UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

DESARROLLO E IMPLEMENTACIÓN DE UN MODELO DIMENSIONAL PARA EL PROCESO DE NEGOCIO DE PREGUNTAS Y RESPUESTAS DE LA PLATAFORMA STACK OVERFLOW

PRESENTADO POR:

HENRY IVAN GODOY GUTIERREZ

NELSON ENRIQUE MIRANDA MIRANDA

JOSÉ RODRIGO PRESA MARIONA

PARA OPTAR AL TITULO DE:

INGENIERO(A) DE SISTEMAS INFORMÁTICOS

CIUDAD UNIVERSITARIA, ENERO 2022

UNIVERSIDAD DE EL SALVADOR

RECTOR:

MSC. ROGER ARMANDO ARIAS ALVARADO

SECRETARIO GENERAL:

ING. FRANCISCO ANTONIO ALARCON SANDOVAL

FACULTAD DE INGENIERÍA Y ARQUITECTURA

DECANO:

DOCTOR EDGAR ARMANDO PEÑA FIGUEROA

SECRETARIO:

ING. JULIO ALBERTO PORTILLO

ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

DIRECTOR:

ING. RUDY WILFREDO CHICAS VILLEGAS

UNIVERSIDAD DE EL SALVADOR

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

Trabajo de Graduación previo a la opción al Grado de:

INGENIERO(A) DE SISTEMAS INFORMÁTICOS

Título :

DESARROLLO E IMPLEMENTACIÓN DE UN MODELO DIMENSIONAL PARA EL PROCESO DE NEGOCIO DE PREGUNTAS Y RESPUESTAS DE LA PLATAFORMA STACK OVERFLOW

Presentado por:

HENRY IVAN GODOY GUTIERREZ

NELSON ENRIQUE MIRANDA MIRANDA

JOSÉ RODRIGO PRESA MARIONA

Trabajo de Graduación Aprobado por:

Docente Asesor:

ING. MARLON ARMANDO MENJIVAR MARTINEZ

SAN SALVADOR, ENERO 2022

Trabajo de Graduación Aprobado por:

Docente Asesor:

ING. MARLON ARMANDO MENJIVAR MARTINEZ

Índice

Índice de figuras

Índice de tablas

1. Introducción

En los últimos años las empresas han tomado muy en cuenta los datos que ellas mismas generan, ya que estos datos se han convertido en el principal activo para la toma de decisiones estratégicas. Con la evolución de las tecnologías a pasos agigantados, se ha contribuido al surgimiento de diferentes fuentes de datos, iniciando por los procesos de negocio tradicionales, hasta el Internet de las Cosas, estos últimos generando estos grandes volúmenes de datos. Las innumerables interacciones que las personas tienen con las tecnologías, conducen a que los datos no solo se produzcan en gran volumen, sino que también en gran variedad. Como parte de la automatización en los procesos de negocio, se incorporan sistemas y maquinaria, provocando que se generen muchísimos datos, en grandes volúmenes, en gran variedad y a gran velocidad por tanto las empresas llegan a tener un banco de datos potencialmente gigantesco.

Para el análisis de Big data se necesita un almacén de datos que soporte múltiples fuentes y formatos de datos, ya que las técnicas convencionales de análisis, procesamiento y almacenamiento de datos son insuficientes. Por lo que se necesita implementar una solución de Big data que buscará satisfacer las diferentes necesidades que las empresas tengan o que desean resolver, tales como la optimización de sus procesos, predecir comportamientos en las ventas o compras; mejorando así el proceso de toma de decisiones estratégica.

En la actualidad la computación en la nube ha sufrido un gran crecimiento y gran demanda, por otra parte, los datos se generan muy rápido, muy variados y en grandes volúmenes, por lo que una solución de Big data encaja de manera perfecta para el análisis adecuado de los datos. Es por ello que se desarrollará una solución de Big data para uno de los sitios web que cuentan con una gran reputación en el ámbito de los profesionales de informática, desarrollo y uso de software el cual es: Stack Overflow, un sitio de preguntas y respuestas muy utilizado. Básicamente es una comunidad colaborativa en el que coexisten muchas profesiones y personalidades de tal manera que en conjunto contribuyen a la solución de problemas que acontecen en el mundo laboral en el área de informática.

Stack Overflow una comunidad gigantesca de profesionales que interactúan constantemente para la resolución de problemas informáticos. Iniciando con un usuario que realiza una pregunta, esta recibe revisiones, vistas, comentarios y respuestas, el usuario acumula votos, sube de reputación, gana insignias, y se van contando el número de preguntas y respuestas hechas en la plataforma. Por lo que todas las interacciones generan enormes cantidades de datos, tanto de usuarios, preguntas, respuestas, votos, comentarios, puntuaciones, tecnologías, es decir todo lo que respecta a la interacción entre los miembros de esta comunidad internauta.

Resulta de interés entonces poder explotar al máximo esa enorme cantidad de datos, darle sentido, y utilizar los potenciales hallazgos para que la comunidad de Stack Overflow pueda apoyarse y mejorar sus correspondientes procesos de planificación estratégica. Logrando que la comunidad pueda tomar sus decisiones basada en sus propios datos, datos que les permitan reorientar sus esfuerzos hacia la mejora continua. Básicamente consistirá en adoptar una nueva variable de interés que les permitan poder estar enterados de lo que realmente está pasando con la plataforma.

Para implementar la solución de Bi data se tendrá de base un Data Lake. Un Data Lake es un almacén de datos masivo en el caso del Data Lake que se utilizará se dividirá en tres capas principales, las cuales sirven para el almacenamiento de la data recién recolectada, pre procesada y finalmente procesada. Como marco de trabajo se adoptará el ciclo de vida de desarrollo de Data Warehouse el cual está

compuesto por un conjunto de elementos que orientan perfectamente la labor que se debe de seguir para la construcción del mismo.

Iniciando con la planificación del proyecto, así como también paralelamente con la definición oficial de los principales requerimientos de negocio, luego de que los requerimientos estén perfectamente establecidos, se proseguirá con la realización del perfilado de datos. Utilizando una potencial herramienta existente en el mercado, que nos permitirá de manera directa conocer el estado actual de la data, así como también adquirir el conocimiento de la misma para poder tomar decisiones importantes para actividades posteriores del ciclo de vida.

La herramienta generará una serie de reportes sobre el estado actual de los datos, a partir de estos reportes se construirá un resumen del análisis. Cada tabla se desglosará en cada uno de sus correspondientes campos que la conforman, para cada campo se dará a conocer el diagnóstico y solución a aplicar, así como también la aprobación para la utilización de los mismos para la construcción del modelo dimensional.

Luego paralelamente se llevará a cabo la elección del diseño técnico arquitectico del proyecto, selección e instalación de los productos a utilizar. Aplicación del modelado dimensional el cual está compuesto por la selección del proceso de negocio, definición de granularidad, identificación de dimensiones y determinación de las métricas. Obteniendo como resultado el diseño del modelo dimensional final, que será lo suficiente potente para no solo satisfacer las necesidades actuales si no necesidades futuras.

Habiendo construido el modelo, se prosigue con la construcción del mapeo de datos para cada uno de los componentes del modelo dimensional. En el mapeo se dan a conocer las indicaciones directas para la construcción de cada una de las dimensiones, Fact tables, bridges que conformarán el modelo.

Luego se proseguirá con la construcción de los correspondientes ETL’s creados con el lenguaje de programación Scala, dichos ETLs se ejecutarán en la plataforma on cloud Databricks que internamente se encuentra ejecutando Apache Spark, que permitirá llevar a cabo toda esta lógica de procesamiento de enormes cantidades de datos, la lógica llevara consigo todos los aspectos que se definieron en el perfilado de datos. Teniendo el modelo construido se proseguirá con la puesta en producción, se implementará con una arquitectura técnica que permita consumir los datos por parte de herramientas externas, los datos podrán ser consumidos para diferentes propósitos, pero en este caso específico se consumirá para Inteligencia Negocio (BI), para ello se utilizará un software especializado para construir reportes que permitan responder de manera directa las necesidades analíticas propuestas.

Los datos del modelo estarán localizados en el almacén de datos de ByQuery, a través del cual Power BI se podrá conectar en modo de consumo directo desde la base de datos. Se implementará el modelo en la herramienta para que posteriormente alimente a la construcción de las diferentes visualizaciones que conformaran los respectivos reportes que darán respuesta a las preguntas analíticas planteadas.

2. Objetivos

a. Objetivo General

Desarrollar e implementar un modelo dimensional para el proceso de negocio de preguntas y respuestas de la plataforma Stack Overflow.

b. Objetivos Específicos

- Analizar las fuentes de datos disponibles de la plataforma Stack Overflow.
- Realizar un perfilado de datos del Dataset de la plataforma Stack Overlow.
- Diseñar el modelo dimensional para el proceso de negocio de preguntas y respuestas realizadas en la plataforma Stack Overflow.
- Construir el proceso de extracción, transformación y carga de datos para el proceso de negocio de preguntas y respuestas realizadas en la plataforma Stack Overflow.
- Integrar el modelo dimensional desde BigQuery con Power BI para la generación de visualizaciones que contribuyan al análisis del proceso de preguntas y respuestas de la plataforma Stack Overflow.

3. Planteamiento del problema.

Objetivo: *Elaborar reportes analíticos a partir de una solución de big data que permita de manera concreta a la comunidad de Stack Overflow identificar a través de la resolución de preguntas que surgen de los procesos de negocio patrones de comportamientos a nivel de usuario dentro de la plataforma, a nivel de la cantidad de respuestas por preguntas hechas, a nivel de periodos de tiempos de incrementos y necesidades de resolución.*

Salidas:

- ✓ *Reportes o Dasboards que permitan a la comunidad de Stack Overflow obtener a través de las visualizaciones que lo conforman la información oportuna para la resolución de la necesidad analítica actual y futuras para apoyar sus correspondientes procesos de toma de decisiones estratégicas.*

Entradas:

- ✓ *Dataset publico correspondiente a la plataforma de Stack Overflow en formato csv, ubicado en el almacén de datos BigQuery.*

Proceso:

- ✓ Procesos ETL's [Extracción, transformación y carga de datos].

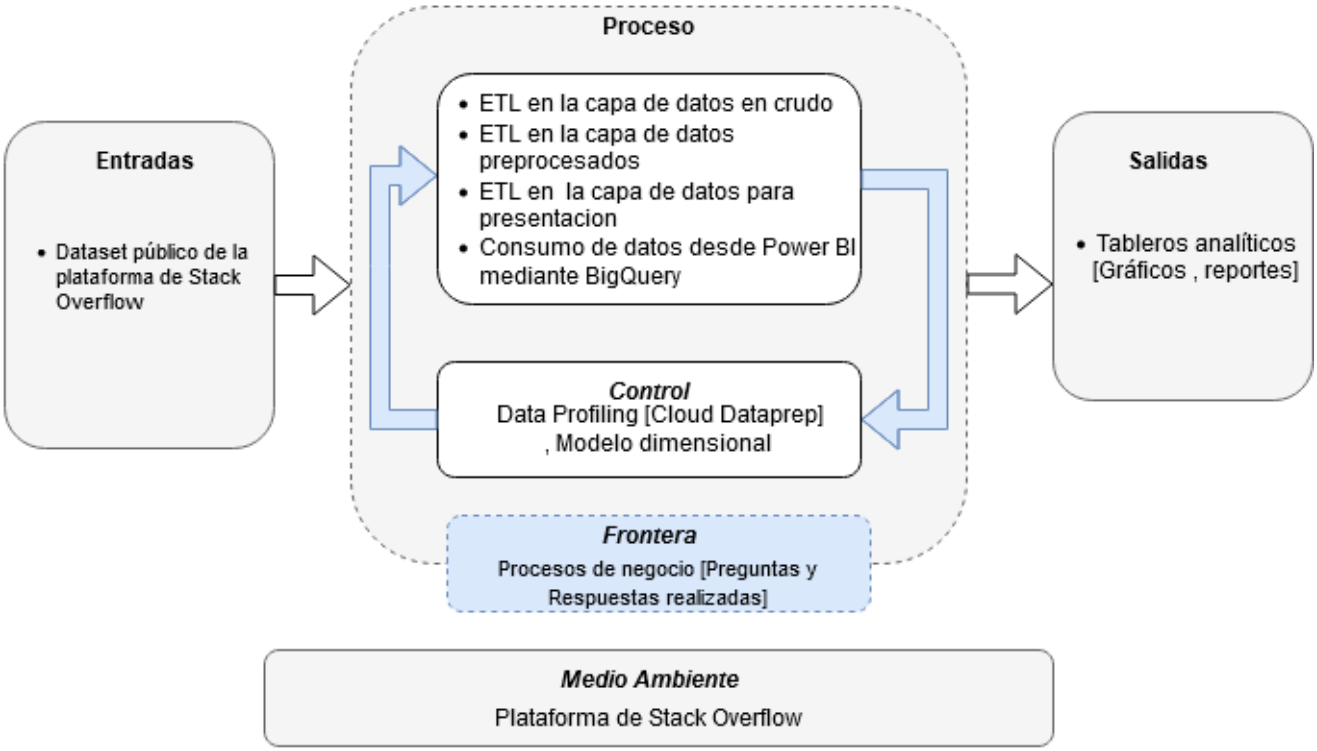

Figura 1. Diagrama del Enfoque de sistemas aplicado.

4. Importancia.

La capacidad que las empresas adquieren en la toma de decisiones basadas en sus datos, las potencializa de manera directa, ya que están apoyando sus decisiones en base al resultado del análisis de sus propios datos, datos que sus procesos de negocio generan en el día a día. El proceso de análisis de sus datos es toda una macro tarea dividida en la recolección, preparación, limpieza, organización y análisis de grandes cantidades de datos, lo que se busca es extraer información de manera directa, la cual debe de ser lo suficientemente útil para identificar los correspondientes parámetros que evidenciaran el comportamiento de la organización.

Como resultado del proceso de análisis aplicado, se elaboran por parte de personas especializadas en el área, una serie de reportes elementales que les permitirán apoyar sus procesos de construcción de estrategias, metas y objetivos, como parte de su planificación estratégica que las organizaciones emplean en su horizonte dentro de los mercados competitivos, por ende adquieren esa capacidad para poder tomar todas sus decisiones con la seguridad y certeza requerida, las cuales a su vez tendrán un impacto en la organización y en sus procesos de negocio.

Una empresa que no tome en cuenta sus datos de sus procesos de negocio a la hora de tomar decisiones, no estará al tanto de lo que realmente está pasando, los acontecimientos que se pueden llegar a tener, el estado actual de sus procesos de negocio, causas principales de sus pérdidas, bajas en los ingresos, estado actual de su flota, nivel de aceptación de sus servicios, necesidades de cambios en las políticas operativas actuales etc. Los datos dotan a los tomadores de decisión sobre un panorama amplio sobre lo que realmente está pasando en la organización.

Es tan importante confiar en sus propios datos, porque pueden permitirles prever acontecimientos futuros, como comportamientos en el mercado, variaciones en los precios, nivel de aceptación de sus clientes, disminuir riesgos, reducir costos, mejorar la calidad de los servicios, aplicar estándares etc.

Resulta interesante entonces para la comunidad de Stack Overflow, poder conocer de primera a mano la situación actual de la plataforma, esto a través de los propios datos que generan sus procesos de negocios, para así poder identificar patrones de comportamiento en los usuarios al momento de realizar preguntas y respuestas en el sitio , patrones de comportamiento en cuanto a las cantidades de respuestas que se reciben por preguntas, los periodos de tiempo en los que estas incrementan, los periodos en los que se necesita que la plataforma sea lo suficiente colaborativa para que todos los usuarios se ayuden mutuamente.

Una de las áreas más importantes dentro de la plataforma es el desempeño de los usuarios, cada uno de ellos pueden tener una serie de insignias, las cuales son asignadas según su grado de aportación ya que esto les hace recibir ciertos permisos dentro de la misma. Otra de las necesidades es poder detectar patrones de comportamiento de las principales tecnologías que más generan preguntas y respuestas, esto permitirá saber cuáles son las tecnologías con mayor demanda, e incluso poder dar mayor suporte a esas tecnologías. Esto permitirá a la comunidad de Stack Overflow poder reorientar sus esfuerzos, tomar mejores decisiones basándose en sus datos, y garantizar que la plataforma se potencialice cada vez más a corto, mediano y largo plazo.

5. Justificación.

Con el aparecimiento de la pandemia, muchas de las organizaciones tuvieron que cambiar su metodología de trabajo, y migraron muchas de sus actividades laborales a los correspondientes hogares de sus empleados, esto evidencio claramente la importancia que puede tener una plataforma como Stack Overflow en el área de la informática, ya que las personas en la realización de sus actividades laborales necesitaban de un sitio de consulta colaborativo, el cual les permitiera resolver sus interrogantes. Esto demuestra lo importante de que dicha plataforma se encuentre activa, así como que sea lo suficientemente sostenible, y colaborativa para poder dar abasto a la innumerable cantidad de preguntas que se realizan por parte de los usuarios.

¿Por qué tomamos en cuenta los datos que la plataforma genero durante este periodo crítico de pandemia?, porque básicamente permitirá a la comunidad detectar los principales patrones de comportamiento de sus usuarios, numero de preguntas hechas , respuestas hechas , tecnologías consultadas , contribución de la comunidad , votos , comentarios , puntajes como producto de la interacción que la comunidad tiene en todo momento, y a través del análisis de estos datos comparados con años anteriores, y actuales permitirá a la plataforma poder evaluar si realmente se tiene la capacidad para la cual fue construida.

Esto es debido a que muchas veces, de todas las preguntas que son realizadas, unas pueden llegar a tener muy poca retroalimentación, por lo que la persona no recibe totalmente la ayuda que se esperaba, también se pueden detectar comportamientos en los usuarios en cuanto a la contribución que estos realizan dentro de la comunidad, así evaluar si realmente están cumpliendo con los verdaderos objetivos de la plataforma.

En base a los resultados obtenidos, la plataforma podrá dar a conocer a la comunidad sobre su situación actual, descubrir que puede estar afectando su rendimiento, que políticas pueden ser mejoradas con tal de mejorar el puntaje de preguntas resueltas, respuestas más oportunas y válidas. Sobre todo, en que se puede mejorar, ya sea optimizando los recursos actuales o incorporando nuevos.

