator para la ejecución de código Python o BigQueryOperator para la ejecución de consultas en BigQuery; como se puede ver con este último, Apache Airflow tiene Operators que permiten la interacción con los servicios de GCP facilitando el uso de los mismos.

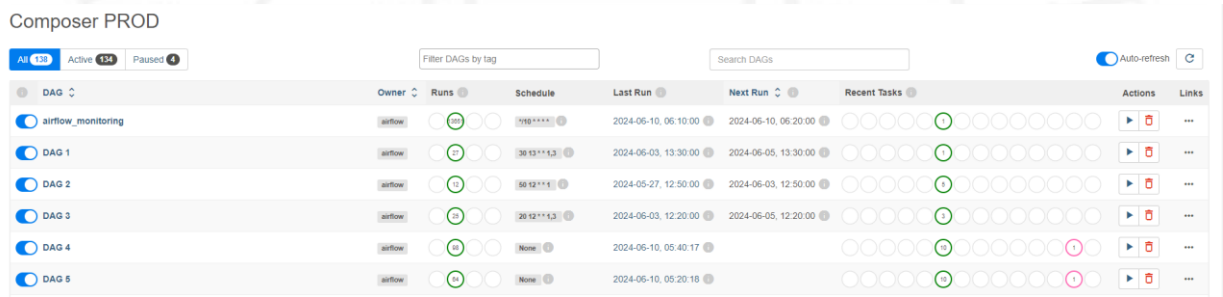
En la Figura 1.6, se tiene la interfaz de administración de Cloud Composer, en el que se puede visualizar un conjunto de DAGs, uno de ellos fue el presentado anteriormente en la Figura

1.5. Como se llega a revisar, el monitoreo es intuitivo y fácil de usar, ya que te presenta por medio de colores si un DAG o Task se han ejecutado de manera correcta. El color verde representa una actividad completada con éxito mientras que el color rojo indica un error en la ejecución; también se tienen otros estados como amarillo cuando va a reejecutarse o rosado cuando la actividad se omitió ya que no se cumplió con una condición. Gracias a esta interfaz, se puede hacer seguimiento de todo lo relación al lakehouse en un solo espacio.

Adicionalmente, el servicio cuenta con un sistema de alertas mediante correo, el cual se realiza configurando un servicio SMTP dentro del Cloud Composer. En los DAGs se puede indicar a quienes deberían mandar una alerta en el caso se presente un error, asimismo, otras de las configuraciones, es la opción de generar reintentos, los cuales son útiles cuando se tiene una comunicación con otros sistemas y la conexión se pierde por algunos instantes permitiendo retomar el Task sin perder lo trabajado. Finalmente, la ejecución de los DAGs se realiza mediante una programación a una hora en específico o por la dependencia de otro DAG.

Figura 1.6

Interfaz de composer



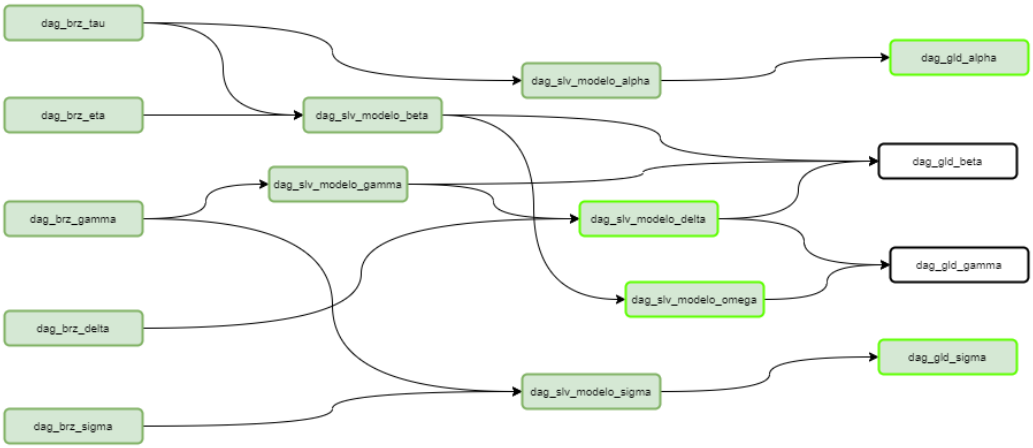
Cloud Composer se encarga de centralizar los DAGs, los cuales tienen la lógica para trabajar en cada una de las capas del lakehouse, pero queda un punto importante a compartir y es la forma de cómo estos DAG interactúan; se había comentado que a nivel de Task se podían indicar la forma de ejecución, pero esto no se da a nivel de DAGs de manera implícita; es por ello, que en la Figura 1.7 se presenta un diagrama donde se tiene la forma en el que estos interactúan.

Uno de los temas importantes para cualquier iniciativa está relacionada a los costos, ya que representa la inversión dada y si esta sale del presupuesto se tiene que medir si es conveniente seguir invirtiendo en ello. Por lo tanto, es importante poder optimizar la ejecución de los DAGs, ya que un reproceso de uno cuando no es necesario representa un gasto que se podría evitar. Con esta premisa, se creó dentro del área una configuración que permitía la ejecución de los DAGs, solo si la dependencia había terminado. Esta forma de trabajo está influenciada en lo explicado

con los Tasks pero aquí se está llevando a un nivel superior agrupando y optimizando la frecuencia de ejecución de los DAGs.

Figura 1.7

Ejecución de los DAGs



Como se aprecia en la Figura 1.7, cada conjunto de DAGs representa una parte de las capas del lakehouse, comenzando por Bronze, los cuales disparan el proceso al terminar la ingesta de las tablas. Cada DAG de Silver se suscribe a un DAG de Bronze con la finalidad de que cuando este termine continue, se puede suscribir a una o varias dependencias generando que el DAG solo se ejecute cuando tenga datos a actualizar. Este mismo flujo se da en las capas superiores.

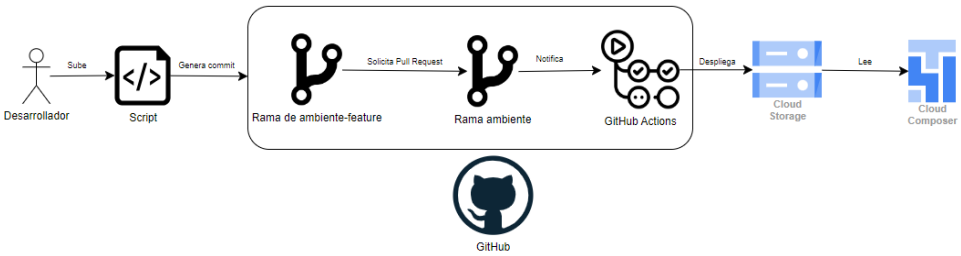
En el ejemplo presentado, se puede ver que la ejecución del dag_gld_beta aún no se realiza porque el dag_slv_modelo_delta aún está en ejecución, mientras que el dag_gld_alpha ya se está ejecutando porque su dependencia ya terminó. Esto permite que el proceso también responda ante eventualidades, ya que, si existe una demora, el DAG esperará que este termine para accionarse, y al mismo tiempo, disminuye los tiempos de espera (timeouts), ya que no se programa una hora en específico para accionar un DAG, sino que su ejecución es una dependencia de una actividad anterior. De este modo, se logra optimizar los procesos a nivel de tiempo y costo, ya que la ejecución se realiza solo cuando es necesario a la hora precisa.

El proceso comentado es la forma en cómo se ejecuta en los ambientes, pero el desarrollo no se hace directo en los servicios. En su lugar, se cuenta con un repositorio de código centralizado en Github, donde el desarrollador realiza su pase y tras la conformidad del Líder Técnico, término que se abordará con mayor amplitud en la siguiente sección, se inicia el proceso de despliegue. Ningún usuario tiene acceso a insertar directamente en las tablas de BigQuery o ejecutar una actividad del Composer, ya que todo el despliegue se realiza por medio de la integración continua

de GitHub Actions, el cual asegura la integridad de lo desarrollado en los repositorios para los ambientes. En la Figura 1.8 se grafica el proceso explicado.

Figura 1.8

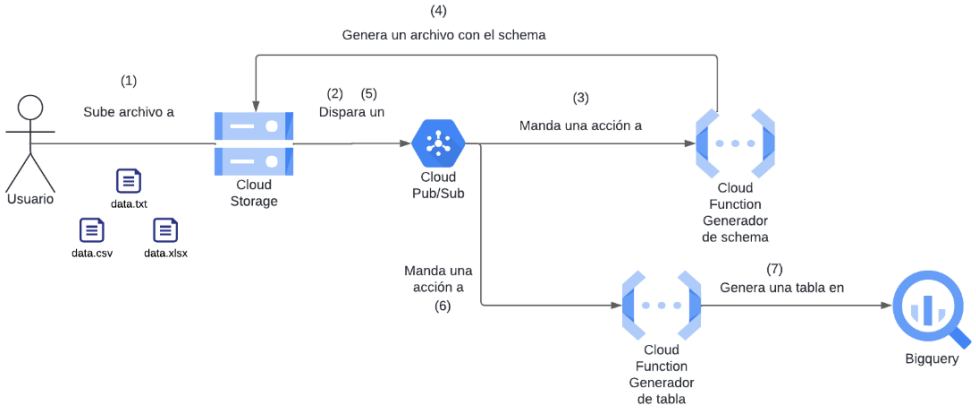
Proceso de despliegue en ambientes



Uno de los procesos que se había comentado con anterioridad, pero no se había profundizado era la ingesta de archivo de negocios o Data Entries. Para poder llevar a cabo esto se utilizaron varios servicios de GCP, donde destaca el uso de Cloud Function que contiene la lógica para la ingesta del archivo, Cloud Storage para el almacenamiento de los archivos, Cloud Pub/Sub como disparador del proceso y BigQuery como destino del archivo.

Figura 1.9

Proceso de Data Entries



En la Figura 1.9 se muestra el flujo para la subida de los archivos en la capa Bronze. El proceso inicia cuando con un usuario sube un archivo a Cloud Storage (1), este archivo debe presentar un formato .txt, .csv o .xlsx, se escogieron estas extensiones ya que son las más usadas por negocio para guardar sus datos. Desde la perspectiva del usuario el proceso es sencillo, ya que entra y deja el archivo. Esta acción desencadena que el Cloud Pub/Sub escuche la acción (2) y mande dos alertas a las Cloud Functions (3), la alerta es recibida por ambas, pero solo acciona una generando el archivo de configuración (4), que contiene el esquema de la tabla y algunos

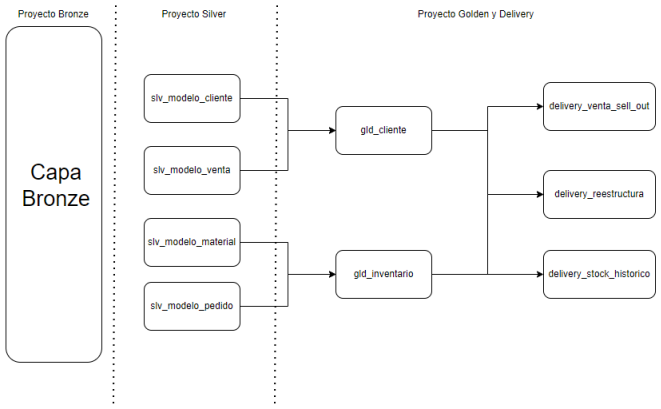
parámetros como la forma de ingesta (truncado o incremental) o el correo de las alertas. Tras la generación de este archivo el cual está en un formato JSON, este se guarda en el Cloud Storage para posteriormente mandar nuevamente una alerta (5), pero esta vez siendo recibida por la segunda Cloud Function (6), la cual utiliza el archivo de configuración para poder ingestar el archivo en BigQuery (7).

Este proceso demora segundos o a lo mucho unos minutos dependiendo del archivo subido. De esta manera, se obtienen todos los datos en la capa Bronze para complementar los datos obtenidos de los sistemas ERP. Además, este servicio también cuenta con un sistema de alertas por si el archivo está mal y es enviado a los usuarios que se registren en el archivo de configuración; esto con la finalidad de que sean notificados en todo momento sobre el proceso y accionen en el caso sea necesario una corrección.

El proceso realizado en las capas Silver, Golden y Delivery son similares donde destaca el uso del DAG de dependencias para ejecutar cada capa y su almacenamiento en BigQuery, estructurado en proyectos (capas) y datasets (modelos). En la Figura 1.10 se muestra la vista a nivel BigQuery, donde se presentan algunos nombres de datasets que contienen diferentes tablas relacionadas, 4 modelos de Silver, 2 modelos de Golden y 3 modelos de Delivery.

Figura 1.10

Ejemplo de datasets en BigQuery



Al tener los datos actualizados en Bronze se comienza a ejecutar los DAG de Silver que alimentan a los modelos, cada modelo es indiferente al sistema ERP extraído ya que se busca ordenar los datos para el uso de la compañía, es por ello, que se puede tener un modelo en silver, por ejemplo “slv_modelo_cliente”, que reciba datos de Odoo, SAP NetWeaver u otros ERPs. En este modelo se tienen tablas relacionadas al cliente como “cliente_documento”, “cliente_email”, “cliente_telefono” o “cliente”.