La presente solución de Big Data recolectará toda la data generada por la plataforma en los últimos años, sin embargo, para motivos de análisis, la solución permitirá la selección del periodo de manera dinámica, de manera que el análisis podrá realizarse para el periodo deseado, se podrá analizar información concreta de los usuarios, sus insignias y reputación, preguntas y respuestas realizadas, tecnologías con mayor demanda, comentarios de parte de los usuarios, etc. Se responderá a las necesidades analíticas establecidas en el proceso de identificación de requerimientos, permitiendo a la plataforma y comunidad en general, mejorar y maximizar la capacidad de la de la misma en relación a la colaboración, para poder resolver de manera más oportuna las diferentes necesidades que los usuarios puedan llegar a tener.

6. Limitaciones.

Las principales limitaciones que posee el proyecto se enumeran a continuación:

1. La principal fuente de datos de la cual se alimentará al proyecto proviene de un conjunto de datos público, almacenado en BigQuery, siendo este Dataset compartido para la comunidad, sin embargo dicho Dataset se encuentra limitado en cuanto a las tablas disponibles, ya que solo se encuentras ciertas tablas del modelo completo que usa Stack Overflow, por lo que nos veremos sujetos a utilizar las tablas disponibles, no obstante las tablas gozan de la suficiente completitud para poder realizar el proyecto y construir el modelo dimensional en base a las necesidades analíticas.

7. Alcances.

Para el proyecto de desarrollo e implementación de un modelo dimensional para el proceso de negocio de preguntas y respuestas realizadas en la plataforma Stack Overflow, se presentan a continuación los siguientes alcances:

1. Brindar un esquema estrella del modelo dimensional que tendrá la potencialidad analítica para poder responder a las necesidades analíticas actuales que originaron el proyecto, así como también está diseñado para poder tener la capacidad de poder responder a futuros requerimientos analíticos o necesidades relacionadas al proceso de negocio de preguntas y respuestas.

2. Brindar una solución de Data Lakehouse sobre los datos generados en la plataforma de Stack Overflow para que pueden apoyarse en el proceso de toma de decisiones basadas en sus propios datos.

3. Presentación de Dashboards que den solución a las necesidades analíticas que originaron el proyecto, de tal manera que los resultados permitirán reorientar los esfuerzos para la contribución del crecimiento de la plataforma a corto, mediana y largo plazo.

4. Presentación de documentación en formato digital que permita a los usuarios miembros de la comunidad de Stack Overflow poder entender la solución completa de Big data que se aplicó; desde la construcción hasta la implementación del modelo dimensional en la plataforma de BigQuery para el correspondiente consumo desde la herramienta Power Bi.

5. Disponibilidad de un repositorio en GitHub que contendrá los pasos técnicos que se siguieron para el desarrollo del proyecto, el repositorio estará dividido por cuatro etapas: perfilado de datos, diseño dimensional, desarrollo e implementación y resultados obtenidos.

8. Marco Teórico

8.1 Historia del Análisis de datos.

Una descripción más útil y moderna sugerida es que el "análisis de datos" es una herramienta importante para obtener información comercial y brindar respuestas personalizadas a los clientes. El análisis de datos, a veces abreviado como "análisis", se ha vuelto cada vez más importante para organizaciones de todos los tamaños. La práctica del análisis de datos ha evolucionado y ampliado gradualmente con el tiempo, brindando muchos beneficios.

El uso de análisis por parte de las empresas se remonta al siglo XIX, cuando Frederick Winslow Taylor inició ejercicios de gestión del tiempo. Otro ejemplo es cuando Henry Ford midió la velocidad de las cadenas de montaje. A fines de la década de 1960, Analytics comenzó a recibir más atención a medida que las computadoras se convirtieron en sistemas de apoyo para la toma de decisiones. Con el desarrollo de Big data , almacenes de datos, la nube y una variedad de software y hardware, el análisis de datos ha evolucionado significativamente. El análisis de datos implica la investigación, el descubrimiento y la interpretación de patrones dentro de los datos. Las formas modernas de análisis de datos se han ampliado para incluir:

a. Análisis predictivo
b. Análisis de grandes datos
c. Análisis cognitivo
d. Analítica prescriptiva
e. Analítica descriptiva
f. Gestión de decisiones empresariales
g. Análisis minorista
h. Análisis aumentado
i. analista de la red
j. Análisis de llamadas

Estadística y Computación

El análisis de datos se basa en estadísticas. Se ha supuesto que las estadísticas se utilizaron desde el Antiguo Egipto para construir pirámides. Los gobiernos de todo el mundo han utilizado estadísticas basadas en censos para una variedad de actividades de planificación, incluida la fiscalidad. Una vez que se han recopilado los datos, comienza el objetivo de descubrir información y conocimientos útiles. Por ejemplo, un análisis del crecimiento de la población por departamento y ciudad podría determinar la ubicación de un nuevo hospital.

El desarrollo de las computadoras y la evolución de la tecnología informática ha mejorado drásticamente el proceso de análisis de datos. En 1880, antes de las computadoras, la Oficina del Censo de EE. UU. tardó más de siete años en procesar la información recopilada y completar un informe final. En respuesta, el inventor Herman Hollerith[1] produjo la “máquina tabuladora”, que se utilizó en el censo de 1890. La máquina tabuladora podría procesar sistemáticamente los datos registrados en las tarjetas perforadas . Con este dispositivo, el censo de 1890 se terminó en 18 meses.

[1] Herman Hollerith fue un inventor que desarrolló un tabulador electromagnético de tarjetas perforadas para ayudar en el resumen de la información y, más tarde, la contabilidad.

Bases de datos relacionales y bases de datos no relacionales

Las bases de datos relacionales fueron inventadas por Edgar F. Codd[2]en la década de 1970 y se hicieron muy populares en la década de 1980. Las bases de datos relacionales (RDBM), a su vez, permitieron a los usuarios escribir en secuencia (SQL) y recuperar datos de su base de datos. Las bases de datos relacionales y SQL brindaron la ventaja de poder analizar datos a pedido y todavía se usan ampliamente. Es fácil trabajar con ellos y muy útiles para mantener registros precisos. En el lado negativo, los RDBM generalmente son bastante rígidos y no fueron diseñados para traducir datos no estructurados.

A mediados de la década de 1990, Internet se volvió extremadamente popular, pero las bases de datos relacionales no podían seguir el ritmo. El inmenso flujo de información combinado con la variedad de tipos de datos provenientes de muchas fuentes diferentes, dio lugar a bases de datos no relacionales, también conocidas como NoSQL. Una base de datos NoSQL puede traducir datos usando diferentes idiomas y formatos rápidamente y evita la rigidez de SQL reemplazando su almacenamiento "organizado" con mayor flexibilidad.

El desarrollo de NoSQL fue seguido por cambios en Internet. Larry Page y Sergey Brin [3]diseñaron el motor de búsqueda de Google para buscar en un sitio web específico, mientras procesan y analizan Big data en computadoras distribuidas. El motor de búsqueda de Google puede responder en unos segundos con los resultados deseados. Los principales puntos de interés del sistema son su escalabilidad, automatización y alto rendimiento.

Almacenes de datos

A fines de la década de 1980, la cantidad de datos recopilados siguió creciendo significativamente, en parte debido a los menores costos de las unidades de disco duro. Durante este tiempo, la arquitectura de los almacenes de datos se desarrolló para ayudar a transformar los datos provenientes de los sistemas operativos en sistemas de apoyo a la toma de decisiones. Los almacenes de datos son normalmente parte de la nube o parte del servidor central de una organización. A diferencia de las bases de datos relacionales, un almacén de datos normalmente está optimizado para un tiempo de respuesta rápido a las consultas. En un almacén de datos, los datos a menudo se almacenan mediante una marca de tiempo y los comandos de operación, como ELIMINAR o ACTUALIZAR, se usan con menos frecuencia. Si todas las transacciones de ventas se almacenaran usando marcas de tiempo, una organización podría usar un almacén de datos para comparar las tendencias de ventas de cada mes.

Inteligencia de negocios

El término inteligencia empresarial (BI) se utilizó por primera vez en 1865 y luego fue adaptado por Howard Dresner en Gartner en 1989, para describir la toma de mejores decisiones comerciales a través de la búsqueda, recopilación y análisis de los datos acumulados guardados por una organización. Usar el término "inteligencia de negocios" como una descripción de la toma de decisiones basada en tecnologías de datos fue novedoso y con visión de futuro. Las grandes empresas primero adoptaron BI en la forma de analizar los datos de los clientes de manera sistemática, como un paso necesario para tomar decisiones comerciales.

[2] Edgar Frank "Ted" Codd fue un científico informático inglés, conocido por crear el modelo relacional de bases de datos.

[3] Lawrence Edward Page es un ingeniero informático y empresario estadounidense, creador junto con Serguéi Brin , un empresario e informático teórico estadounidense de Google (Alphabet).

Procesamiento de datos

La minería de datos comenzó en la década de 1990 y es el proceso de descubrir patrones dentro de grandes conjuntos de datos. El análisis de datos de formas no tradicionales proporcionó resultados sorprendentes y beneficiosos. El uso de la minería de datos surgió directamente de la evolución de las tecnologías de bases de datos y almacenes de datos. Las nuevas tecnologías permiten a las organizaciones almacenar más datos, sin dejar de analizarlos de forma rápida y eficiente. Como resultado, las empresas comenzaron a predecir las necesidades potenciales de los clientes, basándose en un análisis de sus patrones de compra históricos.

Sin embargo, los datos pueden ser malinterpretados. Alguien en los oficios, habiendo comprado dos pares de jeans azules en línea, probablemente no querrá comprar jeans por otros dos o tres años. Dirigirse a esta persona con anuncios de blue jeans es tanto una pérdida de tiempo como irritante para el cliente potencial.

Big Data

En 2005, Roger Magoulas [4]le dio ese nombre a Big data. Estaba describiendo una gran cantidad de datos, que parecían casi imposibles de manejar con las herramientas de Business Intelligence disponibles en ese momento. En el mismo año, Hadoop podía procesar grandes volúmenes de datos. La base de Hadoop se basó en otro marco de software de código abierto llamado Nutch, que luego se fusionó con MapReduce de Google.

Apache Hadoop es un marco de software de código abierto, que puede procesar datos estructurados y no estructurados, transmitidos desde casi todas las fuentes digitales. Esta flexibilidad permite que Hadoop (y sus marcos hermanos de código abierto) procesen Big data. A fines de la década de 2000, surgieron varios proyectos de código abierto, como Apache Spark y Apache Cassandra , para enfrentar este desafío.

Analítica en la Nube

En su forma inicial, la nube era una frase que se usaba para describir el "espacio vacío" entre los usuarios y el proveedor. Luego, en 1997, el profesor de la Universidad de Emory, Ramnath Chellappa[5], describió la computación en la nube como un nuevo "paradigma informático donde los límites de la computación estarán determinados por la lógica económica, en lugar de los límites técnicos únicamente".

En 1999, Salesforce proporcionó un ejemplo muy temprano de cómo usar la computación en la nube con éxito. Aunque primitivo para los estándares actuales, Salesforce usó el concepto para desarrollar la idea de entregar programas de software a través de Internet. Los programas (o aplicaciones) pueden ser accedidos o descargados por cualquier persona con acceso a Internet. Un gerente de la organización podría comprar software en un método bajo demanda rentable sin salir de la oficina. A medida que las empresas y las organizaciones obtuvieron una mejor comprensión de los servicios y la utilidad de la nube, ganó popularidad.

[4] Roger Magoulas es el director de investigación de mercado de O'Reilly Media, dirige un equipo que está construyendo una infraestructura de análisis de código abierto y proporciona servicios de análisis, incluido el análisis de tendencias tecnológicas.

[5] Ramnath Chellappa, profesor titular de gestión de operaciones y sistemas de Información de la fundación goizueta en la escuela de negocios goizueta, universidad de emory.

La nube ha evolucionado significativamente desde 1999, con clientes que “alquilan los servicios”, en lugar de adquirir hardware y software con el mismo propósito. Los proveedores ahora son responsables de la resolución de problemas, las copias de seguridad, la administración, la planificación de la capacidad y el mantenimiento. Y, para varios proyectos empresariales, la nube es simplemente más fácil y eficiente de usar. La nube ahora tiene cantidades significativamente grandes de almacenamiento, disponibilidad para múltiples usuarios simultáneamente y la capacidad de manejar múltiples proyectos.

Análisis predictivo

El análisis predictivo se utiliza para hacer pronósticos sobre tendencias y patrones de comportamiento. El análisis predictivo utiliza varias técnicas tomadas de estadísticas, modelado de datos, minería de datos, inteligencia artificial y aprendizaje automático para analizar datos al hacer predicciones. Los modelos predictivos pueden analizar datos actuales e históricos para comprender a los clientes, los patrones de compra, los problemas de procedimiento y predecir peligros y oportunidades potenciales para una organización.

El análisis predictivo comenzó en la década de 1940, cuando los gobiernos comenzaron a usar las primeras computadoras. Aunque ha existido durante décadas, el análisis predictivo ahora se ha convertido en un concepto cuyo momento ha llegado. Con más y más datos disponibles, las organizaciones han comenzado a usar análisis predictivos para aumentar las ganancias y mejorar su ventaja competitiva. El crecimiento continuo de los datos almacenados, combinado con un interés cada vez mayor en el uso de datos para obtener Business Intelligence, ha promovido el uso de análisis predictivos.

Análisis cognitivo

La mayoría de las organizaciones manejan datos no estructurados. Dar sentido a estos datos no estructurados no es algo que los humanos puedan hacer fácilmente. El análisis cognitivo combina una variedad de aplicaciones para proporcionar contexto y respuestas. Las organizaciones pueden recopilar datos de varias fuentes diferentes, y el análisis cognitivo puede examinar los datos no estructurados en profundidad, ofreciendo a los responsables de la toma de decisiones una mejor comprensión de sus procesos internos, las preferencias de los clientes y la lealtad de los mismos.

Análisis aumentado

El análisis aumentado proporciona Business Intelligence (y conocimientos) automatizados mediante el uso de procesamiento de lenguaje natural y aprendizaje automático. “Automatiza” la preparación de datos y permite compartir datos. El análisis aumentado proporciona resultados claros y acceso a herramientas sofisticadas, lo que permite a los investigadores y gerentes tomar decisiones diarias con un alto grado de confianza. Permite a los responsables de la toma de decisiones obtener información y actuar con rapidez y confianza.

En última instancia, análisis aumentado intenta reducir el trabajo de los científicos de datos mediante la automatización de los pasos utilizados, para obtener información e inteligencia comercial. Un motor de análisis aumentado procesará automáticamente los datos de una organización, los limpiará, los analizará y luego producirá información que conducirá a instrucciones para ejecutivos o vendedores.

Análisis de cartera

El análisis de cartera suele ser utilizado por una agencia de préstamos o un banco, y es una colección de cuentas con valores y riesgos variables. Las cuentas en cartera pueden incluir información sobre el estatus social de sus clientes (pobre, clase media, rico), su ubicación geográfica y muchos otros factores. El análisis de cartera permite al prestamista equilibrar los rendimientos de un préstamo con el riesgo de incumplimiento. El riesgo del préstamo está determinado por factores como los ingresos, el éxito de préstamos anteriores y las declaraciones de quiebra.

Analítica de recursos humanos

Originalmente llamado "análisis de personas", el análisis de recursos humanos son datos de comportamiento que se utilizan para comprender cómo trabajan las personas y cómo cambian la forma en que se administran las organizaciones. El análisis de recursos humanos también se ha denominado análisis de la fuerza laboral, análisis de talento, información de talento, información de personas, información de colegas y análisis de capital humano. El análisis de recursos humanos se utiliza para ayudar a las empresas a administrar sus recursos humanos y es una herramienta estratégica para analizar y pronosticar tendencias en los mercados laborales.

Análisis del viaje del cliente

El viaje del cliente se ocupa de la experiencia holística por la que pasan los clientes al interactuar con una organización o marca. En lugar de centrarse en una parte de la experiencia, el viaje del cliente registra la experiencia completa de un cliente.

El análisis del viaje del cliente examina la información registrada y proporciona información sobre las experiencias del cliente (a menudo en tiempo real). Ayuda a comprender al cliente e influye en cómo las empresas diseñan la experiencia del cliente. El análisis del viaje del cliente admite un método sistemático para evaluar y monitorear el viaje del cliente y mejorar el proceso. Desarrollar y brindar una experiencia óptima al cliente es el objetivo final.

8.2 Data warehouse.

Actualmente es fácil perderse cuando se lidia con datos, existen muchos tipos de datos, cada tipo tiene sus propias peculiaridades e idiosincrasias. En las organizaciones normalmente cada departamento maneja los datos a su propia manera, tienen sus propias aplicaciones, etc. Esto hace difícil que los datos de todos los departamentos se complementen entre sí. Este era el problema al que se enfrentaron muchas organizaciones anteriormente, se dieron cuenta que tener datos, no era lo mismo a tener datos creíbles, se tuvo una discusión acerca de que es “Integridad de datos”, y fue precisamente eso fue lo que hizo que el data warehouse naciera.

Según Kimbal[6], un data warehouse se puede definir una copia de los datos transaccionales, específicamente estructurados para consultas y análisis. Es decir, es el sistema que extrae, transforma y consolida los datos de los sistemas fuentes en un repositorio de datos dimensional.

[6] Ralph Kimball uno de los mayores influyentes en el diseño de modelos dimensionales.

Características de un Data Warehouse según Bill Immon

- ✓ **Orientado a temas:** los datos están organizados por temas para facilitar el entendimiento por parte de los usuarios, de forma que todos los datos relativos a un mismo elemento de la vida real queden unidos entre sí. Por ejemplo, todos los datos de un cliente pueden estar consolidados en una misma tabla, todos los datos de los productos en otra, y así sucesivamente.

- ✓ **Integrado:** los datos se deben integrar en una estructura consistente, debiendo eliminarse las inconsistencias existentes entre los diversos sistemas operacionales. La información se estructura en diversos niveles de detalle para adecuarse a las necesidades de consulta de los usuarios. Algunas de las inconsistencias más comunes que nos solemos encontrar son: en nomenclatura, en unidades de medida, en formatos de fechas, múltiples tablas con información similar.

- ✓ **Histórico (variante en el tiempo):** los datos, que pueden ir variando a lo largo del tiempo, deben quedar reflejados de forma que al ser consultados reflejen estos cambios y no se altere la realidad que había en el momento en que se almacenaron, evitando así la problemática que ocurre en los sistemas operacionales, que reflejan solamente el estado de la actividad de negocio presente. Un Data Warehouse debe almacenar los diferentes valores que toma una variable a lo largo del tiempo. Por ejemplo, si un cliente ha vivido en tres ciudades diferentes, debe almacenar el periodo que vivió en cada una de ellas y asociar los hechos (ventas, devoluciones, incidencias, etc.) que se produjeron en cada momento a la ciudad en la que vivía cuando se produjeron, y no asociar todos los hechos históricos a la ciudad en la que vive actualmente.

- ✓ **No volátil:** la información de un Data Warehouse, una vez introducida, debe ser de sólo lectura, nunca se modifica ni se elimina, y ha de ser permanente y mantenerse para futuras consultas. Por ejemplo, si en el origen se modifica la cantidad de un producto que entra en el almacén, en el Data Warehouse no podemos hacer directamente una actualización sobre ese registro sin dejar ni el más mínimo rastro de que hubo antes otro valor.

Esquema en Estrella

A la hora de modelar el Data Warehouse, hay que decidir cuál es el esquema más apropiado para obtener los resultados que queremos conseguir. Habitualmente, y salvo excepciones, se suele modelar la base de datos utilizando el esquema en estrella (star schema), en el que hay una única tabla central, la tabla de hechos, que contiene todas las medidas y una tabla adicional por cada una de las perspectivas desde las que queremos analizar dicha información, es decir por cada una de las dimensiones.

Modelado Dimensional

El Modelado Dimensional es utilizado hoy en día en la mayoría de las soluciones de BI. Es una mezcla correcta de normalización y desnormalización, comúnmente llamada Normalización Dimensional. Se utiliza tanto para el diseño de Data Marts como de Data Warehouses.

Básicamente hay dos tipos de tablas:

- Tablas de Dimensión (Dimension Tables)

- Tablas de Hechos (Fact Tables)

Tabla de hechos

Los Hechos están compuestos por los detalles del proceso de negocio a analizar, contienen datos numéricos y medidas (métricas) de Negocio a analizar. Contienen también elementos (claves externas) para contextualizar dichas medidas, como por ejemplo el producto, la fecha, el cliente, la cuenta contable, etc.

Componentes de una tabla de hechos

- ✓ **Clave principal:** identifica de forma única cada fila. Al igual que en los sistemas transaccionales toda tabla debe tener una clave principal, en una tabla de hechos puede tenerla o no, y esto tiene sus pros y sus contras, pero ambas posturas son defendibles.
- ✓ **Claves externas** (Foreign Keys): apuntan hacia las claves principales (claves subrogadas) de cada una de las dimensiones que tienen relación con dicha tabla de hechos.
- ✓ **Medidas (Measures):** representan columnas que contienen datos cuantificables, numéricos, que se pueden agregar. Por ejemplo, cantidad, importe, precio, margen, número de operaciones, etc.
- ✓ Metadatos y linaje: nos permite obtener información adicional sobre la fila, como, por ejemplo, que día se incorporó al Data Warehouse, de qué origen proviene (si tenemos varias fuentes), etc. No es necesario para el usuario de negocio, pero es interesante analizar en cada tabla de hechos qué nos aporta y si merece pena introducir algunas columnas de este tipo.

Dimensiones

Una dimensión contiene una serie de atributos o características, por las cuales podemos agrupar, rebanar o filtrar la información. A veces estos atributos están organizados en jerarquías que permiten analizar los datos de forma agrupada, dicha agrupación se realiza mediante relaciones uno a muchos (1:N). Por ejemplo, en una dimensión Fecha es fácil que encontremos una jerarquía formada por los atributos Año, Mes y Día, otra por Año, Semana y Día; en una dimensión Producto podemos encontrarnos una jerarquía formada por los atributos Categoría, Subcategoría y Producto

Tipos de claves de una dimensión

- ✓ **Una Clave subrogada (subrogate key):** es un identificador único que es asignado a cada fila de la tabla de dimensiones, en definitiva, será su clave principal. Esta clave no tiene ningún sentido a nivel de negocio, pero la necesitamos para identificar de forma única cada una de las filas. Son siempre de tipo numérico, y habitualmente también son auto-incrementales. En el caso de SQL Server recomendamos que sean de tipo INT con la propiedad identity activada (es una recomendación genérica, a la que siempre habrá excepciones).

- ✓ **Una Clave natural:** es una clave que actúa como primary key en nuestro origen de datos, y es con la que el usuario está familiarizado, pero no puede ser clave principal en nuestra tabla de dimensiones porque se podrían producir duplicidades, como veremos más adelante al explicar el concepto de Slowly Changing Dimensions.

Infraestructura de data warehouse

La arquitectura de un Data Warehouse varía según el autor, a continuación, presentamos la arquitectura de Immon, la cual es una de las más aceptables en el área de desarrollo de modelos dimensionales.

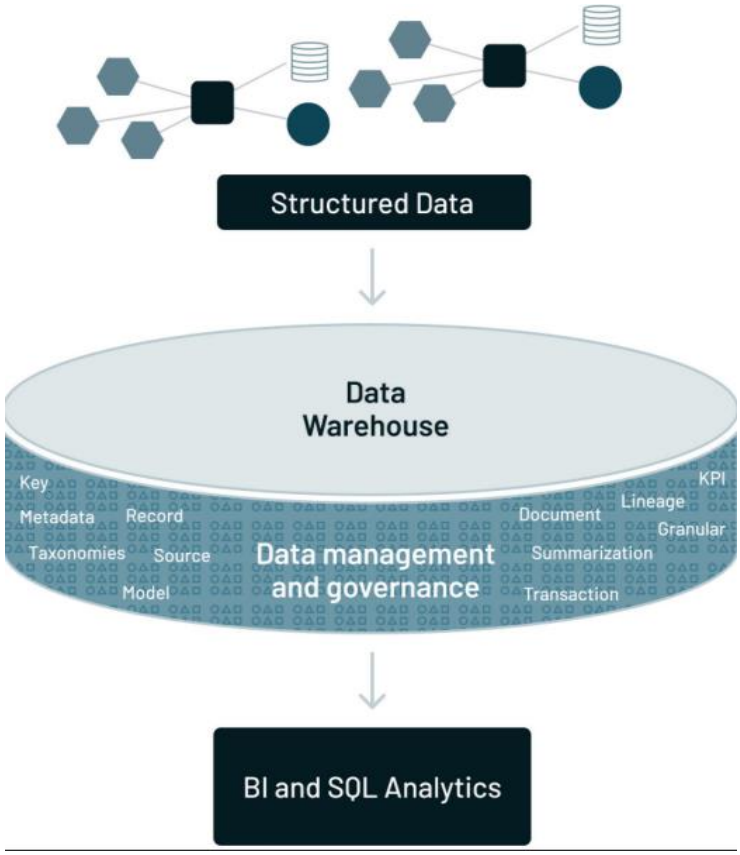

Figura 2. Infraestructura de un data warehouse

La infraestructura analítica incluye:

- ✓ **Metadatos:** una guía sobre qué donde se ubicaron que datos.
- ✓ **Modelo de datos:** una abstracción de los datos encontrados en el almacén de datos.
- ✓ **Linaje de datos**: la historia de los orígenes y transformaciones de datos encontrados en los depósitos de datos.
- ✓ **Resumen:** una descripción del algoritmo para trabajar en la creación de los datos en el almacén de datos.
- ✓ **KPI:** ¿dónde están los indicadores clave de rendimiento.
- ✓ **ETL:** tecnología que permitió que los datos de las aplicaciones se transformaren automáticamente en datos corporativos.

La limitación del Data Warehouse se vuelve evidente con el incremento de la variedad de datos (texto, IoT, imágenes, audio, video, etc.) de la organización. En adición, con el surgimiento del Machine learning (ML) e Inteligencia artificial, los nuevos algoritmos requieren acceso directo a los datos a través de lenguajes que no son SQL lo cual dificulta la extracción.

8.3 Big Data y Cloud computing.

8.3.1 Orígenes del big data

Se han realizado varios estudios sobre los puntos de vista históricos y de desarrollo en el área de análisis de Big Data. Gil Press[7] proporciono una breve historia del Big Data a partir de 1944, cubrió 68 años de historia de evolución del Big Data entre 1944 y 2012 e ilustro 32 eventos relacionados con Big Data en la historia reciente de la ciencia de datos. Como indica Press en su artículo, la línea entre el crecimiento de datos y Big Data se ha desdibujado. Muy a menudo, la tasa de crecimiento de los datos ha sido referido como 'explosión de información'; aunque a menudo ''datos'' e ''información'' son usados indistintamente, ambos términos tienen connotaciones diferentes. Los estudios realizados por Press cubren eventos hasta el 2013, sin embargo, muchos de los eventos cubren tanto a Big Data como a Ciencia de datos, por tanto, el término ''ciencia de datos'' podría ser considerado como un significado complementario al análisis de Big Data (BDA).

En comparación con la investigación de Press, Frank Ohlhorst[8] estableció el origen del Big Data a 1880 cuando se realizó el décimo censo de los Estado Unidos. El problema que se tuvo durante el siglo XIX fue un tema de estadísticas, que fue básicamente como se podía encuestar y documentar a 50 millones de ciudadanos norteamericanos. Aunque Big Data puede contener cálculos de algunos elementos estadísticos, estos dos términos tienen diferentes interpretaciones en la actualidad. Al igual que Frank Ohlhorst, hay muchos otros que coinciden con este ciclo como el origen del Big Data, argumentando que si los conjuntos de datos son tan grandes y complejos que van más allá del proceso tradicional y la capacidad de gestión, entonces este conjunto de datos puede ser considerado como Big Data.

¿Qué es Big Data?

Big data se refiere a los grandes y diversos conjuntos de información que crecen a un ritmo cada vez mayor. Abarca el volumen de información, la velocidad a la que se crea y recopila, y la variedad o alcance de los puntos de datos que se cubren (conocidos como las "tres v" de big data). Big data a menudo proviene de la minería de datos y llega en múltiples formatos.

Características claves de Big Data

- Big data es una gran cantidad de información diversa que llega en volúmenes crecientes y con una velocidad cada vez mayor.

[7] Gil Press, "A Very Short History Of Big Data," Forbes Tech Magazine, May 9, 2013. URL: http://www.forbes.com/sites/gilpress/2013/05/09/a-very-short-history-of-big-data/.
[8] Ohlhorst F. Big data analytics, turning big data into big money. Canada: John Wiley & Sons, Inc ; 2013 p. 2. p. 171.

- Los grandes datos pueden ser estructurados (a menudo numéricos, fáciles de formatear y almacenar) o no estructurados (más libres, menos cuantificables).
- Casi todos los departamentos de una empresa pueden utilizar los resultados del análisis de big data, pero manejar su desorden y ruido puede plantear problemas.
- Los grandes datos se pueden recopilar a partir de comentarios compartidos públicamente en redes sociales y sitios web, recopilados voluntariamente de dispositivos electrónicos y aplicaciones personales, a través de cuestionarios, compras de productos y registros electrónicos.
- Los grandes datos se almacenan con mayor frecuencia en bases de datos informáticas y se analizan mediante un software diseñado específicamente para manejar conjuntos de datos grandes y complejos.

Ventajas y Desventajas de Big Data

El aumento en la cantidad de datos disponibles presenta tanto oportunidades como problemas. En general, tener más datos sobre los clientes (y clientes potenciales) debería permitir a las empresas adaptar mejor los productos y los esfuerzos de marketing para crear el más alto nivel de satisfacción y repetir negocios. Las empresas que recopilan una gran cantidad de datos tienen la oportunidad de realizar análisis más profundos y ricos en beneficio de todas las partes interesadas .

Si bien un mejor análisis es positivo, los grandes datos también pueden generar sobrecarga y ruido, lo que reduce su utilidad. Las empresas deben manejar mayores volúmenes de datos y determinar qué datos representan señales en comparación con el ruido. Decidir qué hace que los datos sean relevantes se convierte en un factor clave.

Además, la naturaleza y el formato de los datos pueden requerir un manejo especial antes de que se tomen medidas al respecto. Los datos estructurados, que consisten en valores numéricos, se pueden almacenar y clasificar fácilmente. Los datos no estructurados, como correos electrónicos, videos y documentos de texto, pueden requerir la aplicación de técnicas más sofisticadas antes de que sean útiles.

8.3.2 Cloud computing

Cloud computing o computación en la nube son servicios de recursos computacionales distribuidos a través de la red. Dichos recursos computacionales son presentados como uno o muchos recursos unificados, esto según sea el acuerdo del consumidor con el proveedor de servicios.

Modelo de servicio ofrecidos en Cloud Computing

Los modelos de servicios más comunes ofrecidos por proveedores de cloud computing son:

1. Software As a Service (SaaS):
   En este modelo las aplicaciones son accedidas bajo demanda, el consumidor no gestiona ni controla la infraestructura de nube, servidores, sistemas operativos o almacenamiento. Un ejemplo de estos servicios son los correos electrónicos.

2. Platform As a Service (PaaS):
   En este modelo el consumidor puede desplegar sus propias aplicaciones, dichas aplicaciones tienen que ser desarrolladas utilizando lenguajes y herramientas de programación soportadas por el proveedor. El consumidor no gestiona ni controla la infraestructura de nube, servidores,

sistemas operativos o almacenamiento, pero tiene control sobre las aplicaciones desplegadas y la posibilidad de controlar las configuraciones de entorno del hosting.

3. Infraestructure As a Service (IaaS):
   En este modelo el consumidor puede seleccionar procesamiento, almacenamiento y otros recursos computacionales fundamentales, de forma que el consumidor pueda desplegar y ejecutar software arbitrario, que puede incluir sistemas operativos y aplicaciones. El consumidor no gestiona ni controla la infraestructura de nube pero
   tiene control sobre los sistemas operativos, almacenamiento, aplicaciones desplegadas y la posibilidad de tener un control limitado de componentes de red.

Modelos de implementación de Cloud Computing

Con independencia del modelo de servicio utilizado (SaaS, PaaS, IaaS,) hay cuatro formas principales en los que se despliegan los servicios de Cloud Computing:

1. Nube publica:
   Los proveedores implementan los servicios en su propia infraestructura y los ponen a disposición del público en general. Los principales desafíos de este modelo están relacionados con la seguridad de la información y la calidad del servicio.

2. Nube privada:
   La infraestructura de la nube la gestiona una organización para suministrar los servicios de TI a sus usuarios internos. Como ventajas el data center se hace más ágil y flexible y se obtiene un mejor manejo de los recursos, pero como desventaja se pierde la escalabilidad de la nube ya que está limitada por los recursos físicos disponibles.

3. Nube hibrida:
   Este modelo consiste en complementar una nube privada con los servicios de una nube publica, obteniendo las ventajas de los dos modelos.

Data Lake

Un Data Lake es un repositorio digital construido para almacenar una gran cantidad de datos en formato original, es decir que su estructura no se modifica, la fuente información para alimentar un Data Lake es muy variada, comprende sistemas transaccionales, sistemas gerenciales, logs, correos electrónicos, Internet de las cosas, etc.

Un Data Lake puede ser usados para múltiples propósitos, a continuación, se presentan un resumen de los comunes:

- Ingestión de datos multi estructurados, semiestructurados, y no estructurados.
- Plataforma de ETLs, preparando y creando perfiles para sistemas de almacenamientos, así las organizaciones no se vean obligadas a expandir sus datawarehouse existentes.
- Mas que un Datawarehouse, ya que un Data Lake permite almacenar datos que no son fáciles de tratar por un Datawarehouse tradicional.
- Archivado y almacenamiento histórico de datos de toda la organización
- Almacenamiento de la información que generan múltiples dispositivos en el internet de las cosas (IOT) que luego serán analizados, por ejemplo, aplicando Machine Learning (ML).

Dado que un Data Lake se puede dedicar para múltiples propósitos, la división interna de directorios dependerá del objetivo que se persigue.

Un Data Lake puede almacenar datos estructurados, semi estructurados y no estructurados.

- **Datos estructurados:** La mayoría de datos estructurados de una organización son creados día a día en sus transacciones, estos datos son escritos en bases de datos SQL, cuando una transacción es ejecutada, una característica importante de estos datos es que cada nuevo dato tiene una estructura similar que el anterior.
- **Datos semi estructurados:** Son datos que no tienen un esquema definido, algunos de estos tipos de datos son formatos **XLM, JSON, TEXT**, estos datos son guardados en bases de datos NoSQL.
- **Datos no estructurados:** Son datos que no siguen reglas definidas, y cada formato de dato conlleva diversos grados de complejidad para analizarlo, en comparación con los anteriores tipos de datos. Entre algunos datos no estructurados tenemos: Imágenes, audio, video.