Con la información organizada y gobernada en la capa Silver se trabaja la capa Golden, donde se busca consolidar los datos en función de las necesidades de negocio. Uno de los primeros datasets creados fue “gld_cliente”, ya que la organización necesitaba contar con los datos de sus clientes para poder realizar un análisis y evaluar las estrategias aplicadas, en ella se cuenta con tablas como “cliente_detalle” o “cliente”, que cuenta con los datos transaccionales y el maestro de cliente, respectivamente. El principal propósito de esta estrategia es que los usuarios no tengan que estar revisando varias tablas para poder generar un análisis sino tener un consolidado de ello para simplificar su trabajo.

Por último, la capa Delivery se encarga de presentar un resumen de la de Golden aplicando algunos filtros o lógicas propia del área solicitante. Un ejemplo de ello es el dataset de “delivery_reestructura” donde se tiene la información a nivel de las distribuidoras filtrando algunos productos que no forman parte del análisis.

Esta arquitectura permitió al área poder afrontar las distintas iniciativas con solvencia, permitiendo un escalado y mejoras constantes en función de las necesidades que se presenten. Asimismo, permitió generar una oportunidad para el empoderamiento del negocio con sus datos, ya no limitándolos a recibir un reporte sino dándole las herramientas para que ellos exploren y conozcan los datos en la búsqueda de mejoras de sus procesos.

1.2 Convivencia del sistema ERP

Esta iniciativa fue una de las más importantes dentro del equipo de Big Data, ya que representó un punto de partida para todo lo que se está viendo actualmente. Asimismo, logró dar mayor visibilidad y presencia al negocio, ya que el equipo empezó a trabajar sus propios proyectos de manera independiente sin descuidar el rol de apoyo para los otros equipos como Data Science o Data Governance.

Este proyecto nació como una necesidad ante un cambio en el sistema de ventas Sell Out (venta al cliente final); debido a que la inclusión de un nuevo ERP cambiaba el proceso que se venía trabajando, inclusive dejaba en desuso lo desplegado hasta el momento. La importancia de que esta información esté disponible es crítica para la compañía, ya que todas las decisiones comerciales parten de ello por la necesidad de un sustento numérico.

En la Figura 1.11 se presenta el proceso AS-IS, el cual se venía trabajando durante los últimos años y consiste en la interacción de varios sistemas hasta que finalmente se muestra la información al negocio.

Figura 1.11

Proceso AS-IS

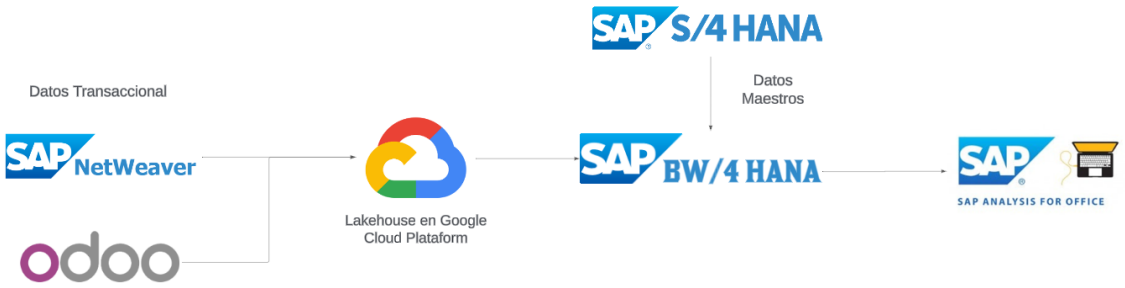
El proceso parte por los datos transaccionales generados en SAP NetWeaver, ya que este ERP es el que cuenta con toda la data generada por las ventas del día a día de las distribuidoras. En este sistema, se genera una tabla con las ventas consolidadas, que se envía a un sistema Oracle Data Warehouse. Este último se usa principalmente como nexo, ya que no se podía realizar una conexión directa con el SAP BW/4 HANA, según lo comentado por el equipo de TI. Después de pasar por el sistema de Oracle los datos llegan al SAP BW/4 HANA, donde se complementa los datos transaccionales con datos maestros obtenidos del sistema SAP S/4 HANA. Finalmente, esta información se presenta al negocio por medio de SAP Analysis for Office (AfO), un complemento de Microsoft Office que permite la visualización de los datos encontrados en SAP BW/4 HANA. El programa utilizado por la compañía para visualizar esta información es Excel y el proceso debe culminar antes de las 9.00 am.

Con el cambio de sistema ERP, era necesario una convivencia temporal; debido a que, las distribuidoras irán trasladándose al nuevo ERP mes a mes, siguiendo una estrategia de migración incremental. Se optó por esta estrategia en vez de un cambio total, debido a que un error en el nuevo sistema terminaría deteniendo las ventas de un día, lo que generaría pérdidas considerables para la compañía. Es por ello, que esto representaba un problema, ya que era necesario la coexistencia de ambos sistemas en el que los datos provenientes de cada ERP cuentan con una estructura de datos diferente a nivel de campos y tablas, ya que se cambiaron algunos conceptos o dejaron de utilizar durante la migración.

Esto generó que se buscarán alternativas para poder mitigarlo, ya que la salida en vivo ya estaba definida y no había opción a una modificación; es por ello, que el equipo de Big Data junto al equipo de TI estuvieron revisando opciones, dando como resultado el lakehouse. Esta solución permitía resolver todos estos problemas mencionados asegurando la disponibilidad y calidad de los datos.

Figura 1.12

Proceso TO-BE

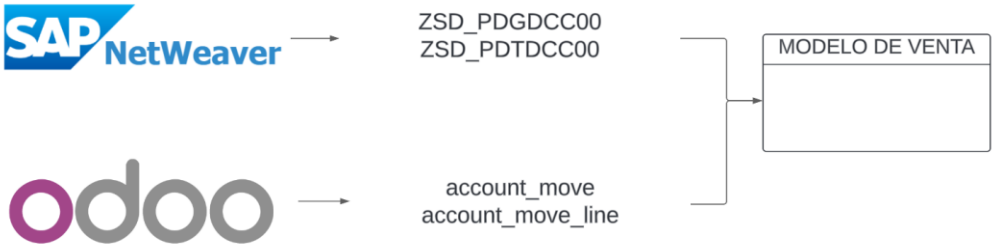


En la Figura 1.12 se tiene el nuevo proceso, el cual redefiniría la etapa inicial para que el resto del proceso no se vea afectado. Como se visualiza ahora se cuenta con 2 sistemas transaccionales, SAP Netweaver y Odoo, los cuales envían sus datos al lakehouse en Google para que pasen por las capas hasta que finalmente esté habilitado en la última de ellas y sea consumido por el SAP BW/4 HANA siguiendo el flujo anterior. Este nuevo proceso permitió la convivencia de ambos sistemas y al mismo tiempo resolvió uno de los problemas relacionados a la homologación de datos, ya que este podía ser resuelto dentro de las capas del lakehouse antes de ser enviados al SAP BW/4 HANA. Además, genera ahorros para la compañía, ya que el sistema Oracle Data Warehouse dejaba de ser utilizado, dado que su principal uso era el de pertenecer a este flujo. De esta manera, los datos ya no se encuentran en silos, sino que la información utilizada por el equipo de Data Science y negocio parten del lakehouse, generando una sola fuente de información en el que basar todas las decisiones comerciales de la compañía.

Como último punto a revisar se encuentra la homologación de datos, para poder entender ello en la Figura 1.13 se presenta un caso para unos de los datos más importantes del proceso.

Figura 1.13

Homologación de datos



Las tablas utilizadas por SAP NetWeaver y Odoo son distintas, no solo a nivel de nombre sino también de estructura, el caso presentado es de las tablas de documento de venta. En SAP

NetWeaver la cabecera del documento se encuentra en la tabla ZSD_PDGDCC00, mientras que en Odoo se encuentra en la account_move. En cuanto al detalle del documento, para SAP NetWeaver está en la ZSD_PDTDCC00 y en Odoo en la account_move_line. Adicional a ello, las columnas también son distintas, por ejemplo, la columna donde se encontraba un dato relacionado al cliente en SAP NetWeaver es el campo ZSD_CCLIEN, mientras que en Odoo es customer_id y para el material es similar, por un lado, es ZSD_CMATER y en el otro es product_id. De igual forma, estas columnas no están homologadas; en SAP NetWeaver se tiene el valor consultado, mientras que en Odoo al trabajar con un modelo relacional emplea identificadores correlativos. Por lo tanto, para obtener el mismo valor que en SAP NetWeaver es necesario ir a los maestros de las tablas. Para el cliente se tiene que ir a la res_partner para obtener el campo customer_code, y para el product_id se tiene que ir a la product_product para obtener el campo default_code.

Todo ello, conllevó mesas de trabajo con el equipo de TI para poder conocer esta información y poder plasmarla dentro de las capas, específicamente en la capa Silver, ya que es ahí donde se estandariza y homologa los conceptos en un modelado. En el ejemplo presentado, los datos relacionados al documento van en el modelo de venta. Este trabajo se realiza en esta capa para finalmente exponer una sola información a Gold y, posteriormente, en la capa Delivery, con el objetivo de que sea un proceso transparente para los usuarios y no tengan que estar consultando dos fuentes.

Esto permitió resolver uno de los grandes problemas que enfrentaba la compañía durante la migración de su sistema, ya que permitió la convivencia de la información en un sistema externo a los ERPs. De este modo, se logró consolidar los datos en un ambiente centralizado, accesible para los diferentes usuarios de la compañía, asegurando la integridad, calidad y disponibilidad de los datos.

1.3 Integración con aplicativos finales

Esta sección es consecuencia de las dos anteriores; debido a que, al contar con una arquitectura escalable se permite agregar nuevos módulos con nueva información; asimismo, el impacto del lakehouse permitió que el negocio tenga una nueva alternativa para generar sus análisis de manera automatizada. Dentro de las iniciativas que se han ido trabajando están:

- Conexión a Aplicativos Móviles
- Conexión a Aplicativos Web
- Reportería en Power BI

La compañía cuenta con varios productos digitales donde muestra información a su fuerza de venta, supervisores o personal administrativo. Uno de estos canales es el aplicativo móvil, que permite al vendedor realizar los pedidos de los clientes y, al mismo tiempo, sirve para darle información relevante sobre el cliente o su avance, con el propósito que conozca qué producto puede ofrecerle o compró el cliente, el comportamiento de compra, la frecuencia y otros indicadores que faciliten la venta. Asimismo, también cuenta con información sobre su desempeño y qué debe realizar para poder cumplir sus objetivos, como la cantidad de clientes, los productos que necesita posicionar en el mercado o la meta en soles. Esta información es relevante para el vendedor porque le permite ganar concursos e incentivos.

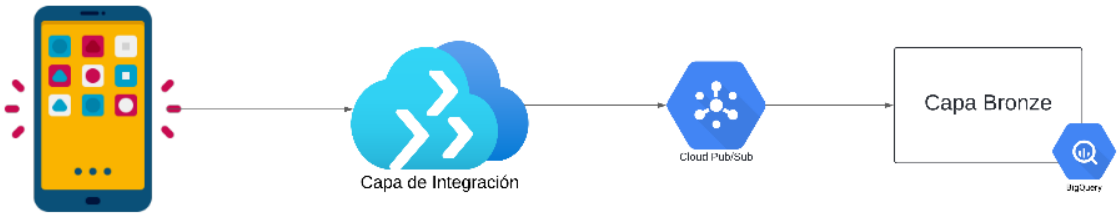
La participación del área consiste en la disponibilización de esta información. Para ello, se organizan mesas de trabajo donde intervienen distintos equipos, como Big Data, desarrolladores del aplicativo móvil, desarrolladores de APIs de integración y analistas de negocio. El objetivo es asegurar el correcto desarrollo y un canal de comunicación directo entre los participantes.

Esta iniciativa por el lado de Big Data se dividió en dos etapas, una de migración y otra de nuevos desarrollos, ya que existían servicios desarrollados en lenguaje ABAP para el ERP NetWeaver y necesitaban complementarlo con los datos del nuevo ERP de Odoo. Para ello, se tuvo que realizar una documentación sobre estos servicios apoyándonos de un programador ABAP. Tras tener los servicios migrados gracias a esta documentación se realizaron las mejoras solicitadas por los analistas de negocio, los cuales involucraron en añadir más información a lo ya trabajado.

La siguiente iniciativa involucra la disponibilización de información en otro producto digital, un aplicativo web, el cual sirve como herramienta para realizar un monitoreo a los vendedores y medir algunos indicadores, como número de visitas, tiempo en ruta entre otros. Para ello, el aplicativo móvil envía información a la capa Bronze del lakehouse y esta pasa por el proceso descrito anteriormente hacia la capa de Delivery, donde se disponibiliza para la plataforma donde los supervisores tienen acceso. Lo diferente de esta iniciativa al resto es la nueva fuente de información, ya que el lakehouse no solo cuenta con datos de un ERP, sino que ya ingresa datos de aplicativos que permite enriquecer los análisis y tener un repositorio más robusto.