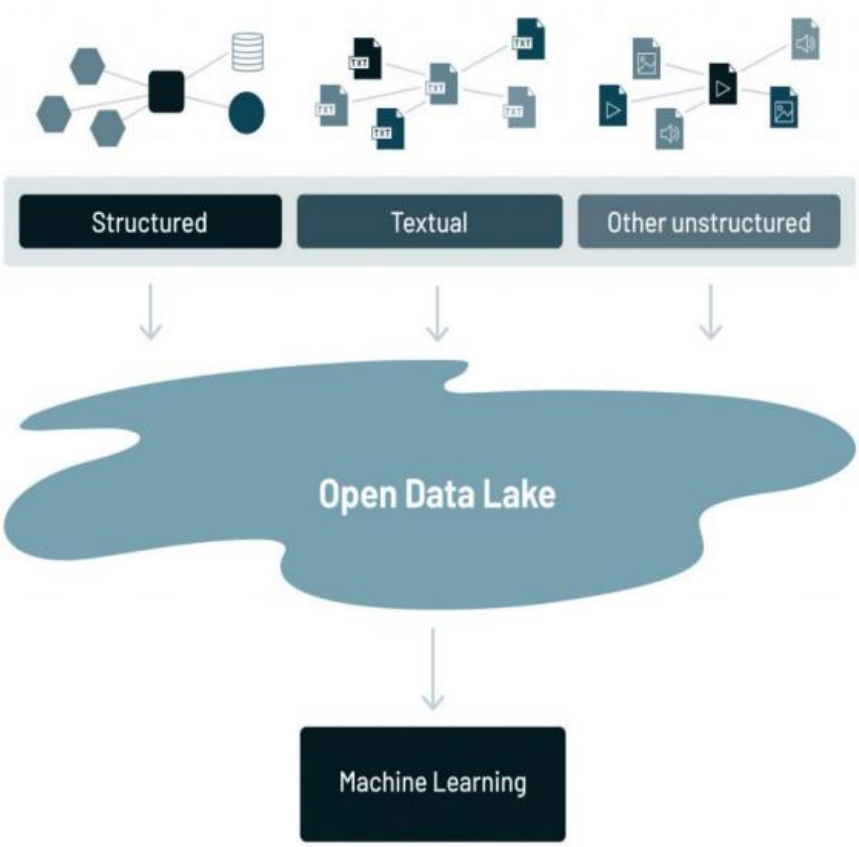

Figura 3. Data Lake aplicado a machine learning

Data Lakehouse

Un Data Lakehouse es una nueva arquitectura de gestión de datos que combina los mejores elementos de los Data Lakes y los Data Warehouses. Toma la flexibilidad, bajo costo y escalabilidad de los Data Lakes, y la gestión de datos con los principios ACID de los Data Warehouse, lo que permite procesos de BI y de Machine Learning para toda la data.

Los Data Lakehuse permiten utilizar las características propias de los DW de manejar data estructurada y gestión de datos directamente en el almacenamiento de bajo costo de los Data Lakes. Teniendo estas características juntas en un mismo sistema, significa que los equipos de datos puedan moverse rápidamente y ser capaces de usar la data sin necesidad de acceder a multiples sistemas. Esto asegura que los datos sean los más completos, actualizados y disponibles para proyectos de Data Science, Machine learning y Business analytics.

Como los Data Lakes, los Data Lakehouse soportan entradas de datos estructurados, semi estructurados y no estructurados. Y permite la gestión o gobernanza de datos con los datos ya procesados, para luego ser consumidos para diferentes aplicaciones, como BI, ML, Data Science, etc.

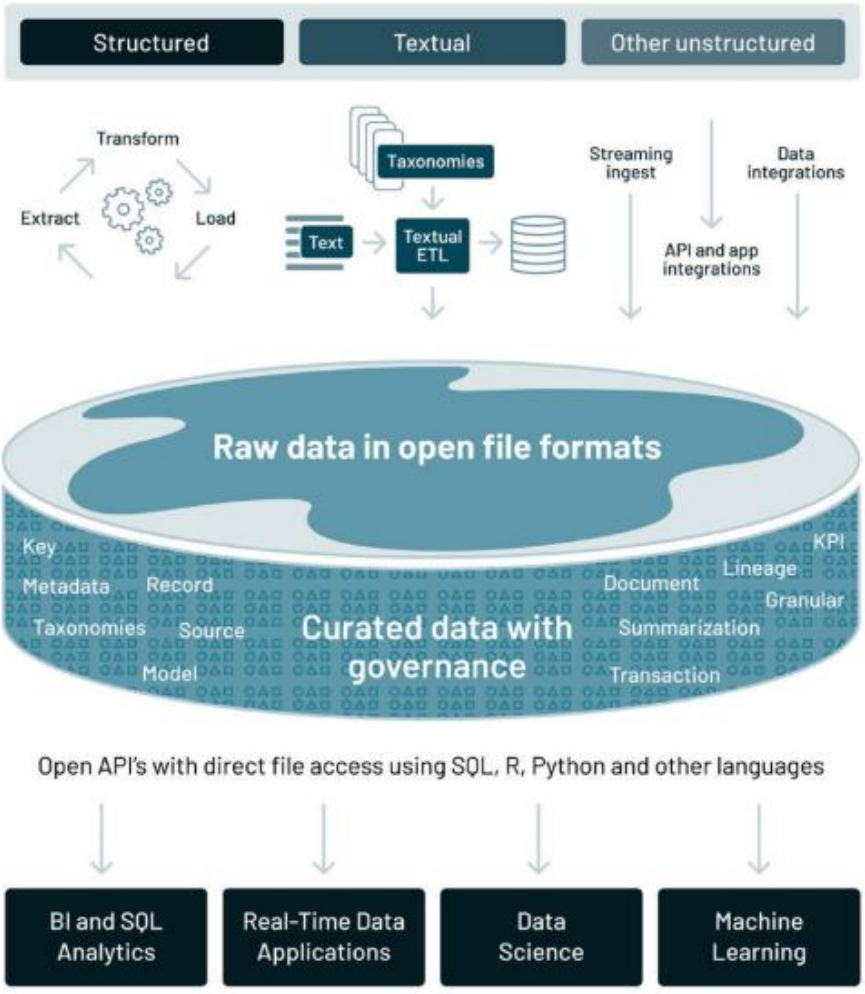

Figura 4. Estructura de una data lake house para el consumo de varias aplicaciones

Comparativa entre un Data Warehouse, Data Lake y Data Lakehouse

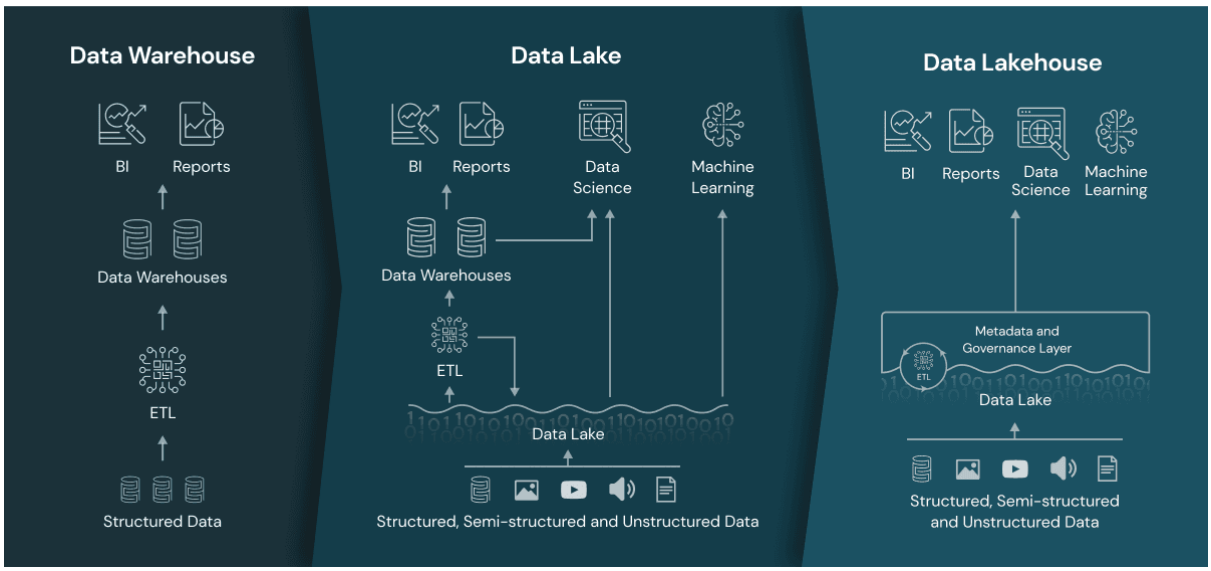

Figura 5. Comparativa entre un Data Warehouse, Data Lake y Data Lakehouse

	Data Warehouse	**Data Lake**	**Data Lakehouse**
Tipos de datos	Solo datos estructurados	Datos estructurados, semi estructurados y no estructurados	Datos estructurados, semi estructurados y no estructurados
Costo	Muy costoso	Bajo costo	Bajo costo
Escalabilidad	Fácil de escalar, pero los costos incrementan exponencialmente.	Fácil de escalar y manteniendo costos bajos	Fácil de escalar y manteniendo costos bajos
Usuarios objetivos	Analista de datos	Científico de datos	Analista de datos, científico de datos, Ingenieros de Machine Learning
Fiabilidad	Alta calidad, datos de confianza	Baja calidad	Alta calidad, datos de confianza
Usabilidad	Simple: La estructura de un DW permite a los usuarios un manejo fácil y rápido de los datos para reportería y análisis.	Difícil: Explorar grandes cantidades de datos puede ser muy difícil si no se cuenta con herramientas para organizar y catalogar los datos	Simple: Provee la simplicidad de un DW y además sus datos pueden ser consumidos para más casos de uso.
Rendimiento	Alto	Bajo	Alto

Tabla 1. Comparacion entre un Data Warehouse , Data Lake y Data Lakehouse

8.4 Ciclo de vida de un proyecto de DW.

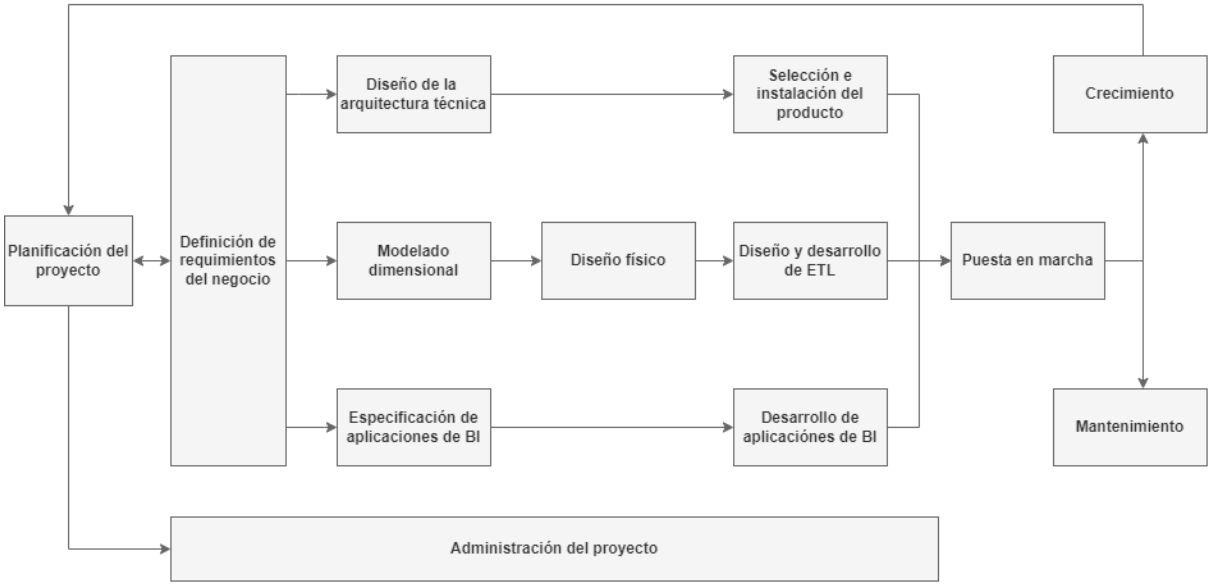

Figura 6. Ciclo de vida de un proyecto de Datawarehouse

8.4.1 Descripción de sus elementos.

8.4.1.1 Planeación del proyecto

En este proceso se determina el propósito del proyecto de DW/BI, sus objetivos específicos y el alcance del mismo, los principales riesgos y una aproximación inicial a las necesidades de información.

En la visión de programas y proyectos de Kimball, Proyecto, se refiere a una iteración simple del KLC (Kimball Life Cycle), desde el lanzamiento hasta el despliegue, esta tarea incluye las siguientes acciones típicas de un plan de proyecto:

1. Definir el alcance (entender los requerimientos del negocio).
2. Identificar las tareas
3. Programar las tareas
4. Planificar el uso de los recursos.
5. Asignar la carga de trabajo a los recursos
6. Elaboración de un documento final que representa un plan del
7. Proyecto.

Además, en esta parte definimos cómo realizar la administración o gestión de esta subfase que es todo un proyecto en sí mismo, con las siguientes actividades:

1. Monitoreo del estado de los procesos y actividades.
2. Rastreo de problemas

3. Desarrollo de un plan de comunicación comprensiva que direccione la empresa y las áreas de TI

8.4.1.2 Definición de requerimiento del negocio

Antes de empezar con cualquier esfuerzo del diseño del modelo dimensional, es necesario entender las necesidades del negocio, así que como la realidad de las fuentes de datos existentes. Es necesario obtener los requerimientos de la manera más clara posible a través de reuniones con los representantes del negocio.

Existen cuatro decisiones clave de diseño que deberán ser tomadas en conjunto con los representantes del negocio, estas serán:

1. Selección del proceso de negocio
2. La granularidad
3. Identificación de dimensiones
4. Identificaciones de métricas.

Las respuestas a dichas preguntas serán determinadas considerando las necesidades del negocio junto con la realidad de los datos que se tienen disponible. Una vez establecido el proceso de negocio, granularidad, dimensiones y métricas que se necesitaran, el equipo de desarrollo puede empezar con el diseño del modelo dimensional y su respectiva implementación.

8.4.1.3 Diseño de la arquitectura técnica

Mientras que la definición de los requerimientos del negocio responde la pregunta de ¿Qué necesitamos para hacerlo?, el diseño de la arquitectura responde a la pregunta **¿Cómo lo haremos?**

La arquitectura técnica es el plan general para que el Data Warehouse esté listo para cuando sea implementado. Esto describe el flujo de datos desde los sistemas de fuente de información hasta los tomadores de decisiones pasando por las transformaciones y almacenamiento de los datos.

En este paso también se especifica las herramientas, técnicas, utilidades, y plataformas necesarias para hacer que el flujo de datos fluya a través de DW.

Típicamente una arquitectura de un DW consiste en cuatro servidores: Un servidor de ETL, un servidor de bases de datos, un servidor OLAP, y un servidor de reportes.

8.4.1.4 Selección del producto e instalación

Es similar a una lista de compras para seleccionar productos que encajen en el marco del plan. Las siguientes seis tareas asociadas con DW/BI con respecto a la selección de productos son bastante similares a cualquier selección de tecnología.

a. Comprender el proceso de compras corporativas.

El primer paso antes de seleccionar nuevos productos es entender el hardware interno y procesos de compra de software.

b. Desarrollar una matriz de evaluación de productos.

Usando el plan de arquitectura como punto de partida, una evaluación basada en una hoja de cálculo se debe desarrollar una matriz que identifique los criterios de evaluación, junto con la ponderación factores para indicar importancia; cuanto más específicos sean los criterios, mejor.

Si los criterios son demasiado vagos o genéricos, todos los proveedores dirán que pueden satisfacer sus necesidades.

c. Realizar estudios de mercado.

Para convertirse en compradores informados al seleccionar productos, debe hacer una investigación de mercado para comprender mejor a los jugadores y sus ofertas. Una solicitud de propuesta (RFP) es una herramienta clásica de evaluación de productos.

Aunque algunas organizaciones no tienen elección sobre su uso, debe evitar esta técnica, si es posible. Construyendo la RFP y evaluar las respuestas consume mucho tiempo para el equipo.

Mientras tanto, los proveedores están motivados para responder a las preguntas de la manera más positiva, por lo que la evaluación de la respuesta suele ser más un concurso de belleza. Al final, el valor del gasto puede no justificar el esfuerzo.

d. Evaluar una lista corta de opciones.

A pesar de la plétora de productos disponibles en el mercado, por lo general sólo un pequeño número de los proveedores pueden cumplir con los requisitos técnicos y de funcionalidad. Por comparación de puntuaciones preliminares de la matriz de evaluación, puede centrarse en una lista limitada de vendedores y descalificar al resto. Después de tratar con un número limitado de proveedores, usted puede comenzar las evaluaciones detalladas.

Los representantes comerciales deben participar en este proceso si está evaluando herramientas de BI. Como evaluadores, deben impulsar el proceso en lugar de permitir que los vendedores conduzcan, compartiendo información relevante de el plan de arquitectura, por lo que las sesiones se centran en sus necesidades en lugar de en el producto.

Asegúrese de hablar con las referencias de los proveedores, tanto las proporcionadas formalmente y los obtenidos de su red informal.

e. Si es necesario, realice un prototipo.

Después de realizar las evaluaciones detalladas, a veces un claro ganador burbujea al superior, a menudo basado en la experiencia o relaciones previas del equipo. En otros casos, el líder emerge debido a los compromisos corporativos existentes, tales como licencias de sitios o compras de hardware heredado. En cualquier situación, cuando surge un único candidato como el ganador, puede omitir el paso del prototipo (y la inversión asociada en ambos tiempo y dinero).

Si ningún proveedor es el aparente ganador, debe realizar un prototipo con no más de dos productos. Una vez más, hágase cargo del proceso desarrollando un estudio de caso de negocio limitado pero realista.

f. Seleccionar producto, instalar en prueba y negociar

Es hora de seleccionar un producto. En lugar de firmar inmediatamente en la línea punteada, preservar su poder de negociación haciendo un compromiso privado, no público, con un vendedor único.

En lugar de informarle al vendedor que está completamente vendido, emprenda en un período de prueba en el que tiene la oportunidad de poner el producto en uso real en tu entorno Se necesita mucha energía para instalar un producto, capacitarse y comenzar a usarlo, por lo que debe seguir este camino solo con el proveedor que tiene completamente la intención de comprar; una prueba no debe llevarse a cabo como otro ejercicio de patear neumáticos.

A medida que la prueba llega a su fin, tiene la oportunidad de negociar una compra que es beneficioso para todas las partes involucradas.

8.4.1.5 Desarrollo de data profiling

Una vez obtenidos los requerimientos por parte del negocio, se debe de realizar una revisión de las fuentes de datos que se tienen y que darán soporte a los requerimientos. La mejor forma de lograrlo es a través de un data profiling, ello ayudara a incrementar el entendimiento de la estructura de la fuente de datos, su contenido, relaciones, y las reglas implícitas de los datos. Es necesario verificar que los datos existen y que se encuentran en un estado que se puedan utilizar. El data profiling podrá ser tan fácil como escribir unas consultas SQL o tan complejo como utilizar una herramienta específica para ello.

Los resultados del data profiling son documentados, normalmente se crea una lista de los elementos de datos que cumplen con una calidad aceptable, estos datos serán los que se utilizaran luego en los procesos ETL.

8.4.1.6 Modelado dimensional

8.4.1.6.1 Características del modelado dimensional.

El modelado dimensional es una técnica de diseño lógico, esta técnica nos proveerá de una estructura de los datos que tengan las siguientes características:

- Un alto rendimiento en la consulta de datos
- Intuitivos para los usuarios del negocio.

Los modelos dimensionales son llamados modelos o esquemas estrellas, estos modelos son guardados en estructuras OLAP conocidas como Cubos OLAP.

Proceso iterativo para crear un modelo dimensional:

1. **Elegir el proceso del negocio:**
   Este proceso depende del análisis de requerimientos del negocio.
2. **Establecer el nivel de granularidad:**
   El nivel de granularidad depende del nivel de detalle que se quiera obtener, se recomienda buscar el nivel de detalle más profundo que permitan los datos, tomando en cuenta lo que se necesite según diga el análisis de requerimientos del negocio.
3. **Elegir dimensiones:**
   Las dimensiones tienen generalmente atributos textuales que nos brindaran el contexto para el nivel de granularidad escogido. Para identificar las dimensiones se deben analizar sus atributos a fin de encontrar atributos candidatos a ser encabezados de informes, cubos o cualquier tipo de visualización unidimensional o multidimensional.
4. **Identificar medidas y tablas de hechos:**
   Las medidas o métricas son aquellos valores que surgen en los procesos de negocios, una medida o métrica es un atributo que se desea analizar, sumarizando o agrupando los datos. Estas medidas o métricas estarán almacenadas en las tablas de hechos, estas tablas están relacionadas con las dimensiones que proveen del contexto de las medidas.

8.4.1.6.2 Dimensiones.

Las tablas de dimensiones son complementos integrales de una Fact table. Las tablas de dimensiones contienen el contexto textual asociado con un evento de medición del proceso de negocio.

Con las dimensiones se busca describir el "quién, qué, dónde, cuándo, cómo y por qué" asociado con el evento. las tablas de dimensiones a menudo tienen muchas columnas o atributos. No es raro que una tabla de dimensiones tenga de 50 a 100 atributos, aunque, naturalmente, algunas tablas de dimensiones solo tienen un puñado de atributos.

Las tablas de dimensiones tienden a tener menos filas que las tablas de hechos, pero pueden ser anchas con muchas columnas de texto grandes. Cada dimensión está definida por una sola clave principal. Los atributos de dimensión sirven como fuente principal de restricciones de consulta, agrupaciones, y etiquetas de informe. En una consulta o solicitud de informe, los atributos se identifican como el por palabras.

Los atributos de la tabla de dimensiones juegan un papel vital en el sistema DW/BI. Porque ellas son la fuente de prácticamente todas las restricciones y etiquetas de informe, los atributos de dimensión son fundamental para hacer que el sistema DW/BI sea utilizable y comprensible. Los atributos deben de ser construidos para representar completamente datos textuales que evidencien contexto, debemos de procurar minimizar el uso de códigos en tablas de dimensiones reemplazándolos con más detallados o más textuales.

La figura 5 muestra que las tablas de dimensiones a menudo representan relaciones jerárquicas. Por ejemplo, los productos se agrupan en marcas y luego en categorías. Para cada fila en la dimensión del producto, debe almacenar la marca y la categoría asociadas descripción.

En vez de buscar la tercera forma normal, las tablas de dimensiones suelen estar muy desnormalizadas con aplanado de relaciones de muchos a uno dentro de una sola tabla de dimensiones. Porque las tablas de dimensiones por lo general son geométricamente más pequeñas que las tablas de hechos o Fact tables, lo que mejora la eficiencia del almacenamiento

a la normalización o la creación de copos de nieve prácticamente no tienen impacto en el tamaño total de la base de datos. Debemos casi siempre de sacrificar el espacio de la tabla de dimensiones por simplicidad y accesibilidad.

Product Key	Product Description	Brand Name	Category Name
1	PowerAll 20 oz	PowerClean	All Purpose Cleaner
2	PowerAll 32 oz	PowerClean	All Purpose Cleaner
3	PowerAll 48 oz	PowerClean	All Purpose Cleaner
4	PowerAll 64 oz	PowerClean	All Purpose Cleaner
5	ZipAll 20 oz	Zippy	All Purpose Cleaner
6	ZipAll 32 oz	Zippy	All Purpose Cleaner
7	ZipAll 48 oz	Zippy	All Purpose Cleaner
8	Shiny 20 oz	Clean Fast	Glass Cleaner
9	Shiny 32 oz	Clean Fast	Glass Cleaner
10	ZipGlass 20 oz	Zippy	Glass Cleaner
11	ZipGlass 32 oz	Zippy	Glass Cleaner

Figura 7. Ejemplo de una tabla de dimension de un modelo dimensional

Las dimensiones mayormente implementadas son las del tipo conformadas , estas dimensiones se relacionan con más de una Fact table , es decir que alimentan de contexto no solo a un proceso de negocio, sino que a varios, esto permite optimizar el proceso de modelado , optimizar el uso de storage , facilitar la construcción de los modelos , por ejemplo en empresas con un buen rodaje en el desarrollo de soluciones de Big data , las mismas ya cuentan con un buena cantidad de modelos implementados , lo que permite tener un banco de elementos del modelo dimensional , cuando se inicien nuevos ciclos de desarrollo , en este camino del modelado dimensiona vamos a poder evaluar si alguna de las dimensiones existentes pueden reutilizarse es decir utilizar dimensiones conformadas que pertenecen a otros modelos de negocio y que serán de gran ayuda para optimizar el trabajo de diseñado.

En una vista de modelo , solemos visualizarlas relacionadas con más de una Fact table lo que permite a la organización optimizar sus tiempos , de tal manera que estas pueden ser sometidas a modificaciones en las que se les añadan más campos contextuales , viéndose entonces beneficiado el modelo a construir y además en segundo plano al modelo del cual estamos reutilizando la dimensión, porque este puede en un momento dado necesitar de esas nuevas informaciones para satisfacer los análisis que se presenten.

Las dimensiones también están sujetas a cambios en el tiempo es por ello que los propietarios de los datos y el correspondiente equipo deben de establecer una serie de estrategias para poder determinar cómo se manejaran los cambios en las dimensiones , estas estrategias son definidas para cada una de las tablas , el objetivo de crear las estrategias es para poder tener un panorama sobre el impacto que estos cambios en los campos de las dimensiones provocaran en el modelo dimensional construido, a continuación se mostraran los tipos de dimensiones SCD que existen :

a. SCD tipo 0, para valores estáticos

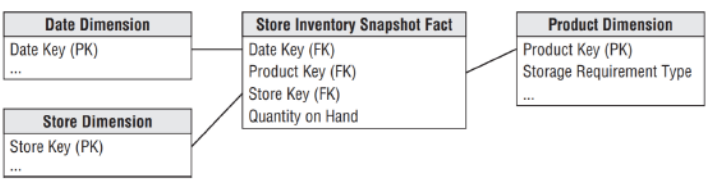

Figura 8. Ejemplo de dimensión tipo SCD 0

Los valores de los atributos para estas dimensiones no suelen cambiar nunca, las métricas son siempre agrupadas por el valor original, son válidas para dimensiones que son fijas en el tiempo, por lo general no necesitan ETL.

b. SCD tipo 1, para valores sobrescritos

Original row in Product dimension:

Product Key	SKU (NK)	Product Description	Department Name
12345	ABC922-Z	IntelliKidz	Education

Updated row in Product dimension:

Product Key	SKU (NK)	Product Description	Department Name
12345	ABC922-Z	IntelliKidz	Strategy

Figura 9. Ejemplo de dimensión tipo SCD 1

Los valores de los atributos para estas dimensiones pueden variar en el tiempo , este tipo de dimensiones siempre reflejan el estado más actual de los atributos , el proceso ETL se encargara de sobrescribir los valores que han cambiado , por lo que no se ve en la necesidad de actualizar keys de dimensiones y Fact tables , entre los aspectos a mencionar en cuanto a implicaciones podemos mencionar que con este tipo de dimensiones se suele perder historia de cualquier cambio , los reportes y métricas pueden variar antes las actualizaciones.

c. SCD tipo 2, para la adición de una nueva fila

Original row in Product dimension:

Product Key	SKU (NK)	Product Description	Department Name	...	Row Effective Date	Row Expiration Date	Current Row Indicator
12345	ABC922-Z	IntelliKidz	Education	...	2012-01-01	9999-12-31	Current

Rows in Product dimension following department reassignment:

Product Key	SKU (NK)	Product Description	Department Name	...	Row Effective Date	Row Expiration Date	Current Row Indicator
12345	ABC922-Z	IntelliKidz	Education	...	2012-01-01	2013-01-31	Expired
25984	ABC922-Z	IntelliKidz	Strategy	...	2013-02-01	9999-12-31	Current

Figura 10. Ejemplo de dimensión tipo SCD 2

Los valores de los atributos para estas dimensiones representan correctamente la historia de los datos, cada vez que una fila de una dimensión es actualizada, una nueva fila es creada conteniendo la última versión de los valores, cada una de las filas tiene una versión de elemento de la dimensión que fue validada en un periodo de tiempo específico.

d. SCD tipo 3, para la adición de un nuevo atributo

Updated row in Product dimension:

Product Key	SKU (NK)	Product Description	Current Department Name	2012 Department Name	2011 Department Name
12345	ABC922-Z	IntelliKidz	Strategy	Education	Not Applicable

Figura 11. Ejemplo de dimensión tipo SCD 3

El uso de este tipo de dimensión resulta ser necesario si se está interesado en analizar datos como si los atributos de la dimensión no hubieran cambiado como una realidad alterna, para soportar este tipo de análisis una SCD 3 añade un nuevo campo para mantener la historia, mientras el atributo principal es actualizado, el atributo guarda la historia, es válido el uso de esta dimensión cuando los sus atributos son ajustados periódicamente.

8.4.1.6.3 Fact tables.

Una fact table contiene medidas numéricas producidas por un evento de medición operacional en el mundo real. A nivel más simple, una fila de una fact table corresponde a una medición de un evento. En adición a las medidas que también son llamadas métricas, una fact table también contiene llaves foráneas que corresponden a las dimensiones asociadas, así como llaves degeneradas. Entre los tipos de Fact table que existen podemos mencionar:

a. Fact Table Transaccional

Una fila de una fact table transaccional corresponde a una medición de un evento en un punto en un espacio y tiempo. La granularidad de la fact table transaccional debe ser lo más atómica posible, esto permitirá cálculos más robustos y una mayor explotación de los datos. Una fact table transaccional tendrá registros solo si mediciones de eventos se llevan a cabo, estos deben de ser consistentes con el nivel de granularidad seleccionado.

b. Fact Table de Snapshot periódicos

Una fila en una fact table de snapshot periódico sumariza muchas mediciones de eventos que ocurren sobre un periodo estándar, como un día, una semana, o un mes. La granularidad es el periodo y no la transacción individual. Es algo común que las fact table de snapshot periódico contentan muchas métricas ya que cualquier medición del evento consistente con el nivel de granularidad es permitida.

c. Fact Table de Snapshot acumulativos

Una fila en una fact table de snapshot acumulativo sumariza las mediciones de eventos que ocurrieron en ciertos pasos predecibles que están entre el inicio y final de un proceso. El flujo de un proceso, como procesar una orden o un proceso de un reclamo, los cuales tienen puntos de inicio definidos, puntos intermedios y puntos de finalización definidos pueden ser modelados con este tipo de fact table.

8.4.1.6.4 Modelo estrella.

Un modelo estrella está conformado por varias dimensiones relacionadas con una fact table, por lo cual un modelo estrella representa un proceso de negocio. Un modelo estrella tiene las siguientes características:

- Una fact table que contiene métricas del proceso de negocio rodeado de dimensiones que proveen del contexto de cuando sucedió el evento.
- Facil de comprender por los usuarios del negocio
- Simplicidad y simetría
- Incrementa el rendimiento
- Altamente extendible con nuevas dimensiones y nuevas métricas

- No es construido para una consulta en especifico

En un Data Warehouse habrá tantos modelos estrellas como procesos de negocio se analicen, y las dimensiones que conforman los modelos estrellas son compartidos por las fact tables.

8.4.1.6.5 Comparación entre una dimensión y una tabla de bases de datos relacional.

Dimensión	Tabla de base de datos relacional
Son completamente desnormalizadas	Siempre se buscan que las tablas estén lo suficientemente normalizadas en su la tercera forma normal.
Proveen contexto textual del evento que la Fact table está registrando.	Resulta ser difícil que una tabla relacional evidencia información contextual ya que están construidas con categorías.
Son extensas a nivel de columnas y pequeñas a nivel de fila.	Debido a la normalización con la que son construidas, estas suelen presentarse como catálogos, pero con la diferencia de que están llenas de columnas banderas, flags.
Construidas con la inexistencia de columnas que representen códigos, flags, banderas provenientes de sistemas transaccionales.	Incluyen campos de tipo código, flags, banderas representativas de los procesos de negocio de las organizaciones.
Aplana las relaciones de muchos a muchos en una sola tabla.	Representa las relaciones de muchos a muchos en varias tablas.
Son construidas, de tal manera que los datos se almacenen con la suficiente limpieza, desnormalización y estandarización.	Son construidas tal cual como los sistemas transaccionales almacenan los datos, teniendo estos datos sucios, pocos estandarizados.

Tabla 2. Comparacion entre una dimension y una tabla de base de datos relacional

8.4.1.7 Diseño físico.

Los modelos dimensionales desarrollados y documentados a través de una fuente a una estructura preliminar deben de ser traducidos dentro de una base de datos física. Con los modelos dimensionales los diseños tanto de modelos lógicos y físicos guardan una gran semejanza, sin embargo, la estructura del modelo estrella no debe de ser cambiada durante el proceso del diseño físico del modelo dimensional.

Estándares de base de datos

Nombres de columnas y tablas son elementos clave para la experiencia del usuario, ambos tanto para la navegación en el modelo de datos y las visualizaciones en las aplicaciones BI, por lo tanto, estos deben de ser lo más significativo al negocio. También se deben establecer estándares sobre llaves y la permisidad de datos nulos.

Desarrollo del modelo físico

Este modelo debe de ser construido inicialmente durante el desarrollo donde será usado para la construcción de los ETL's. Habrán muchas tablas adicionales que deberán ser diseñadas y deployadas como parte del sistema DW/BI, incluyendo las tablas en las capas de row y staging.

8.4.1.8 Diseño y desarrollo de etl's

8.4.1.8.1 Definicion de etl's.

ETL (extract, transform, and load): Extracción, transformación y carga, un ETL es un proceso que permite a las organizaciones extraer datos de múltiples fuentes luego aplicar diversas transformaciones según lo que se necesite para posteriormente cargar estos datos transformados a nuevas bases de datos, Data Mart o Data Warehouse.

8.4.1.8.2 Herramientas en el mercado.

En el mercado existen una inmejorable cantidad de herramientas para la implementación de ETL que dependiendo de las características de los ETL , podemos elegir una herramienta u otra , es importante mencionar que algunos de los factores que debemos de tomar en cuenta a la hora de elegir la herramienta es inicialmente conocer las posibilidades técnicas de cada herramienta ya que nos permite de manera directa entender si la herramienta será lo suficiente potente y capaz para implementar las cosas que deseamos construir.

Un segundo factor es la consideración de los conocimientos del equipo de trabajo es decir tomar en cuenta las opiniones en cuanto a la experiencia que puede tener algunos de los miembros del equipo con el uso de alguna herramienta y el tercer factor es la consideración de los precios existentes, así como también el respectivo licenciamiento que cada una de las herramientas posee, a continuación, se presentan algunas de las herramientas más populares en el mercado.

Figura 12. Principales herramientas para la implementacion de ETL

8.4.1.8.3 Carga Full.

La carga full difiere significativamente de la carga incremental, esta se ejecuta una única vez al inicio de la vida del modelo, incluye datos desde una fecha establecida en un SLA hasta T – 1, donde T puede ser una hora, día, semana, etc. La mayor preocupación durante la carga full es el volumen de datos, a veces miles de veces mayor a una carga incremental diaria. En ocasiones la carga full puede durar varios días, lo cual es usualmente tolerable.

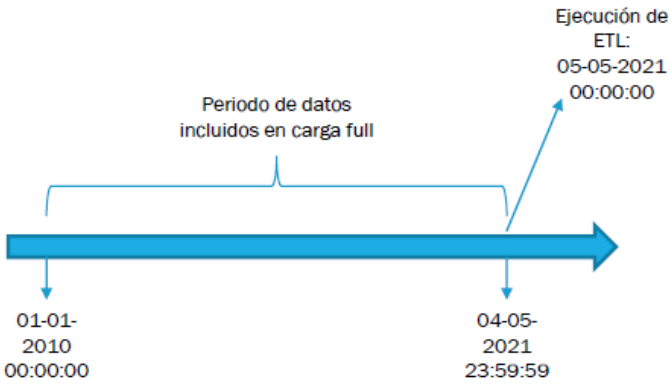

Figura 13. Ejemplo Carga Full

8.4.1.8.4 Carga Incremental.

La carga incremental toma como base una carga full, luego se define un periodo de tiempo entre cada carga incremental. En cada ejecución de una carga incremental se toman los datos que han sido creados o modificados desde la última carga incremental y los actualiza en sus respectivas dimensiones o fact tables.