Figura 1.14

Conexión con Aplicativo móvil

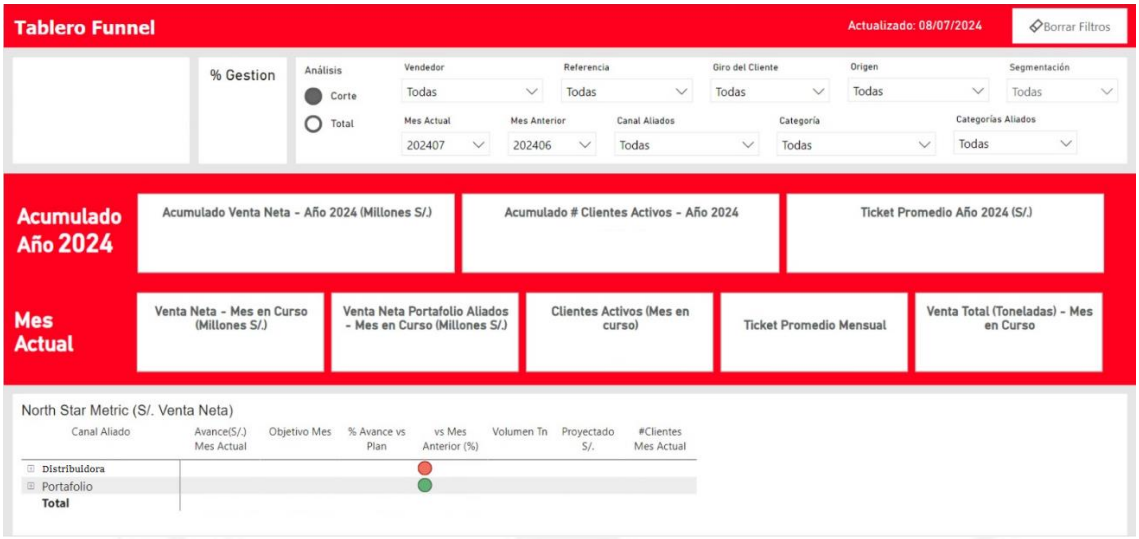


Como se aprecia en la Figura 1.14, la conexión con los productos digitales. tanto de ingesta como de consumo. no se realiza de manera directa; sino que, la compañía cuenta con una Capa de Integración que sirve de middleware para la disponibilización de APIs de consumo. La interacción con el equipo que administra este middleware es constante, dado al número de iniciativas en las que es necesaria la participación de data. Para esta iniciativa, se utilizó un Pub/Sub para poder recibir los datos enviados por la Capa de Integración, con la finalidad de que se puedan encolar las respuestas y no se pierdan los datos.

Finalmente, el lakehouse tiene otra forma de disponibilizar información y es a través de Power BI, los cuales se conectan a la capa de Delivery y extraen las tablas necesarias para la elaboración de los reportes. Este punto ha tenido una gran recepción por parte de negocio, ya que ha visto la posibilidad de que ellos mismos generen su reportería sin tener que recurrir a un proceso de extracción manual de datos. gracias a la automatización que actualiza los datos de manera diaria. Al mismo tiempo, generó una curiosidad en ellos por los datos disponibles en el lakehouse, para lo cual se les habilitó un entorno aislado con los datos disponibles para que ellos puedan trabajar y al ser sentencias SQL están, en su mayoría, familiarizados, a diferencia de una transacción SAP en un sistema ERP. Un ejemplo de ello se ve en la Figura 1.15, donde se muestra un dashboard desarrollado para el área de ventas con datos extraídos del lakehouse de la capa de Delivery. Su principal uso radica en hacer un seguimiento a la fuerza de venta para el cumplimiento de sus objetivos, pudiendo comparar frente a meses pasados. Debido a que la información de venta es sensible para la compañía, se presenta la estructura del dashboard diseñado sin los números obtenidos y solo las etiquetas de los indicadores.

Figura 1.15

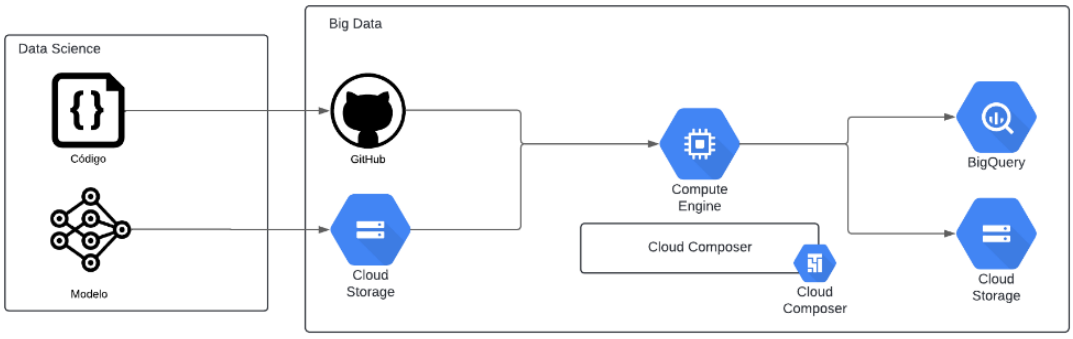
Ejemplo de dashboard conectado



Adicionalmente, a lo presentado con relación al lakehouse, también se trabajó durante un tiempo con el equipo de Data Science para la disponibilización y automatización de los modelos de Machine Learning. Para ello, el equipo de Data Science compartía un notebook con el modelo a implementar, la lógica para la generación de los datos de entrada y el formato de salida para el aplicativo a consumir. El notebook no estaba estandarizado para una automatización, es por ello, que este proceso involucraba seguir algunas prácticas de programación para su optimización, como encapsulamiento, herencia, abstracción y polimorfismo. Esto me llevó a conocer acerca de los modelos, no tan al detalle de su construcción, pero si el proceso realizado, así como la lógica trabajada con negocio.

Figura 1.16

Despliegue de modelos analíticos



Como se visualiza en la Figura 1.16, el proceso emplea varios servicios de Google, así como Github como repositorio de código. El proceso consiste en adecuar el notebook dado por el

equipo de Data Science a trabajar con Docker, debido a que el código debería ser ejecutado en cualquier ambiente sin recibir influencia del entorno instalado. Asimismo, el modelo entregado en un archivo pickle es subido a un Cloud Storage con el propósito de tenerlo en un entorno centralizado y de fácil acceso. Tras tener configurado ambos inputs, se procede a crear un DAG en Cloud Composer, el cual se encarga de la creación de una máquina virtual, de la ejecución del código importado de Github usando el modelo del Cloud Storage y de la eliminación de la máquina por el tema de costos. Antes de la eliminación, el proceso deja el output en un archivo en Cloud Storage para el consumo de los productos digitales, mientras que se sube el mismo contenido a BigQuery para que se pueda analizar y trabajar explotándolo con el resto de los datos que cuenta el lakehouse. Cabe resaltar que se realiza en los 3 ambientes: desarrollo, calidad y producción, con el propósito de evitar errores que afecten a los aplicativos finales.

2. CAPACIDAD DE GESTIÓN

En mi experiencia profesional, he podido participar de manera activa dentro de varios equipos interdisciplinarios, ya que cada iniciativa involucraba la participación de diversos perfiles para el cumplimiento del objetivo; por ejemplo, roles de Big Data, Data Governance, Data Science, Data Translator, Tecnología de la Información (TI), entre otros. Participé como parte del equipo de desarrollo, así como líder técnico por el lado de Big Data, donde he podido trabajar de la mano de un numeroso equipo.

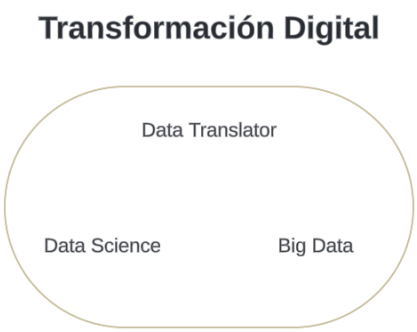
2.1 Participación dentro del área

Como se explicó al inicio, el área de Transformación Digital era relativamente nueva, es por ello, que el equipo no era numeroso, formado por aproximadamente 10 personas, y se podían tener reuniones con todos los participantes. Este equipo incluía roles de Data Science, Big Data y Data Translator; este último hacía de nexo entre el equipo interno y el negocio, facilitando la presentación de las iniciativas en las que trabajábamos y el valor agregado de ellas. Por su parte, los Data Scientist se encargaban de la elaboración de los modelos, y el equipo de Big Data se encargaba del despliegue y automatización. Esta forma de trabajar se dio durante mis inicios en el puesto, donde mi rol era más de desarrollador para el despliegue de las iniciativas y para resolver las dudas relacionadas a data tanto para iniciativas a nivel nacional como internacional.

Está forma de trabajar me permitió poder mejorar mi comunicación e interacción con un equipo no especialista en mi rama, y al mismo tiempo poder conocer más sobre otras capacidades dentro del área. Se obtuvieron logros como el desarrollo de tres grandes iniciativas de modelos analíticos que actualmente siguen vigentes, dos para el mercado nacional y uno para un mercado internacional. En la Figura 2.1 se presenta la forma de trabajar del área.

Figura 2.1

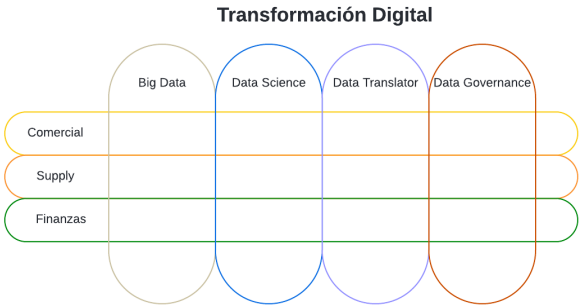
Organización inicial del área



Con el tiempo, el área fue creciendo y el modelo de trabajo ya no era muy eficaz, porque se tenían reuniones de más de una hora donde se tocaban varios temas que no necesariamente involucraban a todos los participantes. Esto se debió a la llegada de varias iniciativas de distintos frentes de negocio; comercial, supply y finanzas. Esta forma de dividir las iniciativas, así como el crecimiento del equipo originó que se reestructure la forma de trabajo buscando la optimización de los tiempos. En la Figura 2.2 se presenta la nueva forma de trabajar del área.

Figura 2.2

Organización cruzada del área



Como se visualiza en la Figura 2.2, ahora se tiene dos perspectivas, una relacionada al negocio (Comercial, Supply y Finanzas) y otra en función a la capacidad (Big Data, Data Science, Data Translator y Data Governance). Ahora se contaba con dos tipos de reuniones más cortas, una para abordar las iniciativas de negocio donde participaban todas las capacidades del frente y otra sobre la especialidad.

En esta nueva organización, me ubique como desarrollador en el equipo de Big Data del frente comercial, donde seguía trabajando en las automatizaciones y despliegues de las iniciativas, pero ahora tenía una nueva función explorando nuevas tecnologías para el área, las cuales serían la base para lo trabajado en lakehouse. Las reuniones de los frentes eran similares a lo trabajado anteriormente con reuniones semanales, y se sumaban las reuniones de capacidad de manera mensual donde se abordaban temas transversales y mejoras en los procesos que veníamos trabajando.

Con esta estructura se dio el proyecto de la convivencia de los ERPs para el frente comercial, el cual tomaba gran relevancia por lo descrito anteriormente. Gracias a ello, obtuve el rol de Líder Técnico en el frente comercial. Dentro de los equipos se podían encontrar varios roles como Líder de Frente, el cual se encargaba de la coordinación con otras capacidades, no solo del área sino de otras, como TI o negocio; el Líder Técnico, el cual tenía el rol de trabajar con el equipo de desarrollo las iniciativas levantadas, buscaba medir el rendimiento del equipo y el

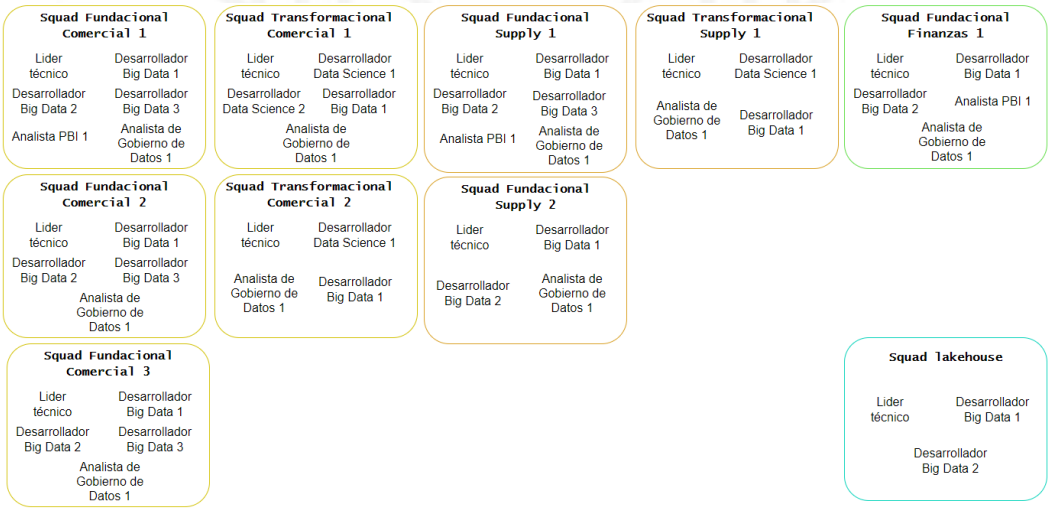
cumplimiento de los objetivos trazados, daba soporte en todo lo relacionado a data, así como la coordinación con sus pares de otras áreas para el levantamiento de consultas y pruebas de las iniciativas; y; finalmente, el equipo de desarrollo, los cuales se encargaban de trabajar las iniciativas y la arquitectura del lakehouse apoyados del Líder Técnico.

Mi equipo de desarrollo contaba con 8 personas para llevar a cabo el proyecto durante 9 meses. Debido al número de iniciativas que estábamos trabajando en paralelo se realizaban reuniones diarias de 30 minutos donde se presentaba una lista de actividades y a la persona asignada a cada una de ella. Esto nos permitía hacer un seguimiento efectivo y detectar si había un inconveniente para poder ayudarlo, asimismo servía para dar visibilidad al equipo interno de lo que se estaba trabajando. Esta forma de trabajo me permitió poder liderar un equipo y me ayudó a mejorar en mis habilidades de gestión, ya que empecé a trabajar con cronogramas y llevar reuniones con distintos equipos interdisciplinarios, conformados por distintas personas y perfiles.

Tras finalizar el proyecto en los tiempos propuestos y cumplir las expectativas, se llevó una nueva reorganización en el área, ya que al contar con el lakehouse se vendrían más iniciativas para los distintos frentes y ya no se contaría con un gran proyecto por frente, sino que se tendría varias iniciativas en paralelo, es por ello, que se reestructuró de la siguiente manera formando squads.

Figura 2.3

Organización por squads



Como se visualiza en la Figura 2.3, los frentes se dividieron en squads, debido al gran número de iniciativas por desarrollar y al mismo tiempo dentro de los frentes se vio una distinción entre las iniciativas que se están trabajando, unas más en función a disponibilización de data y otra

más en función de creación de un valor agregado, las cuales fueron denominadas funcional y transformacional, respectivamente.

El rol de Líder Técnico se mantiene igual, el cual consiste en poder llevar a cabo las iniciativas gestionando al equipo de desarrollo; sin embargo, el rol de Líder de Frente sufre un cambio, ya que se tiene uno por iniciativas fundacionales y otro por transformacionales, con el propósito de poder tener una persona con un perfil de acuerdo a lo que se busca en los proyectos, en las iniciativas fundacionales un perfil más de Big Data mientras que en el transformacional un perfil más cercano a negocio, como un Data Translator o Data Science.

Esta nueva forma de trabajar generó que el lakehouse vaya creciendo con mayor ritmo, ya que se tendrá varios equipos revisando varias iniciativas diferentes, lo cual aportaría en la generación de más información modelada y disponibilizada.

En esta nueva restructuración, continué con el rol de Líder Técnico del frente comercial, pero también ejercía de consultor de tema de data para las squads transformacionales, gracias a la experiencia de trabajar en el proyecto de migración. Además, lideré una squad del lakehouse, la cual se encargaba de llevar las mejoras del composer, la estandarización de los flujos de producción y la sinergia de procesos con el resto de LTs. De esta manera, tenía un rol más híbrido viendo temas relacionados a negocio con mi squad del frente comercial y las consultas del squad transformacional; y el aspecto más técnico y transversal con el squad del lakehouse. Esta nueva forma de trabajar me dio mayor responsabilidad, ya que interactuaba con el resto de los frentes para poder llevar a cabo una estandarización de los procesos, los cuales eran informados a ellos para que lo notifiquen a sus equipos.

2.2 Herramientas

Para poder llevar a cabo mi función de Líder Técnico, me apoyé de varias herramientas y metodologías de trabajo, ya que tenía un equipo bajo mi responsabilidad y varias iniciativas propias del squad y transversales con otros equipos. Es por ello, que se debe seguir una metodología de trabajo, la cual ha sido influenciada por las experiencias previas en anteriores equipos como desarrollador.

El enfoque en el desarrollo de un producto para un usuario se puede dividir en cinco etapas: planificación, análisis, diseño, implementación y mantenimiento (Nugroho et al., 2017). Estas etapas pueden ser abordadas de distintas formas según la metodología adoptada, como cascada, SCRUM o Kanban. Según Saleh et al., (2017), el software se debe trabajar de manera incremental e iterativa, debido a que se busca la participación continua de los usuarios y la mejora continua

del código desarrollado. Esta forma de trabajo lo da la metodología ágil como SCRUM o Kanban, mientras que la metodología de cascada al ser rígida dificulta su implementación y presenta grandes problemas relacionados al tiempo o costos. Hayat et al., (2019) coincide con lo comentado anteriormente, destacando que un buen producto de software se da por medio de una buena gestión que incluye conocimientos, herramientas y técnicas. Para ello indica que la metodología SCRUM es de las más usadas por su gestión diaria, revisión de tiempos, costos y alcance.

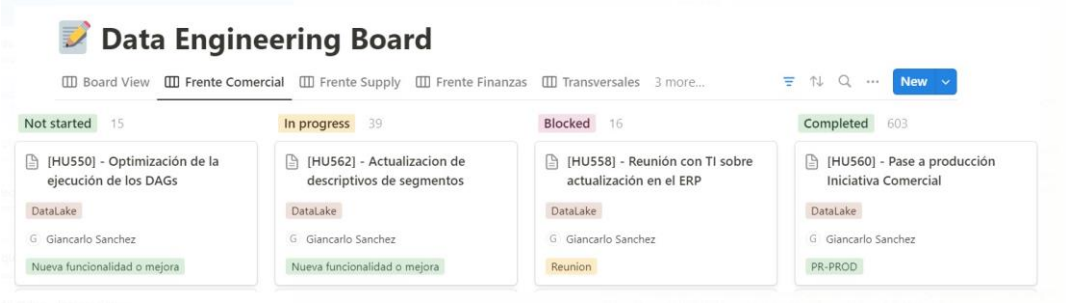
Entre los roles y herramientas mencionados por Adi (2015), Srivastava et al., (2017) y Hayat et al., (2019) adoptados para nuestra forma de trabajo están el Product Owner, Scrum Master y el Scrum Team; el primero recae en la persona del Líder de Frente, el cual define con negocio cuál es la especificación y lo esperado, así como nexo para la resolución de dudas; el Scrum Master recae en el Líder Técnico, el cual se encarga de gestionar al equipo consultando que han trabajado y que trabajarán, además de poder resolver cualquier obstáculo que dificulte el avance; por último, el Scrum Team, compuesto por los desarrolladores que se encargan de construir la solución. Además, también se adoptó los conceptos de Product Backlog, Daily Mettings y Sprint Retrospective; siendo el primero la lista de actividades o casos de usos asignados al equipo para poder cumplir con una iniciativa; el Daily Meeting son reuniones de 30 minutos en las que el equipo expone su avance y aborda cualquier inconveniente; finalmente, el Sprint Retrospective se da al final de la iniciativa, donde se comparten las experiencias y se busca mejorar.

La mayoría de los conceptos de la metodología SCRUM son adoptados en la forma de trabajar del día a día. Se lleva a cabo una reunión en Microsoft Teams de martes a viernes de 9:00 am a 9:30 am para llevar el Daily Meeting, donde se hace seguimiento al Backlog, trabajado en Notion, en la que se colocan las historias de usuario que se vienen trabajando y se comenta brevemente sobre ellas. En la Figura 2.4 se puede visualizar un ejemplo de las historias, donde se tienen cuatro secciones: no iniciado, en progreso, bloqueado y completado.

En “No iniciado” tenemos el Product Backlog, con todas las actividades mapeadas por el Líder de Frente junto al Líder Técnico. En “En progreso” se tiene las actividades que va trabajando el equipo, donde se busca que cada integrante del equipo no maneje más de tres historias de usuario en paralelo para que esté centrado en la finalización de ellos. “En bloqueado” se colocan las actividades que no se pueden avanzar por algún motivo, como una confirmación de usuario, una dependencia u otro; finalmente, “En completado” se colocan las historias que ya han terminado. Este enfoque de trabajo brinda visibilidad a todo el equipo y fomenta la generación de sinergias cuando se vea una oportunidad. Además, crea un espacio donde el equipo puede interactuar de manera efectiva.

Figura 2.4

Backlog de actividades



Estas actividades son parte del día a día del equipo de Big Data comercial, pero también se llevan reuniones con los equipos interdisciplinarios, donde se levantan las actividades que se van trabajando, así como si hay alguna dependencia, es un caso similar a lo que se presentó, pero su frecuencia es semanal. Asimismo, poder aceptar una iniciativa involucra un proceso de evaluación realizado por el Líder de Frente junto al Líder Técnico, donde se revisa la cantidad de participantes del equipo dentro de la iniciativa, el tiempo que se tomará en el proyecto y los riesgos que pueden presentarse. Esta evaluación previa permite tener un alcance claro de lo que se va a trabajar dentro de la iniciativa, asimismo sirve de entregable para que el equipo interdisciplinario tenga transparencia del proceso seguido. Un ejemplo de ello es un proyecto sobre una conexión con el aplicativo móvil para dar información sobre algunas métricas y el avance de los supervisores. En una primera reunión dada por el negocio a los equipos se comentó la necesidad y se envió la documentación del caso, esta documentación fue revisada junto al Líder de Frente y se hicieron algunas consultas al respecto, tras resolver las dudas se envió un cronograma a alto nivel, el cual fue anexado con los cronogramas de los otros equipos viendo la mejor oportunidad de optimizar tiempos (Figura 2.5), para finalmente tener una reunión entre todas las partes y dar por iniciado la iniciativa.

Figura 2.5

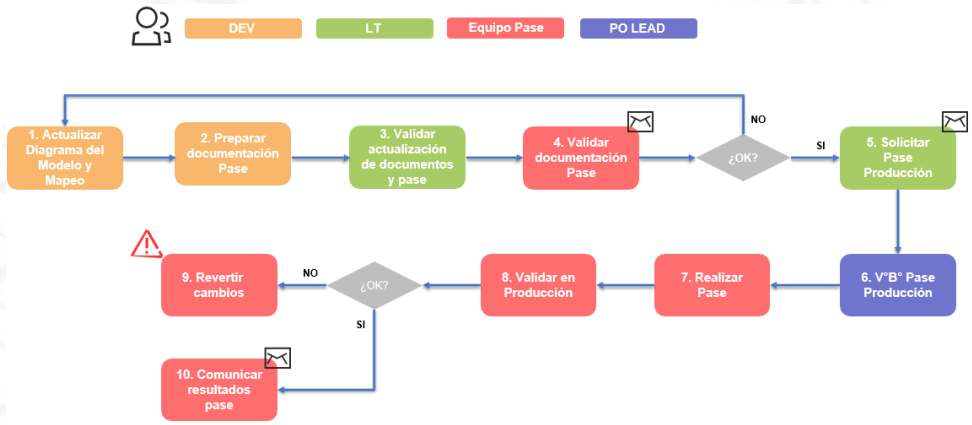
Cronograma de iniciativa

Tarea	Responsable	Ene-24				Feb-24			
		S1		S2		S3		S4	
		08-Ene	15-Ene	22-Ene	29-Ene	05-Feb	12-Feb	19-Feb	26-Feb
Seguimiento Diario									
SAS token, estructura, sql	Big Data								
Swagger, desarrollo, pruebas	Capa Integración								
Maquetación y Creación de servicios BE	App Móvil								
Integración y pruebas funcionales QA	App Móvil								
Pruebas de regresión QA	App Móvil								
Despligue a PRD	App Móvil								
FILTRO Clientes									
SAS token, estructura, sql	Big Data								
Swagger, desarrollo, pruebas	Capa Integración								
Maquetación y Creación de servicios BE	App Móvil								
Integración y pruebas funcionales QA	App Móvil								
Pruebas de regresión QA	App Móvil								
Despligue a PRD	App Móvil								

Lo comentado anteriormente está relacionado a lo trabajado para las iniciativas comerciales e interdisciplinarias; sin embargo, para las actividades del lakehouse tenemos otra forma de trabajar. En este caso, la gestión no es dada por iniciativa, sino por pase a producción. Para ello, la squad que desea solicitar un pase a producción debe cumplir con ciertos requisitos como: la aprobación del modelador, conformidad de las pruebas de calidad, la aprobación del Líder Técnico y la aprobación del Product Owner Lead, este último es un rol creado para centralizar los desarrollos de data y es asumido por el Big Data Head, con la finalidad de que las iniciativas publicadas en producción sean de su conocimiento. El flujo trabajado es el siguiente:

Figura 2.6

Flujo de producción



En la Figura 2.6 se presenta el flujo realizado para poder desplegar en el lakehouse. Este proceso ha ido madurando con el tiempo, y se tuvo que coordinar con todas las partes para poder llevar un control del mismo, siguiendo las mejoras prácticas para no tener inconvenientes con lo ya desplegado. El proceso se apoya en lo presentado en la Figura 1.8 de la sección de capacidad técnica, donde el desarrollador se encarga de generar el pase a producción y dejarlo listo para qué el LT se encargue de solicitar la conformidad del pase. Una vez aprobado, se procede con el despliegue. Este flujo fue elaborado como parte del desarrollo del lakehouse, ya que al trabajar con un gran número de equipos era necesaria la estandarización del proceso.