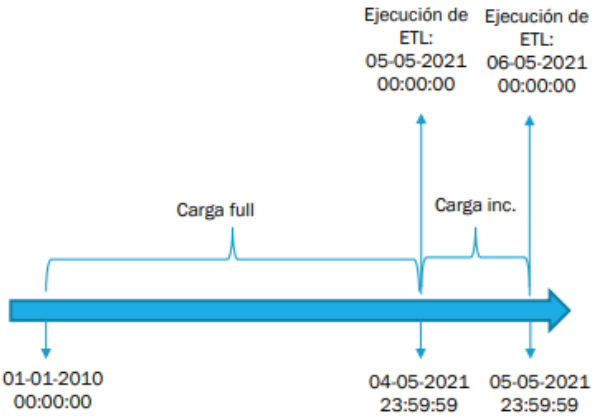

Figura 14. Ejemplo Carga Incremental

8.4.1.9 Elección de herramienta de reportes.

En el mercado existen un sinfín de herramientas que nos permitirán implementar visualizaciones para la construcción de los reportes. Pero uno de los factores ligeros que podemos tomar en cuenta es la posibilidad de que la herramienta nos permite implementar las cosas que deseamos construir, por ejemplo, si el modelo dimensional se alimenta de fuentes de datos generados en tiempo real, tendremos que buscar herramientas que sean los suficientemente potentes para poder implementar las visualizaciones con ese tipo de fuente de datos.

Pero el factor potencial que nos permitirá tomar una decisión con respecto a una herramienta es esa versatilidad que las herramientas pueden tener con respecto a sus capacidades de integrarse a diferentes tipos de fuentes de datos en diferentes tecnologías. Este es un factor elemental porque permite que las herramientas se adapten a diferentes ecosistemas de productos de diferentes empresas en el mercado, a continuación, se presentaran en la siguiente figura una serie de herramientas populares en el mercado para la construcción de reportes.

Figura 15. Principales herramientas para la construccion de dashboard

8.4.1.10 Desarrollo de dashboard en la herramienta seleccionada.

Una vez completado el desarrollo del modelo dimensional y se ha seleccionado la herramienta de BI a utilizar, se procede a crear la conexión entre la fuente de datos y la herramienta de BI, una vez terminado todas las configuraciones se puede empezar a crear las visualizaciones necesarias para responder a las necesidades del negocio.

Todas las opciones de análisis disponible en el dashboard dependerán de la herramienta BI seleccionada, es por ello que se recomienda que la herramienta seleccionada sea de acuerdo a las necesidades que se pretender cubrir.

8.4.1.11 Puesta en marcha del proyecto de DW.

En esta etapa es donde convergen los caminos de BI, datos y tecnología a utilizar en el Data Warehouse, Esta es una etapa crítica donde todo lo anterior debe haberse preparado con éxito para que la puesta en marcha sea satisfactoria.

Para que una puesta en marcha sea exitosa cada etapa anterior debe ser probada para que salgan a la luz posibles problemas que serán solventados en sus respectivas etapas.

8.4.1.12 Mantenimiento del proyecto.

Una vez que el modelo es implementado, es puesto en marcha, existe una de variable importante que suele estar muchas veces implícita y que muchos equipos de BI suelen olvidarse de ella. Esta variable determina prácticamente la vida útil de las soluciones implementadas en el tiempo, por lo que es conocida como mantenimiento de los proyectos. Un mantenimiento consiste en dar el correspondiente soporte técnico de las soluciones de BI/DW, en el que básicamente se monitorea proactivamente el rendimiento y tendencias en la capacidad de la solución. De tal forma que nos permite de manera directa darnos cuenta en todo momento sobre el estado actual de la solución y en base a los hallazgos se toman decisiones de mantenimiento que garanticen que la solución es lo suficientemente potente para satisfacer las necesidades presentes y futuras.

8.4.1.13 Crecimiento del proyecto.

Dado que la mayoría de las industrias se están enfocando en metodologías agiles, el enfoque del ciclo de vida de Kimbal[9] y las metodologías agiles comparten algunas doctrinas comunes. Ambas se enfocan en aportar valor al negocio, colaboración con el negocio y el desarrollo de manera incremental.

Los nuevos requerimientos del proyecto se manejarán de la misma manera en la que se manejó el proyecto inicial, es decir que se repetirán cada uno de los pasos del ciclo de vida de Kimbal de nuevo.

9. Desarrollo.

9.1 Introducción a la lógica del negocio.

StackOverflow es un sitio de preguntas y respuestas para programadores, entusiastas del desarrollo y uso de software. Se fundó en el año de 2008, convirtiéndose en uno de los sitios web que cuenta con una gran popularidad alrededor del mundo. Transformando plenamente la manera de trabajar de las personas. Se trabaja constantemente para crear una biblioteca de respuestas detalladas para todas las preguntas sobre programación, desarrollo y uso de software como tal.

El usuario crea una cuenta para formar parte de la plataforma. A través de esa cuenta realizaran las preguntas que permitan responder a esas necesidades que surgen durante el desarrollo de sus actividades laborales. El usuario elabora la pregunta y la publica en el sitio, esa pregunta se convierte en un post. Por lo que podrá tener las correspondientes respuestas de los demás miembros de la comunidad.

Tanto preguntas como respuestas pueden tener puntos a favor o en contra y según sea el desempeño del usuario pueden ir escalando en méritos, ganándose insignias que le permitan ir aumentando sus privilegios dentro de la plataforma. Gracias a estos méritos podrán convertirse en moderadores. A cada pregunta se le relaciona con una etiqueta especializada que permite optimizar las búsquedas dentro de la comunidad, estas etiquetas están relacionadas con las tecnologías que más preguntas y respuestas generan en el sitio de StakOverflow.

El significado de pertenecer a esta plataforma es ayudar a ese conjunto de personas que se desenvuelven dentro de las profesiones relacionadas con las tecnologías que evolucionan a pasos agigantados. Evidentemente las necesidades han aumentado exponencialmente por lo que la plataforma cada vez gana mucho más valor en el tiempo.

La comunidad de StakOverflow supone un esfuerzo interdisciplinario, es decir un grupo de personas de diferentes áreas trabajando en conjunto para poder aportar valor a la comunidad ante las preguntas que el sitio recibe. Se está apoyando directamente a millones de personas en el mundo, respondiendo el conjunto de preguntas que surgen de la necesidad de solucionar problemas que se encuentran en la vida laboral.

[9] Ralph Kimball uno de los mayores influyentes en el diseño de modelos dimensionales.

Centralizar a la comunidad ante todo es uno de los principales valores que más se persigue, porque de esta manera se fomenta las comunidades saludables. Comunidades saludables que en todo momento tengan la suficiente intención de aprender, retribuir a la comunidad y adoptar una mentalidad de crecimiento ético a largo plazo.

StackOverflow es un sitio que cuenta con un enorme abanico de oportunidades para solventar nuestras necesidades, necesidades que han experimentado un crecimiento en los últimos años. Esto es debido a la exponencial curva de usuarios que interactúan con las nuevas tecnologías, originando el crecimiento desmedido de grandes volúmenes de datos. Es de vital interés poder explotar, usar o analizar estos grandes volúmenes de datos ya que nos darán un panorama del comportamiento de esta comunidad. Resultados que permitirán reorientar los esfuerzos para la consecuente mejora en las políticas operativas del sitio como tal.

Es por ello que se construirá un modelo dimensional que representara plenamente a dos procesos de negocio muy importantes, uno de ellos es preguntas realizadas y el otro es respuestas hechas. De esta manera estamos cubriendo las dos principales áreas de potencial interés analítico para la entidad. El modelo final se almacenará en una capa de presentación en donde estarán los datos transformados, estructurados para que sean consumibles por los usuarios analíticos.

A través de la construcción de un dashboards se permitirá que los datos sean consumidos plenamente y se espera que el mismo satisfaga todas las necesidades que los usuarios tienen de datos. Es de vital importancia garantizar la calidad, la efectividad y por supuesto el performance para que el modelo sea totalmente adoptado por la comunidad de StackOverflow.

Se aplicará el modelado dimensional, conformado por el proceso de negocio, determinación del nivel de granularidad, identificación de dimensiones y determinación de métricas que medirán los eventos que suceden en el proceso de negocio. En la primera etapa se cubrirá todo el proceso de profiling sobre el dataset de StackOverflow, el diccionario de datos para poder conocer tabla por tabla y su estructura. Además, se evidenciará el resultado del modelado dimensional por medio del esquema estrella y finalmente se cubrirá el proceso de mappings por cada tabla que se utilizará para construir el modelo dimensional como tal.

9.2 Descripción del dataset.

A continuación, se detalla de manera concreta como los datos que conforman el dataset de StackOverflow están organizados:

Origen de los datos:	Los datos provienen de un modelo de base de datos transaccional o relacional, almacenados en la plataforma BigQuery para el sitio de preguntas y respuestas de StackOverflow.
Formato:	Los datos están organizados en 16 tablas con formato csv, con encabezados en sus tablas, uso de separadores tales como [,]
Estructura:	Cada una de las tablas con las que cuenta el modelo relacional poseen sus correspondientes llaves primarias y foráneas por las cuales se relacionan. Teniendo consigo metada correspondiente al esquema que cada una posee, es decir sus respectivos tipos de datos para cada columna que la conforma.

Periodo de tiempo del dataset:	El dataset comprende datos desde el año 2008 hasta mayo del año 2021, actualizándose este trimestralmente.

Tabla 3. Descripción del dataset de la plataforma de StackOverflow

A continuación, presentaremos el correspondiente modelo relacional que constituye nuestra principal fuente de datos para la construcción del modelo dimensional:

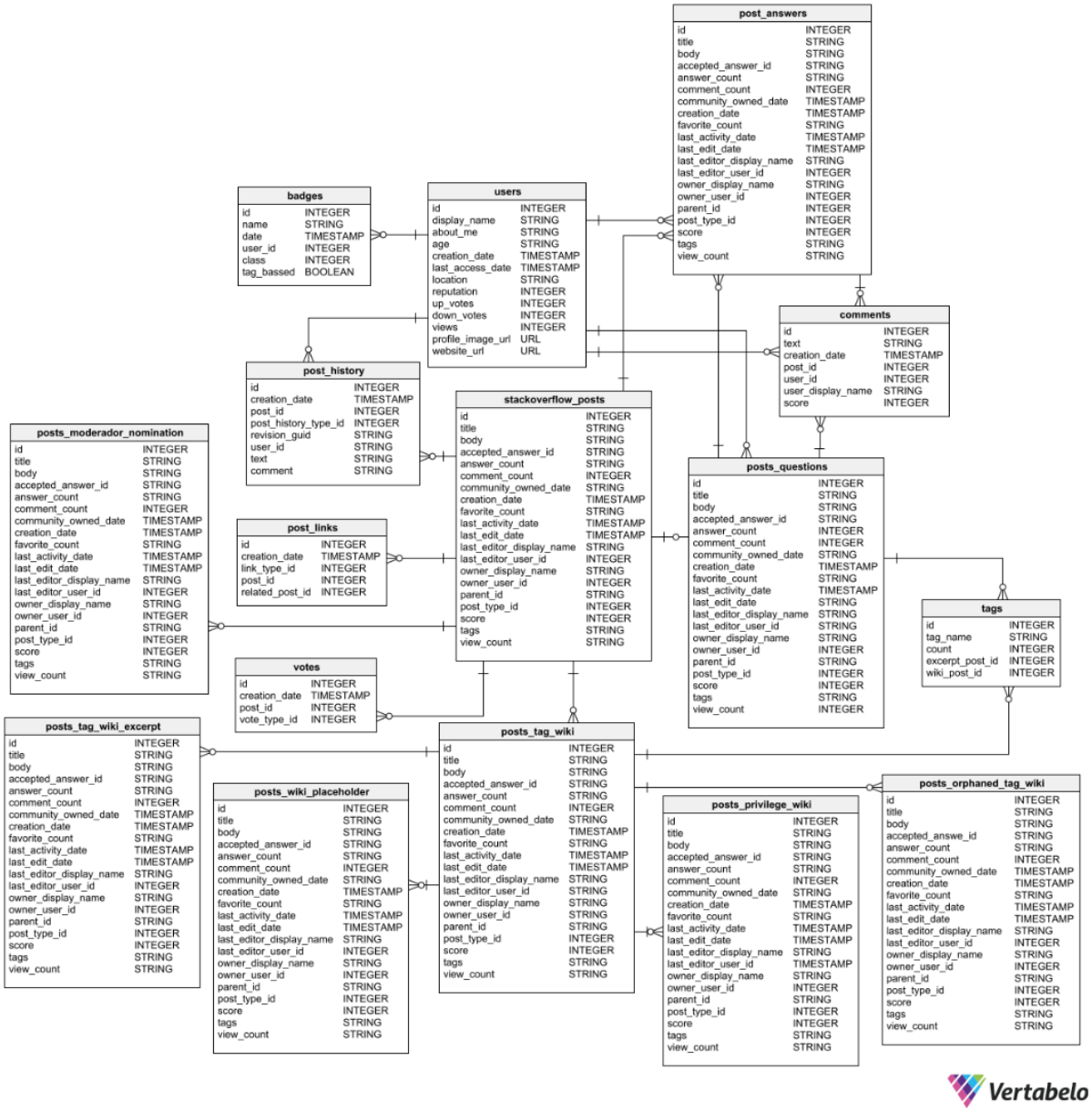

Figura 16. Modelo relacional del dataset para la plataforma StackOverflow

A continuación, describiremos con más detalles cada una de las tablas que conforman nuestro dataset de StackOverflow:

N°	Nombre de tabla	Cantidad de registros	Tamaño	Columnas	Utilizada para-MD
1	badges	39,178,976	1.78 GB	6	Yes
2	comments	79,220,809	16.64 GB	7	No
3	posts_answers	31,169,249	27.6 GB	20	Yes
4	posts_moderator_nomination	334	473.5 KB	20	No
5	posts_orphaned_tag_wiki	167	53.34 KB	19	No
6	post_history	138,808,451	109.24 GB	8	Yes
7	post_links	7,302,589	291.1 MB	5	No
8	users	14,080,580	2.4 GB	13	Yes
9	posts_privilege_wiki	2	3.49 KB	20	No
10	posts_questions	20,890,054	35.58 GB	20	Yes
11	posts_tag_wiki	53036	36.57 MB	20	No
12	posts_tag_wiki_excerpt	53036	11.02 MB	20	No
13	posts_wiki_placeholder	5	5.63 KB	20	No
14	stackoverflow_posts	31,017,889	31.52 GB	20	Yes
15	tags	60,534	2.48 MB	5	Yes
16	votes	208,577,841	6.67 GB	4	Yes

Tabla 4. Descripción de cada una de las tablas del dataset de la plataforma de StackOverflow

9.3 Diccionario de datos del dataset

9.3.1 Badges.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador llave primaria de tabla
name	Varchar	Yes	No	No	Nombre de la categoría de usuario
date	Datetime	Yes	No	No	Fecha y Hora registrada del evento
user_id	Integer	Yes	No	Yes	Identificador de usuario que inicio el evento
class	Integer	Yes	No	No	Numero de clase
tag_based	Boolean	Yes	No	No	Indicador de estado actual del usuario

Tabla 5. Descripción de la tabla badges

9.3.2 comments.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
text	Varchar	Yes	No	No	Descripción de los comentarios
creation_date	Datetime	Yes	No	No	Fecha y Hora de creación del comentario
post_id	Integer	Yes	No	Yes	Identificador del post o evento de consulta
user_id	Integer	Yes	No	Yes	Identificador de usuario
user_display_name	Varchar	Yes	No	No	Username del usuario
score	Integer	Yes	No	No	Puntaje alcanzado del post

Tabla 6. Descripción de la tabla comments

9.3.3 post_answer.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
title	Varchar	Yes	No	No	Descripción del título de las respuestas
body	Varchar	Yes	No	No	Descripción de la respuesta correspondiente
accepted_answer_id	Integer	Yes	No	Yes	Identificador de respuesta aceptada
answer_count	Integer	Yes	No	No	Conteo del número de respuestas recibidas para la pregunta
comment_count	Integer	Yes	No	No	Conteo de los comentarios
community_owned_date	Datetime	Yes	No	No	Fecha y Hora registrada por la plataforma
creation_date	Datetime	Yes	No	No	Fecha y Hora de creación del evento
favorite_count	Integer	Yes	No	No	Conteo de los posts o respuestas favoritas
last_activity_date	Datetime	Yes	No	No	Fecha y Hora de la última actividad en el post
last_edit_date	DateTime	Yes	No	No	Fecha y Hora de la edición del post
last_editor_display_name	Varchar	Yes	No	No	Contiene el nombre del editor del post incluyendo el nombre e id
last_editor_user_id	Integer	Yes	No	Yes	Identificador único del editor del post
owner_display_name	Varchar	Yes	No	No	Nombre del dueño del post o evento de consulta
owner_user_id	Integer	Yes	No	Yes	Identificador del dueño del post o evento de la consulta
parent_id	Integer	Yes	No	Yes	Identificador del evento padre
post_type_id	Integer	Yes	No	Yes	Identificador del tipo de post o evento de consulta
score	Integer	Yes	No	No	Puntaje del post

tags	Varchar	Yes	No	No	Etiqueta del post
view_count	Integer	Yes	No	No	Conteo de vistas del post