2.3 Iniciativas

Mi rol actual está relacionado también al seguimiento del equipo en el desarrollo y a poder llevar reuniones con otros equipos o usuarios de negocio para cumplir una iniciativa. Las herramientas usadas en la actividad diaria son el Notion para el seguimiento de las actividades, Microsoft Teams para las reuniones y el cronograma elaborado junto al Líder de Frente, que facilita la participación del equipo en términos de tiempos y responsabilidades. A continuación, se presentarán dos casos

donde se aplicó esta metodología; la convivencia del ERP y la disponibilización de datos para un aplicativo móvil.

Para la iniciativa de la convivencia del ERP, Netweaver y Odoo, se trabajó con 4 personas de mi squad y se llevaron reuniones interdisciplinarias con el equipo de TI y una consultora especialista en BW para la configuración y el despliegue en el sistema SAP. En estas reuniones participaba en representación del equipo y en función de alguna consulta puntual llamaba a alguien del equipo de desarrollo para que me apoye.

En la Tabla 2.1 se colocaron las actividades, el equipo responsable y el tiempo asignado, las cuales se hacían en su mayoría en cascada, debido a las dependencias de la actividad anterior. Asimismo, se tenía una reunión semanal entre los responsables de cada equipo; TI, la consultora y Big Data donde se abordaban avances, la gestión de permisos o consultas.

Tabla 2.1

Actividades de la iniciativa de convivencia de ERP

Sección	Actividad	Responsable	Tiempo
Elaboración de Diseño Técnico	Reunión de relevamiento y acuerdos	Consultora/Empresa	2 semanas
	Documentación	Consultora	2 semanas
	Aprobación	Consultora/Empresa	1 semana
Configuración en sistema SAP BW	Configuración de extractor	Consultora	2 semanas
	Configuración de Cadena de proceso	Consultora	1 semana
	Configuración de vistas	Consultora	1 semana
	Ajuste Composite	Consultora	1 semana
	Transporte a ambiente de calidad	Consultora	1 semana
Integración de Datos	Desarrollo y homologación de Avance de Venta en DEV	Empresa	2 semana
	Desarrollo y homologación de Venta Reestructurada en DEV	Empresa	2 semana
	Desarrollo y homologación de Avance de Venta en QA	Empresa	1 semana
	Desarrollo y homologación de Venta Reestructurada en QA	Empresa	1 semana
	Ejecución de Pruebas Integrales y Ajustes GCP y BW	Consultora/Empresa	4 semanas
	Pase a producción	Empresa	1 semana
Go Live	Documentación Técnica de la Solución y Entrega	Consultora/Empresa	2 semanas
	Soporte Post Producción	Consultora	1 semana

Esta iniciativa se finalizó según lo estimado y se desplegó en producción en el momento que se dio la convivencia de los ERPs, de esta manera los usuarios de negocio no tuvieron un impacto significativo en sus actividades. Se notificó sobre una marcha blanca y, actualmente, la información proviene del lakehouse. Durante la iniciativa. se trabajaron algunos indicadores de gestión por parte del responsable de TI, que englobaban las actividades llevadas por el equipo de Big Data y la consultora. En la Tabla 2.2 se presentan los indicadores llevados en las reuniones generales:

Tabla 2.2

Indicadores de gestión y control de la iniciativa de convivencia de ERP

Indicador	Control
Tiempo de la iniciativa	Declarado en el Gantt inicial permitía hacer un seguimiento de la ruta crítica para notificar a los otros equipos si la iniciativa pudiera atrasarse y tomar acciones para recuperar algún tiempo perdido.
Cantidad de desarrollos realizados	El objetivo era la convivencia de 2 macroreportes de la compañía, por ende, se hacía seguimiento a cada desarrollo para ver en qué situación se encontraba y si se cumplía con lo planificado.
Cantidad de pruebas realizadas	Para poder realizar el cambio era necesario realizar pruebas integrales para verificar que lo enviado por el lakehouse incluía el mismo valor que se enviaba por el proceso antiguo, para ello se trabajaron casos de uso los cuales fueron adjuntados en el acta final del proyecto.
Cantidad de usuarios afectados	Para la realización del cambio y poder medir el impacto se tuvo que coordinar con los distintos usuarios de negocio para notificarles y que sean partícipes de las reuniones donde se comunicaban las pruebas realizadas.
Calidad de la iniciativa	Se debía asegurar la disponibilidad correcta de la información presentando los números dados en los sistemas ERPs sobre la venta realizada hasta el día anterior.

Otra iniciativa trabajada es la disponibilización de datos para un aplicativo móvil en la que participé con 2 personas de mi squad para el tema del desarrollo. Al igual que el caso anterior, se trabajaron reuniones interdisciplinarias con otras áreas, como Integración de Aplicaciones y Desarrollo de aplicaciones móviles. El propósito de esas reuniones era revisar los avances y coordinar las pruebas, ya que la iniciativa involucraba mayor colaboración de los equipos. De manera similar a lo presentado en la iniciativa anterior, en la Tabla 2.3 se detallan las actividades trabajadas. La principal diferencia radica en que las secciones se podían avanzar en paralelo, debido a que son pantallas que se podían manejar de manera independiente permitiendo reducir los tiempos para la salida en vivo.

Tabla 2.3

Actividades de la iniciativa de disponibilización de datos para un aplicativo móvil

Sección	Actividad	Responsable	Tiempo
Prioridades de Cobertura	Disponibilización de la tabla	Big Data	3 semanas
	Configuración de conexión	Big Data	1 semana
	Desarrollo y generación de Swagger	Integración de Aplicaciones	3 semanas
	Actualización de servicio	Desarrollo de aplicaciones móviles	2 semanas
	Integración y pruebas funcionales	Desarrollo de aplicaciones móviles	2 semanas
	Pruebas de regresión en QA	Desarrollo de aplicaciones móviles	2 semanas
Cliente Efectivos	Disponibilización de la tabla	Big Data	3 semanas
	Configuración de conexión	Big Data	1 semana
	Generación de Swagger	Integración de Aplicaciones	3 semanas
	Actualización de servicio	Desarrollo de aplicaciones móviles	2 semanas
	Integración y pruebas funcionales	Desarrollo de aplicaciones móviles	2 semanas
	Pruebas de regresión en QA	Desarrollo de aplicaciones móviles	2 semanas
Venta	Disponibilización de la tabla	Big Data	3 semanas
	Configuración de conexión	Big Data	1 semana
	Generación de Swagger	Integración de Aplicaciones	3 semanas
	Actualización de servicio	Desarrollo de aplicaciones móviles	2 semanas
	Integración y pruebas funcionales	Desarrollo de aplicaciones móviles	2 semanas
	Pruebas de regresión en QA	Desarrollo de aplicaciones móviles	2 semanas
Despliegue a Producción	Coordinación entre los equipos	Todos	1 semana
	Pase a producción	Todos	1 semana

Esta iniciativa del lado de Big Data concluyó en las fechas previstas, pero se retrasó unas semanas debido a la repriorización de otras iniciativas en los otros equipos. Durante este tiempo, continuamos apoyando en las consultas relacionadas a data, lo cual no requería que una persona estuviera asignada a tiempo completo al proyecto. En la Tabla 2.4 se colocaron algunos indicadores revisados en las reuniones semanales, similar a la anterior iniciativa, pero enfocada en un nuevo desarrollo, ya que no se podía comparar con algo ya existente y se necesitaban realizar un mayor número de pruebas.

Tabla 2.4

Indicadores de gestión y control de la iniciativa de disponibilización de datos para un aplicativo móvil

Indicador	Control
Tiempo de la iniciativa	Consolida lo enviado por los 3 equipos donde se indica en que parte se tendrá mayor presencia o apoyo para la realización de algunas pruebas, servía para notificar sobre los avances y ver la respuesta de los equipos para finalizar según lo planificado.
Cantidad de pantallas realizadas	Permitía hacer un seguimiento sobre las pantallas finalizadas por cada etapa para que un nuevo equipo entre a continuar el trabajo, sirvió para conocer el estado del proyecto y medir los tiempos de respuesta.
Cantidad de pruebas realizadas	Permitía tener constancia de que los datos enviados por cada equipo cumplían con las expectativas y mostraba los valores reales para los usuarios
Facilidad de uso de los usuarios	Manejado por el equipo del aplicativo buscaba tener una alta usabilidad para que los usuarios se sientan a gusto y empleen la nueva característica de la herramienta.

Finalmente, manejé algunos indicadores generales del squad, los cuales son transversales a los proyectos y se utilizan diariamente para medir el rendimiento del equipo y las respuestas que brindamos ante el negocio u otros equipos. Estos indicadores se presentan en la Tabla 2.5.

Tabla 2.5

Indicadores de gestión y control del squad

Indicador	Control
Porcentaje de historias cumplidas vs proyectadas	Contrasta con lo llevado en las reuniones transversales (historias proyectadas) con el objetivo de revisar si lo planificado esta ajustado al rendimiento del equipo o se tiene que modificar.
Cantidad de historias cumplidas	Permite medir la cantidad de historias realizadas por cada integrante del equipo para evaluar su rendimiento y su capacidad para cumplirlo.
Cantidad de historias bloqueadas	Permite medir la cantidad de historias que están detenidas por una dependencia, con el fin de poder notificarlo en las reuniones para que cambien de estado, sirve para proyectar futuras iniciativas y tenerlo en cuenta dentro del cronograma.
Cantidad de historias modificadas	Permite conocer la cantidad de historias modificadas durante una iniciativa, principalmente por un ajuste dado por el responsable de la iniciativa, esto ayuda a poder repriorizar algunas historias y notificar sobre algún cambio en el cronograma.
Tiempo promedio invertido en cada historia	Permite conocer cuánto tiempo toma al equipo desarrollar cada historia, lo cual nos permite proyectar en futuras iniciativas y medir el avance de cada integrante durante el tiempo trabajo.
Cantidad de incidencias reportadas	Permite conocer la cantidad de incidencias presentadas en producción y poder realizar un análisis de cuáles son las más frecuentes para que el equipo este capacitado, asimismo detectar la ocurrencia y mitigarlo.
Tiempo promedio de respuesta frente a incidencias	Permite evaluar los tiempos de respuesta frente a una incidencia detectada y poder indicarles a los usuarios en cuanto tiempo se obtiene una solución.

3. APRENDIZAJE CONTINUO

Tras finalizar mis estudios de pregrado e ingresar a trabajar, me he interesado por la capacitación constante, debido a que la carrera y el rol que desempeño requiere siempre estar al día por el crecimiento y la aparición de nuevas tecnologías. En un principio me interesaron los cursos técnicos, pero la compañía siempre fomenta el lado de gestión de equipos y administración del tiempo propio con la finalidad de obtener un equilibrio del mismo.

Mi aprendizaje lo he trabajado de diferentes formas, ya sea por medio de cursos en línea en plataformas de autoaprendizaje, cursos de extensión dictados en universidades, capacitaciones dadas por la compañía, autocapacitación en temas de interés, investigaciones de alguna tecnología, la experiencia del rol y consulta a pares y expertos.

Cada uno de ellos ha tenido una gran influencia durante mi etapa profesional y me ha servido para poder aplicarlo en el trabajo. En esta sección se contará sobre la experiencia que me he llevado de cada modalidad sin un orden en particular, ya que las he estado aplicando en diferentes momentos en función del tiempo disponible y la necesidad de aprendizaje.

3.1 Cursos en línea

Esta modalidad es una de las más fáciles de acceder y de gestionar, debido a su modalidad remota permitiendo que la propia persona defina los tiempos que le pueda dedicar y es flexible, ya que se puede llevar en cualquier espacio sin tener la presión de perder una clase. Al tener una gran variedad de cursos en línea permite a la persona escoger cuales son los de su interés para llevarlo por un tiempo y ver si le aporta un valor adicional o desea ver otros cursos. Como plataforma use dos, Udemy y Coursera, siendo la última donde he llevado la mayoría de mis cursos.

Esta modalidad la he llevado durante dos etapas de mi vida laboral. La primera enfocada en el aprendizaje de las herramientas vistas en el trabajo, siendo el principal tema la nube de Google (Google Cloud Platform); ya que, como era nuevo en el área y no contaba con la experiencia de trabajar en un entorno GCP, me recomendaron llevar algunos cursos para familiarizarme con los servicios y entender las funcionalidades de cada uno. Esto me sirvió para relacionarlo con mis actividades diarias y encontrar oportunidades de mejora en los procesos que estaba trabajando.

En la segunda etapa, lo lleve como refuerzo a lo aprendido durante el trabajo. Ya había interactuado con varios servicios de Google, pero ligado a la experiencia y a la investigación que

realizaba, esto me ayudó a poder conectar la parte práctica con los conceptos aprendidos en los cursos permitiendo ver nuevas formas y tener un enfoque más general de lo revisado. Todos los conocimientos estuvieron relacionados a GCP, pero también a temas de código y buenas prácticas, como comandos en el terminal, técnicas de debugging, automatizaciones e interacción con git.

3.2 Cursos de extensión en universidades

Esta modalidad implica un compromiso mayor por parte de la persona, ya que se cuenta con un horario de clases donde se tiene que asistir y se cuenta con varias actividades con fechas límite. Por ello, estos cursos los he priorizado en épocas donde he contado con mayor tiempo libre, ya que son cursos de 3 meses a más los cuales se les tiene que dedicar varias horas. Además, fomenta la interacción entre los estudiantes permitiendo expandir el networking con personas de otras empresas y de diferentes perfiles.

Un curso que lleve hace poco es de Big Data & Analytics aplicado a negocio, el cual me ayudó a tener una perspectiva más general del área, ya que como comenté anteriormente se trabaja con equipos interdisciplinarios y esto me permitió conocer más acerca de modelos de predicción, redes neuronales, gestión de proyecto y desarrollo en programas de visualización como Power BI. En estos cursos he podido trabajar con personas que no ven temas relacionados a Big Data, inclusive con personas ajenas a la carrera de Ingeniería de Sistemas, pero con una afinidad a este, gracias a ello y a la metodología de los cursos he podido ver y aprender acerca de otros temas y otras perspectivas que no tenía en mente inicialmente; asimismo, participaba activamente dentro del equipo del curso, aportando ideas a partir de los conocimientos adquiridos en el trabajo.

3.3 Capacitaciones dadas por la compañía

Esta modalidad es impulsada por la propia compañía, ya que genera cursos de manera bimensual para todos los colaboradores y, al mismo tiempo, realiza una supervisión para poder medir si la persona va a culminar en el tiempo indicado. Los cursos son breves, no más de 5 a 6 horas y se dictan a través de plataformas de cursos en línea. No están orientados a la especialización sino a dar un concepto general o alguna pauta de gestión para aplicarlo en el día a día.

Durante mi tiempo en la compañía, me he inscrito en cuatro cursos, dos técnicos y dos de gestión. He intentado unirme a otros en algún tiempo, pero se cuenta con vacantes limitadas y. a veces. no he llegado a ellas. Los cursos técnicos eran de dos herramientas muy conocidas, Excel y Power BI, ya que me interesaba reforzar los conocimientos que tenía y poder ver alguna nueva forma de trabajar o consejo que no he estado aplicando. Por otro lado, los cursos de gestión si me ayudaron en mis funciones, ya que estoy en constante interacción con negocio y con otros equipos

por lo que es necesario mantener una comunicación asertiva, así como también conocer algunas pautas para llevar una presentación o reunión, las cuales me sirvieron para tener nuevas herramientas y conocimientos para desenvolverme.

3.4 Autocapacitación en temas de interés e investigación

Esta modalidad la he aplicado mucho, ya que inicialmente la usaba para la comprensión de ciertas incidencias dentro del trabajo, así como para implementar mejoras en los procesos buscando una mayor automatización y simplificación. Actualmente, cuento con un equipo para la revisión de mejoras que se puedan aplicar al lakehouse, con el fin de contar con una arquitectura robusta y de fácil uso.

Al comienzo revisaba mucha documentación técnica sobre los servicios desplegados por el área y, alguna vez, he tenido que investigar por algún incidente dado en los procesos. Por ejemplo, hubo una actualización de una librería en GCP que generó incompatibilidad con un servicio que estábamos utilizando, así que estuve revisando la documentación técnica para poder entender qué es lo que sucedía y buscar una alternativa para poder corregirlo.

Últimamente, el esfuerzo tanto de mi lado como del equipo es revisar y aplicar mejoras en el lakehouse. Para ello, se realiza una lluvia de ideas entre lo comentado por los LTs y algunos aspectos que vemos que puedan mejorarse, se selecciona una idea y se comienza a revisar la viabilidad de la propuesta, a partir de ahí se revisa y si esta todo conforme se trabaja en una prueba de concepto para finalmente ver cómo se integraría, si todo sale correcto se termina desplegando y notificando.

Entre los tópicos revisados están el uso de Dockers para las automatizaciones, lo cual me ayudó a poder entender y mejorar algunos procesos, como trabajar con escenarios particulares donde se necesitaba el uso de librerías de diferente versión para un mismo pipeline. También exploré el empleo de Cloud Composer y la integración con otros servicios de Google, inicialmente, los procesos no estaban centralizados y solo eran programados en sus servicios. Por ello, el área propuso revisar Cloud Composer para ver su funcionamiento y como ayudaría a reducir este problema; revisé configuración, uso de variables, DAGs, Operators y la integración con otros servicios como Compute Engine, Cloud Function, BigQuery entre otros.

También revisé acerca de la versión 2 cuando se realizó una actualización y el manejo de las librerías en cada versión de Airflow. Asimismo, exploré el uso de servicios SMTP para la notificación de alertas en Cloud Composer, debido a una necesidad de simplificar la configuración de notificaciones, ya que se tenía que estar instanciando en cada proyecto, ahora la configuración

se hace en el mismo Cloud Composer y manda un mensaje con el error encontrado empleando una plantilla personalizada.

Se investigó acerca del despliegue continuo empleando Git Actions, ya que los proyectos necesitan estar versionados y se buscaba simplificar el despliegue para evitar la interacción de los usuarios con los ambientes de calidad y producción, es por ello, que utilizando un servicio dado por GitHub nos conectamos a GCP para desplegar el servicio al escribir en las ramas de calidad o producción. Además, revisé los permisos en IAM, lo que me permitió solicitar los permisos de menor nivel para el equipo o un usuario de negocio, así como conocer las opciones y reglas dadas en GCP para administrar a los usuarios.

Se implementó la aplicación de enmascaramiento, debido a que se buscaba tener un mayor control de los permisos y del gasto de cada iniciativa o servicio. Era necesario poder identificarlo de manera rápida cada uno, es por ello, que se crearon cuentas de servicio para cada iniciativa que el Cloud Composer lo utilizaría empleando el enmascaramiento, ya que no debería usar su propia cuenta de servicio sino una propia de la iniciativa. Estos son algunos ejemplos de las investigaciones realizadas, los cuales nacieron de una necesidad o de una curiosidad para mejorar los procesos.

3.5 Experiencia en el rol

Esta modalidad forma parte del día a día y está muy relacionada al punto anterior por el lado técnico. Por el lado de gestión, he aprendido de la interacción con otros equipos y de las experiencias previas como desarrollador. Esto me ha permitido tener una mayor comprensión acerca de la forma de trabajar, poder aprovechar el tiempo de las reuniones y dejar claro el alcance y los acuerdos. Esta comprensión es fundamental para que el squad rinda de manera efectiva y cumpla los objetivos trazados, siempre fomentando la participación y el aporte de cada integrante del equipo. Además, las reuniones a final de cada iniciativa me han servido para conocer lo que hemos hecho bien para seguir aplicándolo y qué cosas pueden mejorar para nuevas iniciativas.

3.6 Consulta a pares y expertos

Esta modalidad se da también en el día a día del trabajo, ya que la interacción con diversas personas de distintos equipos y las personas del propio equipo ayudan en mi crecimiento profesional. Además de ello, los consejos dados por mis jefes me ayudan a poder ver otra perspectiva más general y poder aprender de ello.

En la compañía, la mayor parte de las personas está dispuesta ayudar y enseñar si cuentan con el tiempo para ello, lo cual lo agradezco, ya que permite acelerar algunas etapas de

aprendizaje. Por ejemplo, para la comprensión de cómo funciona el ERP de NetWeaver y Odoo, el equipo de TI fue fundamental, explicándome cada vez que tenía una duda sobre el funcionamiento o los datos que contenía. Del mismo modo, el equipo de Data Science también me ayudó con la comprensión de los modelos, así como de su forma de trabajar. Finalmente, las reuniones mensuales de uno a uno llevadas por mis jefes me ayudan a poder ver una perspectiva diferente a la que tengo, ya que ellos ven oportunidades de mejora tanto a nivel técnico como de gestión que puedo aplicar. y la misma forma de trabajar intento replicarla con el equipo cada cierto tiempo.

A continuación, en la Tabla 3.1 presentó los cursos que he finalizado durante mi etapa laboral, así como la fecha en los que los llevé y una breve descripción.

Tabla 3.1

Cursos de aprendizaje

Curso	Fecha	Descripción
Google Cloud Fundamentals: Core	Febrero a marzo 2022	Conocer acerca de Google Cloud Plataform (GCP), su infraestructura y servicios que disponibiliza.
Essential Google Cloud Infraestructure: Foundation	Marzo 2022	Interactuar con la consola de Google, redes VPC y Compute Engine.
Essential Google Cloud Infraestructure: Core Service	Abril 2022	Conocer y utilizar servicios de Google como IAM, Cloud Storage, Cloud SQL y herramientas de monitoreo de recursos y facturación.
Crash Course on Python	Agosto a septiembre 2022	Desarrollar soluciones automatizadas en Python siguiendo buenas prácticas.
Using Python to Interact with the Operating System	Septiembre 2022	Poder desarrollar programas que interactúen con el sistema operativo de Linux, comandos bash y el uso de expresiones regulares.
Introduction to Git and Github	Septiembre a octubre 2022	Comprender acerca del versionamiento de código e interactuar con comandos git y Github.
Troubleshooting and Debugging Techniques	Octubre a noviembre 2022	Emplear técnicas para encontrar y resolver problemas de código o infraestructura.
Configuration Management and the Cloud	Noviembre 2022	Configurar, desarrollar y desplegar un proyecto en la nube empleando servicios como Compute Engine.
Automating Real-World Tasks with Python	Diciembre 2022	Emplear la serialización de datos para el envío de mensajes entre programas usando librerías de Python.
¿Cómo desarrollar una presentación efectiva?	Febrero 2023	Elaborar una presentación eficaz y óptima para una audiencia general.
Fundamentos del storytelling para la presentación de data	Diciembre 2023	Dar a conocer algunas pautas para llevar una reunión relacionada a data con el fin de obtener un alto impacto en el negocio.
Big Data Analytics aplicada a los Negocios	Marzo a junio 2024	Identificar oportunidades basadas en datos empleando herramientas de visualización y aplicación de técnicas de Machine Learning en distintos tipos de datos.

4. CONDUCTA ÉTICA

Como profesional en la carrera de Ingeniería de Sistemas, es importante seguir lineamientos y pautas generales para el uso de cualquier información conocida en la compañía, siguiendo los criterios de transparencia, integridad y respeto. Además, es esencial tener presente el código ético y conducta profesional en cualquier momento e iniciativa trabajada.

Mi rol dentro de la compañía siempre ha estado relacionado a temas de data, por lo que cumplir con estos criterios es fundamental e importante en mi labor. He trabajado con data transaccional, de clientes y trabajadores; la cual he utilizado solo con fines laborales debido a los principios éticos y profesionales inculcados. Estos datos no están disponibles para todas las áreas, sino que se tiene que solicitar un permiso al equipo de TI y al responsable del dato definido por la compañía. Una vez que se obtiene la aprobación de ambas partes, se le da accesos al usuario de conexión para su disponibilización en la capa Bronze del lakehouse. Por lo tanto, no solo basta con contar con la conexión sino también tener los accesos disponibles para la descarga.

Esta solicitud es un proceso que demora un tiempo, aproximadamente 4 semanas, ya que se busca la seguridad y el buen uso de los datos. Los responsables asignados son los mismos usuarios de negocio elegidos por la compañía, los cuales ocupan altos cargos dentro de sus áreas y son datos que generan o administran. Por ejemplo, para obtener datos relacionados al stock de los productos, es necesario solicitar los permisos al área de planeamiento comercial. De manera similar se trabaja con las áreas de supply y finanzas. Esta forma de trabajar es definida y administrada por el equipo de TI y ciberseguridad, siendo nuestro rol el de usuarios ya que mandamos solicitudes para el acceso.

Del mismo modo, el lakehouse también ejecuta los lineamientos de la compañía por medio de las políticas de accesos para los usuarios de negocio que quieren consultar cierta data, así como el propio equipo interno de desarrollo a través de roles. Estos son importantes para la administración de los permisos, ya que nos permite crear perfiles que consoliden y faciliten la gestión para un grupo de usuarios similares. Esto simplifica los procesos de evaluación y el retiro de permisos o usuarios asignados.

Los permisos son trabajados a nivel de frente y de tipo de dato a consumir. A partir de ello, los usuarios de negocio tienen que solicitar un permiso a las áreas responsables para poder visualizarlo. Por el lado del equipo de desarrollo, los permisos son responsabilidad del Líder Técnico, el cual solicita al Big Data Head permisos para que su equipo pueda visualizar los datos

por un periodo de tiempo. La asignación de permisos no se realiza a un usuario particular, sino mediante los roles comentados, algunos ejemplos de roles son: desarrollador de Big Data del equipo comercial, desarrolladores de Power BI del equipo comercial, líder técnico del equipo comercial, líder de frente del equipo comercial, usuario de negocio del área de venta, usuario de negocio del área de planeamiento, entre otros.

En mi rol como Líder Técnico del frente comercial, asumí esta responsabilidad para el equipo de desarrollo y de responder algunas consultas dadas por los usuarios de negocio sobre sus accesos. Al conocer las iniciativas trabajadas por el equipo, pude solicitar los permisos necesarios para facilitar el desarrollo, y al mismo tiempo, concientizar sobre los lineamientos y responsabilidades que los Data Engineers tienen en el uso de los datos.