Tabla 7. Descripción de la tabla post_answer

9.3.4 post_moderator_nomination.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
title	Varchar	Yes	No	No	Descripción del titulo
body	Varchar	Yes	No	No	Descripción de las acciones del moderador del post
accepted_answer_id	Integer	Yes	No	Yes	Identificador de la respuesta aceptada
answer_count	Integer	Yes	No	No	Conteo de las respuestas del post
comment_count	Integer	Yes	No	No	Conteo de los comentarios que el post recibe
community_owned_date	Datetime	Yes	No	No	Fecha y Hora de las acciones de la comunidad
creation_date	Datetime	Yes	No	No	Fecha y Hora de creación
favorite_count	Integer	Yes	No	No	Conteo de los posts favoritos
last_activity_date	Datetime	Yes	No	No	Fecha y Hora de la última actividad del moderador
last_edit_date	Datetime	Yes	No	No	Fecha y Hora de la última edición del moderador
last_editor_display_name	Varchar	Yes	No	No	Nombre del ultimo editor del post
last_editor_user_id	Integer	Yes	No	Yes	Identificador del ultimo editor del post
owner_display_name	Varchar	Yes	No	No	Nombre del dueño del post o evento de consulta
owner_user_id	Integer	Yes	No	Yes	Identificador del usuario
parent_id	Integer	Yes	No	Yes	Identificador del propietario
post_type_id	Integer	Yes	No	Yes	Identificador del tipo de post
score	Integer	Yes	No	No	Puntaje

tags	Varchar	Yes	No	No	Etiquetas
view_count	Integer	Yes	No	No	Conteo de vistas

Tabla 8. Descripción de la tabla post_moderator_domination

9.3.5 post_orphaned_tag_wiki.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
Title	Varchar	Yes	No	No	Descripción el titulo
Body	Varchar	Yes	No	No	Descripción de las acciones
accepted_answer_id	Integer	Yes	No	Yes	Identificador de la respuesta aceptada
answer_count	Integer	Yes	No	No	Conteo de las respuestas del post
comment_count	Integer	Yes	No	No	Conteo de los comentarios recibidos por el post
community_owned_date	DateTime	Yes	No	No	Fecha y Hora de las acciones en la comunidad
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
favorite_count	Integer	Yes	No	No	Conteo de los favoritos
last_activity_date	DateTime	Yes	No	No	Fecha y Hora de la última actividad
last_edit_date	DateTime	Yes	No	No	Fecha y Hora de la última edición
last_editor_display_name	Varchar	Yes	No	No	Nombre del ultimo editor del post
last_editor_user_id	Integer	Yes	No	Yes	Identificador del ultimo usuario que llevo a cabo la edición
owner_display_name	Varchar	Yes	No	No	Nombre del dueño del post o evento de la consulta
parent_id	Integer	Yes	No	Yes	Identificador padre
post_type_id	Integer	Yes	No	Yes	Identificador del tipo de post
Score	Integer	Yes	No	No	Puntaje
Tags	Varchar	Yes	No	No	Etiquetas
view_count	Integer	Yes	No	No	Conteo de vistas del post

Tabla 9. Descripción de la tabla post_orphaned_tag_wiki

9.3.6 post_history.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
post_id	Integer	Yes	No	Yes	Identificador del post
post_history_type_id	Integer	Yes	No	Yes	Identificador del tipo de history
revision_guid	Varchar	Yes	No	No	Identificador de agrupa todos los registros generados en un historial de cambio
user_id	Varchar	No	No	Yes	Identificador del usuario
Text	Varchar	No	No	No	Texto referente al cambio
Comment	Varchar	No	No	No	Comentario opcional

Tabla 10. Descripción de la tabla post_history

9.3.7 post_links.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
link_type_id	Integer	Yes	No	Yes	Identificador del tipo de link
post_id	Integer	Yes	No	Yes	Identificador del post
related_post_id	Integer	Yes	No	Yes	Identificador del post relacionado al post

Tabla 11. Descripción de la tabla post_links

9.3.8 users.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla

display_name	Varchar	Yes	No	No	El nombre a mostrar del usuario
about_me	Varchar	No	No	No	Descripción acerca del usuario
Age	Varchar	No	No	No	Edad del usuario
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
Last_access_date	DateTime	Yes	No	No	Fecha y Hora del ultimo acceso al sitio
Location	Varchar	No	No	No	Lugar donde vive el usuario
Reputation	Integer	Yes	No	No	Reputacion que se ha ganado el usuario
up_votes	Integer	Yes	No	No	Votos a favor
down_votes	Integer	Yes	No	No	Votos en contra
Views	Integer	Yes	No	No	Visitas que ha tenido en su perfil
profile_image_url	Varchar	No	No	No	Avatar del usuario
website_url	Varchar	No	No	No	Url del sitio web del usuario

Tabla 12. Descripción de la tabla users

9.3.9 posts_privilege_wiki.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
title	Varchar	No	No	No	Título de la wiki
body	Varchar	Yes	No	No	Descripción de la wiki
accepted_answer_id	Integer	No	No	Yes	Identificador de la respuesta aceptada
answer_count	Integer	No	No	No	Conteo de las respuestas
comment_count	Integer	No	No	No	Conteo de los comentarios
community_owned_date	DateTime	No	No	No	Fecha y Hora de las acciones en la comunidad
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
favorite_count	Integer	No	No	No	Conteo de favoritos
last_activity_date	DateTime	Yes	No	No	Fecha y Hora de la última actividad

last_edit_date	DateTime	Yes	No	No	Fecha y Hora de la última edición
last_editor_display_name	Varchar	No	No	No	Nombre del ultimo editor de la wiki
last_editor_user_id	Integer	Yes	No	Yes	Identificador del ultimo usuario que llevo a cabo la edición
owner_display_name	Varchar	No	No	No	Nombre del usuario que creo la wiki
owner_user_id	Integer	No	No	Yes	Identificador del usuario que creo la wiki
parent_id	Integer	No	No	Yes	Identificador de la wiki padre
post_type_id	Integer	Yes	No	Yes	Identificador del tipo de wiki
score	Integer	Yes	No	No	Puntaje
tags	Varchar	No	No	No	Etiquetas
view_count	Integer	No	No	No	Conteo de vistas de la wiki

Tabla 13. Descripción de la tabla post_privilege_wiki

9.3.10 posts_questions.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
title	Varchar	Yes	No	No	Título del Post
body	Varchar	Yes	No	No	Descripción del post
accepted_answer_id	Integer	No	No	Yes	Identificador de la respuesta aceptada
answer_count	Integer	Yes	No	No	Conteo de las respuestas
comment_count	Integer	Yes	No	No	Conteo de los comentarios
community_owned_date	DateTime	No	No	No	Fecha y Hora de las acciones en la comunidad
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
favorite_count	Integer	No	No	No	Conteo de favoritos
last_activity_date	DateTime	Yes	No	No	Fecha y Hora de la última actividad
last_edit_date	DateTime	No	No	No	Fecha y Hora de la última edición

last_editor_display_name	Varchar	No	No	No	Nombre del ultimo editor del post
last_editor_user_id	Integer	No	No	Yes	Identificador del ultimo usuario que llevo a cabo la edición
owner_display_name	Varchar	No	No	No	Nombre del usuario que creo el post
owner_user_id	Integer	No	No	Yes	Identificador del usuario que creo el post
parent_id	Integer	No	No	Yes	Identificador del post padre
post_type_id	Integer	Yes	No	Yes	Identificador del tipo de post
score	Integer	Yes	No	No	Puntaje
tags	Varchar	Yes	No	No	Etiquetas
view_count	Integer	Yes	No	No	Conteo de vistas del post

Tabla 14. Descripción de la tabla post_question

9.3.11 posts_tag_wiki.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
title	Varchar	No	No	No	Título del tag de wiki
body	Varchar	No	No	No	Descripción del tag de wiki
accepted_answer_id	Integer	No	No	Yes	Identificador de la respuesta aceptada
answer_count	Integer	No	No	No	Conteo de las respuestas
comment_count	Integer	Yes	No	No	Conteo de los comentarios
community_owned_date	DateTime	No	No	No	Fecha y Hora de las acciones en la comunidad
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
favorite_count	Integer	No	No	No	Conteo de favoritos
last_activity_date	DateTime	Yes	No	No	Fecha y Hora de la última actividad
last_edit_date	DateTime	Yes	No	No	Fecha y Hora de la última edición

last_editor_display_name	Varchar	No	No	No	Nombre del ultimo editor del tag wiki
last_editor_user_id	Integer	No	No	Yes	Identificador del ultimo usuario que llevo a cabo la edición
owner_display_name	Varchar	No	No	No	Nombre del usuario que creo el tag wiki
owner_user_id	Integer	No	No	Yes	Identificador del usuario que creo el tag wiki
parent_id	Integer	No	No	Yes	Identificador del post padre
post_type_id	Integer	Yes	No	Yes	Identificador del tipo de wiki
score	Integer	Yes	No	No	Puntaje
tags	Varchar	No	No	No	Etiquetas
view_count	Integer	No	No	No	Conteo de vistas de la wiki

Tabla 15. Descripción de la tabla post_tag_wiki

9.3.12 posts_tag_wiki_excerpt.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
Title	Varchar	Yes	No	No	Descripción el titulo
Body	Varchar	Yes	No	No	Cuerpo del extracto de la wiki
accepted_answer_id	Integer	Yes	No	Yes	Identificador de la respuesta aceptada
answer_count	Integer	Yes	No	No	Conteo de las respuestas del post
comment_count	Integer	Yes	No	No	Conteo de los comentarios recibidos por el post
community_owned_date	DateTime	Yes	No	No	Fecha y Hora de las acciones en la comunidad
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
favorite_count	Integer	Yes	No	No	Conteo de favoritos
last_activity_date	DateTime	Yes	No	No	Fecha y Hora de la última actividad
last_edit_date	DateTime	Yes	No	No	Fecha y Hora de la última edición

last_editor_display_name	Varchar	Yes	No	No	Nombre del ultimo editor del post
last_editor_user_id	Integer	Yes	No	Yes	Identificador del ultimo usuario que llevo a cabo la edición
owner_display_name	Varchar	Yes	No	No	Nombre del dueño del post o evento de la consulta
parent_id	Integer	Yes	No	Yes	Identificador padre
post_type_id	Integer	Yes	No	Yes	Identificador del tipo de post
Score	Integer	Yes	No	No	Puntaje
Tags	Varchar	Yes	No	No	Etiquetas
View_count	Integer	Yes	No	No	Conteo de vistas del post

Tabla 16. Descripción de la tabla post_tag_wiki_excerpt

9.3.13 posts_wiki_placeholder

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
Title	Varchar	Yes	No	No	Descripción el titulo
Body	Varchar	Yes	No	No	Cuerpo del placeholder con la wiki
accepted_answer_id	Integer	Yes	No	Yes	Identificador de la respuesta aceptada
answer_count	Integer	Yes	No	No	Conteo de las respuestas del post
comment_count	Integer	Yes	No	No	Conteo de los comentarios recibidos por el post
community_owned_date	DateTime	Yes	No	No	Fecha y Hora de las acciones en la comunidad
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
favorite_count	Integer	Yes	No	No	Conteo de favoritos
last_activity_date	DateTime	Yes	No	No	Fecha y Hora de la última actividad
last_edit_date	DateTime	Yes	No	No	Fecha y Hora de la última edición
last_editor_display_name	Varchar	Yes	No	No	Nombre del ultimo editor del post

last_editor_user_id	Integer	Yes	No	Yes	Identificador del ultimo usuario que llevo a cabo la edición
owner_display_name	Varchar	Yes	No	No	Nombre del dueño del post o evento de la consulta
parent_id	Integer	Yes	No	Yes	Identificador padre
post_type_id	Integer	Yes	No	Yes	Identificador del tipo de post
Score	Integer	Yes	No	No	Puntaje
Tags	Varchar	Yes	No	No	Etiquetas
View_count	Integer	Yes	No	No	Conteo de vistas del post

Tabla 17. Descripción de la tabla post_wiki_placeholder

9.3.14 stackoverflow_posts.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
Title	Varchar	Yes	No	No	Descripción el titulo
Body	Varchar	Yes	No	No	Cuerpo de los posts de stackoverflow
accepted_answer_id	Integer	Yes	No	Yes	Identificador de la respuesta aceptada
answer_count	Integer	Yes	No	No	Conteo de las respuestas del post
comment_count	Integer	Yes	No	No	Conteo de los comentarios recibidos por el post
community_owned_date	DateTime	Yes	No	No	Fecha y Hora de las acciones en la comunidad
creation_date	DateTime	Yes	No	No	Fecha y Hora de creación
favorite_count	Integer	Yes	No	No	Conteo de favoritos
last_activity_date	DateTime	Yes	No	No	Fecha y Hora de la última actividad
last_edit_date	DateTime	Yes	No	No	Fecha y Hora de la última edición
last_editor_display_name	Varchar	Yes	No	No	Nombre del ultimo editor del post
last_editor_user_id	Integer	Yes	No	Yes	Identificador del ultimo usuario que llevo a cabo la edición

owner_display_name	Varchar	Yes	No	No	Nombre del dueño del post o evento de la consulta
parent_id	Integer	Yes	No	Yes	Identificador padre
post_type_id	Integer	Yes	No	Yes	Identificador del tipo de post
Score	Integer	Yes	No	No	Puntaje
Tags	Varchar	Yes	No	No	Etiquetas
View_count	Integer	Yes	No	No	Conteo de vistas del post

Tabla 18. Descripción de la tabla stackoverflow_post

9.3.15 tags.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
Tag_name	Varchar	Yes	No	No	Nombre de la etiqueta
Count	Integer	Yes	No	No	Conteo de la etiqueta
Excerpt_post_id	Integer	No	No	Yes	Identificador del extracto del post
Wiki_post_id	Integer	No	No	Yes	Identificador de la wiki del post

Tabla 19. Descripción de la tabla tags

9.3.16 votes.

Attribute name	Type	Is mandatory	Is Primary	Is Foreing	Description
Id	Integer	Yes	Yes	No	Identificador de llave primaria de tabla
Creation_date	DateTime	Yes	No	No	Fecha de creación del voto
Post_id	Integer	Yes	No	Yes	Identificador del post
Vote_type_id	Integer	Yes	No	Yes	Indentificador del tipo de voto

Tabla 20. Descripción de la tabla votes

9.4 Resultados del data profiling del dataset.

A continuación, daremos una breve descripción de las principales herramientas que fueron utilizadas para realizar el proceso de data profiling:

Plataforma	Descripción
	Es una herramienta que brinda un servicio inteligente de datos en la nube que te permite examinar, limpiar y preparar datos de forma visual para analizarlos y crear modelos de aprendizaje automático, cabe destacar que será utilizada directamente para perfilar los datos del dataset.

Tabla 21. Descripción de la herramienta Cloud Dataprep para el profiling del dataset

Para realizar el data profiling sobre el dataset de StackOverflow que se encuentra en los recursos públicos de ByQuery se utilizó Cloud Dataprep.

A continuación, se mostrarán los correspondientes resultados del perfilado de datos, describiéndolo por cada una de las tablas como tal:

9.4.1 Badges

Nombre del Campo	Tipo	Proporción de validos [=validos/total]	Resultado	Conclusión
Id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
name	Varchar	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
date	Datetime	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio.	Requiere transformación
user_id	Integer		El campo experimenta una proporción de 100% de registros válidos y 0% no válidos,	Apto para MD

		100%	por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	
class	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD, ya que es un campo completamente limpio.	Requiere limpieza
Tag_based	Boolean	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD, pero requerirá de transformación para buscar el valor más textual.	Requiere transformación

Tabla 22. Descripción del resultado del data profiling para la tabla badges

9.4.2 Comments.

Nombre del Campo	**Tipo**	**Proporción de validos [=validos/total]**	**Resultado**	**Conclusión**
Id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
text	Varchar	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
creation_date	DateTime	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio.	Requiere transformación
post_id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse	Apto para MD

			en la construcción del MD ya que es un campo completamente limpio.	
user_id	Integer	98%	De las 79.2 M de filas, solo 78.1 M son válidas que representan el 98%, 1.14 M contienen el valor de NULL que representan el 2%.	Apto para MD
user_display_name	Varchar	1.45%	De las 79.2 M de filas, solo 1.15 M son válidas que representan el 0.014%, 78.1M contienen el valor de NULL que representan el 98.5%, existen valores atípicos también.	No apto para MD
score	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD

Tabla 23. Descripción del resultado del data profiling para la tabla comments

9.4.3 post_answer

Nombre del Campo	Tipo	Proporción de validos [=validos/total]	Resultado	Conclusión
id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
title	Varchar	0%	El campo experimenta una proporción de 0% de registros no válidos y 0% válidos, por lo que es catalogada para no poder usarse en la construcción del MD.	No se puede usar en MD
body	Varchar	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válido, todas las filas se encuentran en un formato específico que no denota explícitamente el contexto que da a conocer por lo que requerirá de transformación.	Requiere transformación
accepted_answer_id	integer	0%	El campo experimenta una proporción de 0% de registros válidos y 100% de no válidos, todas las filas para esta columna poseen el valor de null, por lo que se	No se puede usar en MD

			imposibilita obtener los identificadores de las respuestas aceptadas.	
answer_count	Integer	0%	El campo experimenta una proporción de 0% de registros válidos y 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se imposibilita obtener el conteo del número de respuestas recibidas.	No se puede usar en MD
comment_count	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% de registros no válidos, se presentaron algunos valores atípicos por lo que se requerirá aplicar una solución de limpieza de los mismos.	Requiere limpieza
xommunity_owned_date	Datetime	36%	Se presentaron solamente 113,484 valores correctos que corresponden al 36% de los válidos y el 64% de no validos por lo que se desconoce el paradero de los datos faltantes.	No se puede usar en MD
creation_date	DateTime	100%	El campo experimenta una proporción de 100% de registros válidos y 0% de no válidos, la columna posee una mezcla de formatos tanto para fecha y hora, por lo que se requerirá de transformación.	Requiere transformación
favorite_count	Integer	0%	El campo experimenta una proporción de 100% de registros no válidos y 0% válido, por lo que es catalogada para no poder usarse en la construcción del MD.	No se puede usar en MD
last_activity_date	Datetime	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio	Requiere transformación
last_edit_date	Datetime	30%	De las 31.2M filas solo 9.59 M presentaron valores correctos que representan el 30% de válidos, 21.6 M que representan el 60% de no válidos, estos contienen valores nulos	Requiere de limpieza
last_editor_display_name	Varchar		De las 31.2 M filas solo 149, 159 presentaron valores correctos, 31.0 M contiene valores nulos.	No se puede usar en MD

last_editor_user_id	Integer	30%	De las 31.2M filas solo 9.51 M presentaron valores correctos que representan el 30% de válidos, 21.7 M contiene valores nulos que representan el 60% de no válidos.	No se puede usar en MD
owner_display_name	Varchar	2%	De las 31.2 M filas solo 666,892 presentaron valores validos o correctos que representan el 2% de válidos, 30.5 M contiene valores nulos que representan el 98% de no validos	No se puede usar en MD
owner_user_id	integer	98%	De las 31.2 M filas solo 30.9 M presentaron valores validos o correctos que representan el 98% de válidos, 299,663 contiene valores nulos que representan el 2% de no válidos.	Requiere limpieza
parent_id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
post_type_id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
Score	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
Tags	Varchar	1%	De las 31.2M filas solo 1 fila presento valores correctos que representan el 1% de válidos, 31.1M contiene valores nulos que representan el 99% de no válidos.	No se pueden usar en MD
view_count	integer	0%	De las 31.2M filas ninguna presento valores correctos que representan el 0% de valores válidos, 31.1M contiene valores nulos que representan el 100% de no válidos.	No se puede usar en MD

Tabla 24. Descripción del resultado del data profiling para la tabla post_answer

9.4.4 post_moderator_nomination.