El rol del desarrollador comercial está en constante evaluación; según la política de la compañía, cada tres meses se revisa quienes son los usuarios asignados a cada rol; y si alguno se le da de baja, ya sea porque fue reasignado a otro frente o salió de la organización. Además, los permisos asignados se revisan cada 6 meses para verificar si estos deben ser renovados o retirados, según la necesidad de las iniciativas.

Según lo comentado por ACM (n.d.), el código de ética y conducta profesional puede ser abordado en varias secciones, las cuales serán comentadas a continuación, relacionándolo con mi experiencia laboral.

4.1 Principios éticos generales

Estos principios están relacionados con el actuar del profesional a nivel sociedad y compañía, respetando y protegiendo los derechos para evitar alguna consecuencia negativa o daño que afecte a la seguridad, confidencialidad o privacidad de alguien. Asimismo, se debe actuar con honestidad, confiabilidad y justicia, respetando las ideas y el trabajo de todas las personas.

Estos principios los he cumplido siendo transparente en mi actuar y enfocado en las iniciativas de la compañía, siguiendo las políticas y procedimientos instaurados por las áreas de ciberseguridad o las creada en el lakehouse. Por ejemplo, una política que es un poco tediosa pero necesaria es el uso del aplicativo móvil para la autenticación cada vez que se accede al correo. El equipo de ciberseguridad se aseguró que todos los usuarios lo tengan, y cuando tuve un inconveniente, pedí su apoyo para regularizarlo.

Asimismo, las políticas aplicadas en el lakehouse indican que los desarrolladores deben tener solo los accesos a los datos de manera temporal y en un ambiente controlado. El respeto

hacia el trabajo del equipo y lo realizado por otras áreas es fundamental, ya que involucró su tiempo y dedicación para la obtención de un logro que debe ser reconocido sin buscar adueñarse de ello. Por último, el acceso a los datos no pueden tenerlo todos los usuarios de la compañía, ya que existen personas a cargo que son los responsables de velar por el buen uso y la confidencialidad del mismo, es por ello, que para acceder a estos datos se tiene que solicitar un permiso donde se explica la razón por la cual desea acceder, y si se da la conformidad, se le habilita los permisos.

4.2 Responsabilidades profesionales

Las responsabilidades profesionales son un pilar importante para el desarrollo de cualquier iniciativa, ya que se debe estar en la búsqueda de una alta calidad en cualquier trabajo, siguiendo los estándares y reglas indicados por la compañía. Asimismo, es fundamental estar constantemente revisando y evaluando los desarrollos y capacidades propias, considerando siempre el respeto a los recursos informáticos y fomentando su buen uso.

Con relación a mis responsabilidades profesionales, siempre se está retando lo aplicado en el lakehouse, no solo a nivel de funcionalidades técnicas, sino también al proceso, buscando la mejora en la aplicación de estándares de la industria. Cualquier desarrollo que hemos trabajado como área ha aplicado los lineamientos de la compañía, respetándolos y siguiendo cada pauta indicada, como la solicitud de permisos a las personas responsables o creando roles para el acceso de los datos.

Otro punto para considerar es que el lakehouse también se somete a un proceso de auditoría, a nivel técnico y de proceso. Durante estas auditorias, se han observado algunas mejoras que se podrían aplicar, las cuales se les asigna una fecha límite para subsanarlo. Por el lado de procesos también recibimos una retroalimentación de este equipo especializado, el cual nos indica las buenas prácticas que han aplicado a proyectos similares.

Adicional a ello, cuento con varios accesos de visualización a los sistemas ERP por temas de validación de data, los cuales los utilizó de manera responsable y consciente, siempre con autorización de los responsables y siguiendo los lineamientos de la compañía. A nivel técnico, se ha configurado la gestión de permisos a través de roles, configuración de alertas y algunas herramientas de monitoreo centralizado. Gracias al convenio que tiene la compañía con Google, hemos recibido capacitaciones sobre algunas herramientas que nos ayudarían a tener un mayor control de la data trabajada. Por ejemplo, recientemente se estuvo revisando la incorporación de una herramienta de detección de data sensible para su enmascaramiento en todos los niveles. Esto

ayudará a tener todo un análisis exhaustivo, alineado con el equipo de Data Governance, para seguir las buenas prácticas.

4.3 Principios de liderazgo profesional

La aplicación de los principios de liderazgo profesional puede ser asumida por cualquier persona dentro del área, teniendo mayor incidencia en aquellos que tienen a su cargo un equipo. Dentro de sus funciones esta la labor de fomentar la responsabilidad social, aplicar políticas y procesos que aseguren el buen uso de los sistemas, así como buscar mejores oportunidades y calidad para su equipo.

Al asumir el rol como Líder Técnico, no solo me preocupo por el cumplimiento de las iniciativas, sino también de velar por el crecimiento profesional del equipo. Es fundamental tener en cuenta a cada uno de ellos y poder mantener una comunicación de cordialidad y respeto, donde se tenga un ambiente laboral en el que se sientan cómodos y vean la oportunidad de crecer. Al realizar trabajo remoto con la mayoría de ellos la comunicación directa se puede complicar, ya que no se tiene a la persona en la oficina, pero intentamos reducir esta distancia con reuniones cortas sobre temas puntuales y comprendiendo que al estar en su domicilio puede presentarse algún contratiempo que cause demora en ciertos momentos.

Del mismo modo, la interacción con otros equipos es clara notificando cuando se realiza una modificación o una actualización en el lakehouse que represente un tiempo de inoperatividad de los procesos, así como informando cuando se presenta una incidencia que está siendo atendida. Asimismo, los procesos heredados que están siendo absorbidas por el lakehouse se han asumido con responsabilidad ética explicándole a los usuarios como serán incluidos dentro de esta arquitectura. Esto simplifica varios procesos que realizan, pero se necesita su apoyo para ejecutarlo; de esta manera, se le da una visibilidad y transparencia para ayudarlos en este cambio.

La relación con proveedores también ha estado presente en mis funciones, tanto con consultoras especialistas en SAP como algunos equipos ajenos a la compañía para la creación de pruebas de conceptos o capacitaciones. El trato con ellos se basa en el respeto y el cumplimiento de los lineamientos de la compañía con relación a los datos compartidos y a los accesos dados. He trabajado con dos consultoras donde no he presentado inconvenientes, logrando trabajar eficazmente en la búsqueda de cumplir con las iniciativas.

Lo comentado anteriormente se aplica a todas las iniciativas que he trabajado en la compañía; por ejemplo, en los dos proyectos previamente mencionados, la iniciativa de la convivencia del ERP y la disponibilización de datos para un aplicativo móvil. Para la primera se

trabajó con una consultora especializada en el desarrollo de módulos en SAP BW, la cual necesitaba acceso a producción para poder disponibilizar la solución, por ello se les solicitó la firma de documentos de confidencialidad, así como la generación de usuarios para que puedan acceder solo a la parte de los sistemas que iban a modificar, esta coordinación y gestión lo realizó el equipo de TI, mientras mi participación se dio con el equipo interno encargado de la construcción del desarrollo en el lakehouse, la cual ya fue comentada con anterioridad en el tema de accesos y gestión; siempre se mantuvo una transparencia sobre lo desarrollado al negocio para que tenga visibilidad de lo trabajado mediante reuniones de coordinación y mesas de validación.

Por otro lado, para la segunda iniciativa no se trabajó con un equipo externo a la compañía, sino con otras áreas, las cuales necesitaban acceder al lakehouse para poder hacer las validaciones, la solicitud y las coordinaciones para sus accesos lo gestioné con el equipo encargado de los permisos y el Líder de Frente, siguiendo los lineamientos y estándares.

5. LECCIONES APRENDIDAS

El poder trabajar y participar activamente en la creación de una infraestructura de data desde cero me ayudó a conocer el proceso de construcción y de decisión, así como las buenas prácticas y sugerencias dadas por mis jefes que ya habían trabajado en otras empresas con una arquitectura similar. Esto me dio un punto de vista distinto y enriquecedor, ya que no trabajo con una solución ya desplegada y robusta, sino que he sido participe de la idea inicial que fue mejorando con el tiempo hasta la solución que actualmente está vigente.

Asimismo, la interacción con todas las áreas de la compañía enriqueció mi criterio y mi forma de trabajar, lo que me ha permitido apoyar y liderar las iniciativas de data de manera eficaz. Dentro de los aprendizajes obtenido puedo destacar lo siguiente:

- Una arquitectura de datos está en constante evolución, siempre salen nuevas necesidades ya sea internas o del negocio, lo cual nos permite enriquecer la solución trabajada.
- La tecnología cambia con el tiempo, siempre es bueno mantenerse al día con las mejoras o actualizaciones que lanzan, en mi caso al trabajar con la nube de Google revisó varias soluciones que están en vista previa y gracias a la compañía he podido tener capacitaciones con Google de nuevas herramientas que están trabajando.
- Es fundamental orientar el desarrollo a los usuarios finales, debido a que ellos son los que lo usarán en sus actividades, puede ser el proyecto más innovador, pero si los usuarios no lo ven necesario es difícil que prosperé, cuando se presenta una necesidad es una buena oportunidad para captar al público.
- Las reuniones con personas que no son de tecnología pueden ser complicadas si no estás preparado, ya que utilizar muchas palabras técnicas causará confusión y evitará que quieran participar, mientras más simple y clara la explicación mejor.
- Poder trabajar con un equipo me ayudó a entender y distribuir bien el tiempo, al final cada persona es distinta y tiene sus puntos fuertes como débiles, siempre es bueno conocerlos para poder ayudarlos en su crecimiento, lo cual termina en el beneficio del equipo.
- La metodología ágil es la tendencia para el desarrollo de las iniciativas, conservando algunas herramientas de otras metodologías como el uso del Gantt, tener una sinergia y adecuarse al equipo y a la necesidad es crucial para poder cumplir con éxito.

- Las reuniones diarias con el equipo son importantes para conocer cómo van, debido a que la tendencia actual es el trabajo remoto y la comunicación se hace complicada por mensajes; contar con un espacio para interactuar permite al equipo expresarse e informar acerca de sus avances.
- La participación en equipos interdisciplinarios me permitió tener una vista global acerca de las iniciativas y no limitarme a cumplir con lo solicitado, me ayudó a conocer otras herramientas y otros puntos de vista.
- Lo aprendido a nivel teórico me ayudó a contar con una base para poder llevarlo a la práctica donde aparecen la mayoría de las consultas, las cuales se pueden resolver investigando o consultando a algunos pares que tuvieron una experiencia similar.
- Estar en una capacitación constante es bueno para uno mismo como para la compañía, aprovechar las oportunidades para aprender algo nuevo es importante, ya sea por medio de cursos, charlas o investigaciones.
- Se debe fomentar el uso correcto de la tecnología siguiendo las buenas prácticas y el estándar dado por la compañía, ya sea para evitar un reproceso o por la constante evaluación por equipos de auditoría y ciberseguridad.

Como actividades que aún siguen pendientes revisar, destaca la mejora constante del lakehouse, no solo a nivel de arquitectura, sino a nivel de proceso, ya que cada vez son más los usuarios que quieren participar para incrementar la cantidad de datos que se cuenta, y para ello, contar con lineamientos o estándares consolidados facilitará el aprendizaje de estas nuevas personas que no estarán en nuestro equipo, pero si estarán bajo supervisión del área.

Asimismo, a nivel personal me interesaría revisar más el lado de gestión de presupuesto, ya que la gestión del equipo y del tiempo lo he trabajado y un paso adicional a mi crecimiento profesional seria incorporar esa nueva visión, la cual la asume el Líder de Frente, el cual me ha explicado algunos conceptos, pero no tengo el detalle para trabajar uno solo.

6. GLOSARIO DE TÉRMINOS

- ABAP: Es el lenguaje de Programación Avanzada de Aplicaciones Empresariales empleado por los sistemas SAP para la carga y extracción de datos. (Sisyukov et al., 2020)
- Aecorsoft: Es un conjunto de soluciones de integración que ofrece conectividad unificada y acceso a los modelos de datos de SAP asegurando los requisitos de seguridad. (Aecorsoft, n.d.)
- Apache Airflow: Es una plataforma diseñada para crear, programar y monitorear flujos de trabajo utilizando código de programación. Entre sus características están el escalado, dinamismo, extensible y elegancia. (The Apache Software Foundation, n.d.)
- BigQuery: Es una plataforma de análisis de datos completamente administrada y preparada para el uso de Inteligencia Artificial para maximizar el valor de los datos permitiendo trabajar con datos estructurados y no estructurados. (Google, n.d.-a)
- Big Data: Procesamiento y análisis de datos empresariales en un entorno informático de alto rendimiento. (Sisyukov et al., 2020)
- Cloud Composer: Es un servicio administrado que implementa Apache Airflow para la automatización de la programación, administración y seguimiento de flujos de trabajo definidos en Python a través de DAGs. (Google, n.d.-b)
- Cloud Computing: Es un modelo de computación que permite el acceso a la información a través de la red de Internet de manera sencilla y bajo demanda de recursos que se pueden configurar conjuntamente. (Falah et al., 2021)
- Cloud Functions: Es un entorno de ejecución sin servidor para crear servicios en la nube de propósito único a través de eventos emitidos desde una infraestructura u otros servicios en la nube. (Google, n.d.-c)
- Cloud Pub/Sub: Es un servicio de mensajería asíncrono y escalable que procesa los mensajes generados por otros servicios permitiéndolos utilizar para un análisis de transmisión de carga y distribución de datos. (Google, n.d.-e)
- Cloud Storage: Es un servicio administrado para almacenar archivos de cualquier formato en contenedores llamados Buckets en Google Cloud. (Google, n.d.-d)

- DAGs (Gráfico Acíclico Dirigido): Es una especie de gráfico de nodos y aristas que forman una canalización de datos (pipeline), donde cada nodo representa una tarea y cada arista como se está dirigiendo. (Shukla, 2022)
- Daily Meetings: Una reunión diaria para informar lo que ha trabajado cada miembro del equipo, no tienen que estar todos sino los involucrados en el desarrollo de alguna funcionalidad. (Adi, 2015)
- Data Driven: Consiste en el empleo de técnicas para evaluar procesos, productos y servicios mediante el uso de información y conocimiento a partir de los datos. (Schneider et al., 2023)
- Data Lake: Colección masiva de conjunto de datos de diferentes sistemas y formatos obtenidos del sistema que pueden modificarse en el tiempo; no suele ir acompañado de su metadata. (Nargesian et al., 2019)
- Data Mesh: Es un concepto sociotécnico que define a los datos como un producto para su uso federado a través de un modelo democratizado de datos. (Oreščanin et al., 2024)
- Data Warehouse: Colección de datos empleando una base de datos relacional con una estructura definida, permite garantizar el modelado y la gobernanza de datos. (Schneider et al., 2023)
- Docker: Es una plataforma diseñada para que los desarrolladores crean, compartan y ejecuten código en contenedores evitando la configuración y solo centrándose en el código. (Docker, n.d.)
- GitHub: Es una plataforma en la nube que permite guardar, compartir y trabajar en conjunto con otros usuarios escribiendo código. (GitHub, n.d.-a)
- GitHub Actions: Es una plataforma de integración y despliegue continuo (IC/DC) que permite la automatización de la compilación, pruebas y despliegues creando flujos de trabajo por medio de las solicitudes de cambios. (GitHub, n.d.-b)
- Infraestructura como Servicio (IaaS): Permite a los usuarios acceder a varias partes de la infraestructura de la nube, como el almacenamiento, servidores o capas de virtualización. Además de permitir la ejecución de sus propios sistemas operativos y aplicaciones de software. (Borra, 2024)

- Lakehouse: Modelo arquitectónico novedoso que parte de una evolución de la arquitectura de data lake de dos capas; el data warehouse forma parte integral de la arquitectura permitiendo tener un enfoque claro y una estructura de datos. (Oreščanin et al., 2024)
- Odoo: Es un conjunto de aplicaciones de código abierto que cubren las necesidades de una empresa, como contabilidad, CRM, inventario, ventas, recursos humanos, entre otros. (Odoo, n.d.)
- Operator: Son tareas en Airflow, las cuales pueden realizar distintas funciones como ejecutar códigos en Python o crear un servicio como un cluster en Dataproc. (Shukla, 2022)
- Plataforma como Servicio (PaaS): Permite a los usuarios crear y ejecutar aplicaciones sin preocuparse por la infraestructura, herramientas de desarrollo o sistema operativo que son llevados por el proveedor de servicio. (Borra, 2024)
- Product Backlog: Es una lista priorizada de características del producto que desea el Product Owner. (Del Nuevo et al., 2011)
- Product Owner: Representa a los stakeholders (usuarios interesados), el cual consolida los requisitos del producto y empuja al equipo hacia la perspectiva de negocio. (Hayat et al., 2019)
- SAP NetWeaver: Es la base técnica para las aplicaciones SAP, usada como servidor de aplicaciones para establecer un modelo de datos común para BI permitiendo el uso de desarrollos personalizados por medio del lenguaje ABAP. (SAP, n.d.)
- Scrum: Es un marco de respuesta para la elaboración de proyecto de software, gestión de productos o desarrollo de aplicaciones basado en la estrategia de crear un producto holístico y flexible en lugar de seguir un enfoque tradicional. (Adi, 2015)
- Scrum Master: Representa a los desarrolladores y busca eliminar los impedimentos para alcanzar los objetivos del sprint. (Hayat et al., 2019)
- Scrum Team: Es el equipo de desarrollo del proyecto en los que cada integrante destaca por sus habilidades, pero no están sujetos a realizar una sola actividad con el fin de minimizar cualquier impacto en el software. (Saleh et al., 2017)
- Sistemas ERP: Sistema de Planificación de Recursos Empresariales es la cadena analítica y de agregación superior construido para las empresas, donde se cuentan con flujo de datos

consolidados y sin procesar generados por dispositivos inteligentes u otros sistemas (Sisyukov et al., 2020)

- Software como Servicio (SaaS): Los usuarios ya no necesitan instalar un software, ya que se les permite acceder y utilizar las aplicaciones a través de la nube, como Office 365 o Amazon Chime. (Borra, 2024)
- Sprint: Es el bloque más pequeño del Scrum en el que el equipo trabaja un conjunto de tareas asignadas durante 1 a 3 semanas para un entregable (Srivastava et al., 2017)
- Sprint Retrospective: Una reunión donde el equipo analiza el Sprint anterior para ver cómo se trabajó y que lecciones aprendidas tienen que aplicar en el siguiente. (Del Nuevo et al., 2011)
- Streaming de datos: Realizar análisis y procesamiento de datos en tiempo real (Falah et al., 2021)

REFERENCIAS

ACM. (n.d.). Código de Ética y Conducta Profesional de ACM. Https://Www.Acm.Org/about-Acm/Code-of-Ethics-in-Spanish.

Adi, P. (2015). Scrum Method Implementation in a Software Development Project Management. International Journal of Advanced Computer Science and Applications, 6(9). https://doi.org/10.14569/IJACSA.2015.060927

Aecorsoft. (n.d.). Aecorsoft. Retrieved July 19, 2024, from https://www.aecorsoft.com/

Borra, P. (2024). Comparison and analysis of leading cloud service providers (AWS, AZURE and GCP). International Journal of Advanced Research in Engineering and Technology (IJARET), 15(3), 266–278.

Chauhan, A. (2020). A Comparative Study of Cloud Computing Platforms. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 11(1), 821–826. https://doi.org/10.17762/turcomat.v11i1.13563

Del Nuevo, E., Piattini, M., & Pino, F. J. (2011). Scrum-based Methodology for Distributed Software Development. 2011 IEEE Sixth International Conference on Global Software Engineering, 66–74. https://doi.org/10.1109/ICGSE.2011.23

Docker. (n.d.). What is Docker? Retrieved July 19, 2024, from https://www.docker.com/

Falah, M. F., Fridelin Panduman, Y. Y., Sukaridhoto, S., Cornelius Tirie, A. W., Kriswantoro, M. C., Satria, B. D., & Usman, S. (2021). Comparison of cloud computing providers for development of big data and internet of things application. Indonesian Journal of Electrical Engineering and Computer Science, 22(3), 1723. https://doi.org/10.11591/ijeecs.v22.i3.pp1723-1730

GitHub. (n.d.-a). About GitHub and Git. Retrieved July 19, 2024, from https://docs.github.com/en/get-started/start-your-journey/about-github-and-git

GitHub. (n.d.-b). Understanding GitHub Actions. Retrieved July 19, 2024, from https://docs.github.com/en/actions/learn-github-actions/understanding-github-actions

Google. (n.d.-a). BigQuery overview. Retrieved July 19, 2024, from https://cloud.google.com/bigquery/docs/introduction

Google. (n.d.-b). Cloud Composer overview. Retrieved July 19, 2024, from https://cloud.google.com/composer/docs/composer-2/composer-overview

Google. (n.d.-c). Cloud Functions overview. Retrieved July 19, 2024, from https://cloud.google.com/functions/docs/concepts/overview

Google. (n.d.-d). Product overview of Cloud Storage. Retrieved July 19, 2024, from https://cloud.google.com/storage/docs/introduction

Google. (n.d.-e). What is Pub/Sub? Retrieved July 19, 2024, from https://cloud.google.com/pubsub/docs/overview

Hayat, F., Rehman, A. U., Arif, K. S., Wahab, K., & Abbas, M. (2019). The Influence of Agile Methodology (Scrum) on Software Project Management. 2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 145–149. https://doi.org/10.1109/SNPD.2019.8935813

Mazumdar, D., Hughes, J., & Onofre, J. (2023). The Data Lakehouse: Data Warehousing and More.

Nargesian, F., Zhu, E., Miller, R. J., Pu, K. Q., & Arocena, P. C. (2019). Data lake management. Proceedings of the VLDB Endowment, 12(12), 1986–1989. https://doi.org/10.14778/3352063.3352116

Nugroho, S., Waluyo, S. H., & Hakim, L. (2017). Comparative Analysis of Software Development Methods between Parallel, V-Shaped and Iterative. https://doi.org/10.5120/ijca2017914605

Odoo. (n.d.). Odoo. Retrieved July 19, 2024, from https://www.odoo.com/es

Oreščanin, D., Hlupić, T., & Vrdoljak, B. (2024). Managing Personal Identifiable Information in Data Lakes. IEEE Access, 12, 32164–32180. https://doi.org/10.1109/ACCESS.2024.3365042

Raju, R., Mital, R., & Finkelsztein, D. (2018). Data Lake Architecture for Air Traffic Management. 2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC), 1–6. https://doi.org/10.1109/DASC.2018.8569361

Saleh, S., Rahman, M., & Pavel, K. (2017). Comparative Study on the Software Methodologies for Effective Software Development. International Journal of Scientific and Engineering Research, 8, 1018–1025.

SAP. (n.d.). SAP NetWeaver. Retrieved July 19, 2024, from https://www.sap.com/products/technology-platform/netweaver.html

Schneider, J., Gröger, C., Lutsch, A., Schwarz, H., & Mitschang, B. (2023). Assessing the Lakehouse: Analysis, Requirements and Definition. Proceedings of the 25th International Conference on Enterprise Information Systems, 44–56. https://doi.org/10.5220/0011840500003467

Shukla, S. (2022). Developing Pragmatic Data Pipelines using Apache Airflow on Google Cloud Platform. International Journal of Computer Sciences and Engineering , 10(8), 1–8.

Sisyukov, A. N., Bondarev, V. K., & Yulmetova, O. S. (2020). ERP Data Analysis and Visualization in High-Performance Computing Environment. 2020 IEEE

Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), 509–512. https://doi.org/10.1109/EIConRus49466.2020.9038949

Srivastava, A., Bhardwaj, S., & Saraswat, S. (2017). SCRUM model for agile methodology. 2017 International Conference on Computing, Communication and Automation (ICCCA), 864–869. https://doi.org/10.1109/CCAA.2017.8229928

The Apache Software Foundation. (n.d.). Apache Airflow. Https://Airflow.Apache.Org/. Retrieved July 19, 2024, from https://airflow.apache.org/

Zaharia, M. A., Ghodsi, A., Xin, R., & Armbrust, M. (2021). Lakehouse: A New Generation of Open Platforms that Unify Data Warehousing and Advanced Analytics. Conference on Innovative Data Systems Research.

BIBLIOGRAFÍA

Cuzzocrea, A. (2021). Big Data Lakes: Models, Frameworks, and Techniques. 2021 IEEE International Conference on Big Data and Smart Computing (BigComp), 1–4. https://doi.org/10.1109/BigComp51126.2021.00010

Faniran, V. T., Badru, A., & Ajayi, N. (2017). Adopting Scrum as an Agile approach in distributed software development: A review of literature. 2017 1st International Conference on Next Generation Computing Applications (NextComp), 36–40. https://doi.org/10.1109/NEXTCOMP.2017.8016173

Garzaniti, N., Briatore, S., Fortin, C., & Golkar, A. (2019). Effectiveness of the Scrum Methodology for Agile Development of Space Hardware. 2019 IEEE Aerospace Conference, 1–8. https://doi.org/10.1109/AERO.2019.8741892

Harby, A. A., & Zulkernine, F. (2022). From Data Warehouse to Lakehouse: A Comparative Review. 2022 IEEE International Conference on Big Data (Big Data), 389–395. https://doi.org/10.1109/BigData55660.2022.10020719

Kukreja, M., & Zburivsky, D. (2021). Data Collection Stage - The Bronze Layer. In Data Engineering with Apache Spark, Delta Lake, and Lakehouse: Create scalable pipelines that ingest, curate, and aggregate complex data in a timely and secure way (pp. 98–106). Packt Publishing Ltd.

Menon, P. (2022). The Data Lakehouse Architecture Overview. In Data Lakehouse in Action: Architecting a modern and scalable data analytics platform (pp. 20–34). Packt Publishing.

Singh, A. (2019). Architecture of Data Lake. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 411–414. https://doi.org/10.32628/CSEIT1952121

3% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- Bibliografía
- Coincidencias menores (menos de 10 palabras)

Exclusiones

- N.º de fuente excluida

Fuentes principales

3%	Fuentes de Internet
0%	Publicaciones
2%	Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alertas de integridad para revisión

No se han detectado manipulaciones de texto sospechosas.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.