Nombre del Campo	Tipo	Proporción de validos [=validos/total]	Resultado	Conclusión
Id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
title	Varchar	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
body	Varchar	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio	Apto para MD
accepted_answer_id	Integer	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
answer_count	Integer	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
comment_count	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio	Apto para MD
community_owned_date	Datetime	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio.	Requiere transformacion

creation_date	Datetime	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio.	Requiere transformacion
favorite_count	Integer	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
last_activity_date	Datetime	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio.	Requiere transformacion
last_edit_date	Datetime	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio.	Requiere transformacion
last_editor_display_name	Varchar	1.4%	De las 334 filas solo 5 presentaron valores validos o correctos que representan el 1.4% de válidos, el 98.6% representa los no validos contiene valores nulos.	No se puede usar en MD
last_editor_user_id	Integer	98%	De las 334 filas solo 329 presentaron valores validos o correctos que representan el 98% de válidos, el 2% representan valores no válidos.	Requiere de limpieza
owner_display_name	Varchar	1.8%	De las 334 filas solo 6 presentaron valores validos o correctos que representan el 1.8% de válidos, 98.2% representa los no validos conteniendo valores nulos.	No se puede usar en MD
owner_user_id	Integer	98%	Del 334 flas solo 328 presentaron valores validos o correctos que representan el 98% de válidos, 2% representa a los valores no válidos, por lo que requiere de limpieza.	Requiere limpieza

parent_id	Integer	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
post_type_id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
score	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
tags	Varchar	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
view_count	integer	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD

Tabla 25. Descripción del resultado del data profiling para la tabla post_moderator_nomination

9.4.5 post_orphaned_tag_wiki.

Nombre del Campo	**Tipo**	**Proporción de validos** **=validos/total**	**Resultado**	**Conclusión**
Id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
title	Varchar	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo	No se puede usar en MD

			que se dificulta obtener el contexto que representa la columna.	
body	Varchar	33%	De las 167 filas solo 111 presentaron valores correctos que representan el 33% de válidos, 67% que representan no validos contiendo estos valores nulos, por lo que requerirá un proceso de limpieza previamente.	Requiere limpieza
accepted_answer_id	Integer	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
answer_count	Integer	0%	El campo experimento 0% de registros válidos y el 100% de registros no válidos, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
comment_count	Integer	100%	El campo experimenta una proporción de 100% de registros válidos siendo estos 167 y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
community_owned_date	Datetime	1.8%	El campo experimenta una proporción de 1.8% de registros validos siendo estos solamente 3, 98.2% que representan los no validos siendo estos 164 registros.	No se puede usar en MD
creation_date	Datetime	100%	El campo experimenta una proporción de 100% de registros válidos y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio.	Requiere transformacion
favorite_count	Integer	0%	El campo experimenta una proporción de 0% de registros válidos y el 100% de registros no válidos siendo estos 167, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD

last_activity_date	Datetime	100%	El campo experimenta una proporción de 100% de registros válidos siendo estos 167 y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio.	Requiere transformacion
last_edit_date	Datetime	100%	El campo experimenta una proporción de 100% de registros válidos siendo estos 167 y 0% no válidos, pero Las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD ya que es un campo completamente limpio.	Requiere transformacion
last_editor_display_name	Varchar	1.2%	De las 167 filas solo 2 presentaron valores correctos que representan el 1.2%, el resto 98.8% contiene valores nulos	No se puede usar en MD
last_editor_user_id	Integer	98.8%	De las 167 filas solo 165 presentaron valores correctos que representan el 98.8% de válidos, el resto 1.2% contiene valores nulos, por lo que requerirá de un proceso de limpieza.	Requiere limpieza
owner_display_name	Varchar	0%	El campo experimenta una proporción de 0% de registros válidos y el 100% de registros no válidos siendo estos 167, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
parent_id	Integer	0%	El campo experimenta una proporción de 0% de registros válidos y el 100% de registros no válidos siendo estos 167, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
post_type_id	Integer	100%	El campo experimenta una proporción de 100% de registros válidos siendo estos 167 y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
Score	Integer	100%	El campo experimenta una proporción de 100% de registros válidos siendo estos 167 y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD

Tags	Varchar	0%	El campo experimenta una proporción de 0% de registros válidos y el 100% de registros no válidos siendo estos 167, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD
view_count	Integer	0%	El campo experimenta una proporción de 0% de registros válidos y el 100% de registros no válidos siendo estos 167, todas las filas para esta columna poseen el valor de null, por lo que se dificulta obtener el contexto que representa la columna.	No se puede usar en MD

Tabla 26. Descripción del resultado del data profiling para la tabla post_orphaned_tag_wiki

9.4.6 post_history.

Nombre del Campo	Tipo	Proporción de validos =validos/138M	Resultado	Conclusión
id	Integer	100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
creation_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD.	Requiere transformación
post_id	Integer	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
post_history_type_id	Integer	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
revision_guid	Varchar	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Requiere transformación
user_id	Varchar	95%	Proporción del 95% de registros válidos y 5% no válidos, sin embargo, este 5% equivale a 6	Requiere limpieza

			millones de registros por lo que requerirá de limpieza.	
text	Varchar	96%	Proporción del 95% de registros válidos y 4% no válidos, sin embargo, este 4% equivale a 4.2 millones de registros por lo que requerirá de limpieza.	Requiere limpieza
comment	Varchar	31%	Proporción del 31% de registros válidos y 69% no válidos, además de que el campo no tiene es un campo explotable para análisis, por lo que no se usara.	No se puede usar en MD

Tabla 27. Descripción del resultado del data profiling para la tabla post_history

9.4.7 post_links.

Nombre del Campo	**Tipo**	**Proporción de validos =validos/7M**	**Resultado**	**Conclusión**
Id	Integer	100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
creation_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD	Requiere transformación
link_type_id	Integer	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
post_id	Integer	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
related_post_id	Integer	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD

Tabla 28. Descripción del resultado del data profiling para la tabla post_links

9.4.8 users

Nombre del Campo	Tipo	Proporción de validos =validos/14M	Resultado	Conclusión
Id	Integer	100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
display_name	Varchar	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
about_me	Varchar	9%	Proporción del 9% de registros válidos y 0% no válidos, además el campo no aporta ningún valor analítico por lo que no será usada en el MD.	No se puede usar en MD
Age	Varchar	0%	Proporción del 0% de registros válidos y 100% no válidos, por lo que no es posible usarse en el MD.	No se puede usar en MD
creation_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD.	Requiere transformación
Last_access_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD.	Requiere transformación
Location	Varchar	25%	Proporción del 25% de registros válidos y 75% no válidos, además muchos de los valores no siguen un estándar ya que son direcciones de distintos países y con texto libre, por lo que no se usara en el MD.	No se puede usar en MD
Reputation	Integer	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
up_votes	Integer	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD

down_votes	Integer	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
Views	Integer	100%	Proporción del 100% de registros válidos y 0% no válidos, por lo que es catalogada para poder usarse en la construcción del MD ya que es un campo completamente limpio.	Apto para MD
profile_image_url	Varchar	84%	Proporción del 84% de registros válidos y 16% no válidos, el campo requerirá limpieza, pero por el valor que aportará a la hora de generar el dashboard se tomará en cuenta.	Requiere limpieza
website_url	Varchar	6%	Proporción del 6% de registros válidos y 94% no válidos, por lo que no es posible usarlo en el MD.	No se puede usar en MD

Tabla 29. Descripción del resultado del data profiling para la tabla users

9.4.9 posts_privilege_wiki

Nombre del Campo	**Tipo**	**Proporción de validos =validos/2**	**Resultado**	**Conclusión**
Id	Integer	100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
title	Varchar	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD.
body	Varchar	100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD.
accepted_answer_id	Integer	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD
answer_count	Integer	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD
comment_count	Integer	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD
community_owned_date	DateTime	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD

creation_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD.	Requiere transformación
favorite_count	Integer	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD
last_activity_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD.	Requiere transformación
last_edit_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD.	Requiere transformación
last_editor_display_name	Varchar	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD
last_editor_user_id	Integer	100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
owner_display_name	Varchar	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD
owner_user_id		100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
parent_id	Integer	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD
post_type_id	Integer	100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
score	Integer	100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
tags	Varchar	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD
view_count	Integer	0%	Proporción del 0% de registros válidos por lo que es descartada a usar en el MD.	No apto para MD

Tabla 30. Descripción del resultado del data profiling para la tabla post_privilege_wiki

9.4.10 posts_questions

Nombre del Campo	Tipo	Proporción de validos =validos/20 M	Resultado	Conclusión
id	Integer	100%	Proporción del 100% de registros válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
title	Varchar	100%	Proporción del 100% de registros válidos, únicamente requiere limpieza por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
body	Varchar	100%	Proporción del 100% de registros válidos, únicamente requiere limpieza por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
accepted_answer_id	Integer	51%	Proporción del 51% de registros válidos, ya que el campo es bueno para análisis se realizará limpieza y se buscará la forma de manejar los valores nulos.	Requiere limpieza
answer_count	Integer	100%	Proporción del 100% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
comment_count	Integer	100%	Proporción del 100% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
community_owned_date	DateTime	0%	Proporción del 100% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apta para MD
creation_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD.	Requiere transformación
favorite_count	Integer	22%	Proporción del 22% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
last_activity_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se	Requiere transformación

			busque estandarizar el mismo para la construcción del MD.	
last_edit_date	DateTime	54%	Proporción de 54% de registros válidos, se buscará la forma de manejar los valores inválidos ya que el campo aporta mucho análisis.	Requiere limpieza
last_editor_display_name	Varchar	1%	Proporción del 1% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
last_editor_user_id	Integer	53%	Proporción del 53% de registros válidos, el campo no genera mucho valor por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
owner_display_name	Varchar	2%	Proporción del 2% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
owner_user_id	Integer	98%	Proporción del 98% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
parent_id	Integer	0%	Proporción del 0% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
post_type_id	Integer	100%	Proporción del 100% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
score	Integer	100%	Proporción del 100% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
tags	Varchar	100%	Proporción del 100% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
view_count	Integer	100%	Proporción del 100% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD

Tabla 31. Descripción del resultado del data profiling para la tabla post_questions

9.4.11 posts_tag_wiki

Nombre del Campo	Tipo	Proporción de validos =validos/total	Resultado	Conclusión
Id	Integer	100%	Registros 100% válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
Title	Varchar	0%	Proporción del 0% de registros válidos, no es catalogada para poder usarse en la construcción del MD.	No apto para MD
body	Varchar	72%	Proporción del 72% de registros válidos, no genera valor analítico por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
accepted_answer_id	Integer	0%	Proporción del 0% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
answer_count	Integer	0%	Proporción del 0% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
comment_count	Integer	100%	Proporción del 100% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
community_owned_date	DateTime	0.6%	Proporción del 0.6% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
creation_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD.	Requiere transformación
favorite_count	Integer	0%	Proporción del 0% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
last_activity_date	DateTime	100%	Proporción de 100% de registros válidos, las filas contienen un formato de fecha y hora que se tendrá que transformar de tal manera que se busque estandarizar el mismo para la construcción del MD.	Requiere transformación

last_edit_date	DateTime	100%	Registros 100% válidos por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
last_editor_display_name	Varchar	1.1%	Proporción del 1.1% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
last_editor_user_id	Integer	92.2%	Registros 100% válidos por lo que es catalogada para poder usarse en la construcción del MD.	Requiere limpieza
owner_display_name	Varchar	0.6%	Proporción del 0.6% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
owner_user_id	Integer	99.4%	Proporción del 98% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
parent_id	Integer	0%	Proporción del 0% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
post_type_id	Integer	100%	Proporción del 100% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
score	Integer	100%	Proporción del 100% de registros válidos, por lo que es catalogada para poder usarse en la construcción del MD.	Apto para MD
tags	Varchar	0%	Proporción del 0% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD
view_count	Integer	0%	Proporción del 0% de registros válidos, por lo que no es catalogada para poder usarse en la construcción del MD.	No apto para MD

Tabla 32. Descripción del resultado del data profiling para la tabla posts_tag_wiki

9.4.12 posts_tag_wiki_excerpt

Nombre del Campo	Tipo	Proporción de validos =validos/5	Resultado	Conclusión
Id	Integer		Todos válidos, pero hay pocos registros.	No se puede usar en MD

		100%		
Title	Varchar	0%	Todos son nulos	No se puede usar en MD
Body	Varchar	100%	Todos válidos, pero hay pocos registros	No se puede usar en MD
accepted_answer_id	Integer	0%	Todos son nulos	No se puede usar en MD
answer_count	Integer	0%	Todos son nulos	No se puede usar en MD
comment_count	Integer	100%	Todos válidos, pero hay pocos registros	No se puede usar en MD
community_owned_date	DateTime	0%	Todos son nulos	No se puede usar en MD
creation_date	DateTime	100%	Todos válidos, pero hay pocos registros	No se puede usar en MD
favorite_count	Integer	0%	Todos son nulos	No se puede usar en MD
last_activity_date	DateTime	100%	Todos válidos, pero hay pocos registros	No se puede usar en MD
last_edit_date	DateTime	100%	Todos válidos, pero hay pocos registros	No se puede usar en MD
last_editor_display_name	Varchar	0%	Todos son nulos	No se puede usar en MD
last_editor_user_id	Integer	100%	Todos válidos, pero hay pocos registros	No se puede usar en MD
owner_display_name	Varchar	0%	Todos son nulos	No se puede usar en MD
Owner_user_id		100%	Todos válidos, pero hay pocos registros	No se puede usar en MD
parent_id	Integer	0%	Todos son nulos	No se puede usar en MD
post_type_id	Integer	100%	Todos válidos, pero hay pocos registros	No se puede usar en MD

Score	Integer	100%	Todos válidos, pero hay pocos registros	No se puede usar en MD
Tags	Varchar	0%	Todos son nulos	No se puede usar en MD
view_count	Integer	0%	Todos son nulos	No se puede usar en MD

Tabla 33. Descripción del resultado del data profiling para la tabla posts_tag_wiki_excerpt

9.4.13 posts_wiki_placeholder

Nombre del Campo	Tipo	Proporción de validos =validos/53036	Resultado	Conclusión
Id	Integer	100%	Todos válidos	Apto para MD
Title	Varchar	1%	Mayoría nulos	No se puede usar en MD
Body	Varchar	92%	La mayoría de registros tienen datos, pero requerirá limpieza para los que están nulos.	Requiere limpieza
accepted_answer_id	Integer	0%	Todos son nulos	No se puede usar en MD
answer_count	Integer	0%	Todos son nulos	No se puede usar en MD
comment_count	Integer	100%	Todos válidos	Apto para MD
community_owned_date	DateTime	0%	Todos son nulos	No se puede usar en MD
creation_date	DateTime	100%	Todos válidos	Apto para MD
favorite_count	Integer	0%	Todos son nulos	No se puede usar en MD
last_activity_date	DateTime	100%	Todos válidos	Apto para MD

last_edit_date	DateTime	100%	Todos válidos	Apto para MD
last_editor_display_name	Varchar	1%	Mayoría nulos	No se puede usar en MD
last_editor_user_id	Integer	99%	Todos válidos, pero hay pocos registros	Requiere limpieza
owner_display_name	Varchar	1%	Mayoría nulos	No se puede usar en MD
Owner_user_id		99%	La mayoría de registros tienen datos, pero requerirá limpieza para los que están nulos.	Requiere limpieza
parent_id	Integer	0%	Todos son nulos	No se puede usar en MD
post_type_id	Integer	100%	Todos válidos	Apto para MD
Score	Integer	100%	Todos válidos	Apto para MD
Tags	Varchar	0%	Todos son nulos	No se puede usar en MD
view_count	Integer	0%	Todos son nulos	No se puede usar en MD

Tabla 34. Descripción del resultado del data profiling para la tabla posts_wiki_placeholder

9.4.14 stackoverflow_posts

Nombre del Campo	**Tipo**	**Proporción de validos =validos/31M**	**Resultado**	**Conclusión**
Id	Integer	100%	Todos válidos	Apto para MD
Title	Varchar	38%	Muchos nulos, pero se puede rescatar	Requiere limpieza
Body	Varchar	100%	La mayoría de registros tienen datos, pero requerirá limpieza para los que están nulos.	Apto para MD
accepted_answer_id	Integer	21%	Mayoría nulos	No se puede usar en MD

answer_count	Integer	38%	Muchos nulos, pero se puede rescatar	Requiere limpieza
comment_count	Integer	100%	Todos válidos	Apto para MD
community_owned_date	DateTime	0%	Todos son nulos	No se puede usar en MD
creation_date	DateTime	100%	Todos válidos	Apto para MD
favorite_count	Integer	9%	Mayoría nulos	No se puede usar en MD
last_activity_date	DateTime	100%	Todos válidos	Apto para MD
last_edit_date	DateTime	36%	Muchos nulos, pero se puede rescatar	Requiere limpieza
last_editor_display_name	Varchar	1%	Mayoría nulos	No se puede usar en MD
last_editor_user_id	Integer	36%	Muchos nulos, pero se puede rescatar	Requiere limpieza
owner_display_name	Varchar	2%	Mayoría nulos	No se puede usar en MD
Owner_user_id		99%	La mayoría de registros tienen datos, pero requerirá limpieza para los que están nulos.	Requiere limpieza
parent_id	Integer	62%	Mayoría validos	Requiere limpieza
post_type_id	Integer	100%	Todos válidos	Apto para MD
Score	Integer	100%	Todos válidos	Apto para MD
Tags	Varchar	38%	Muchos nulos, pero se puede rescatar	Requiere limpieza
view_count	Integer	38%	Muchos nulos, pero se puede rescatar	Requiere limpieza

Tabla 35. Descripción del resultado del data profiling para la tabla stackoverflow_posts

9.4.15 tags.

	Tipo	Proporción de validos =validos/60534	Resultado	Conclusión
Id	Integer	100%	Todos válidos	Apto para MD
Tag_name	Varchar	100%	Todos válidos	Apto para MD
Count	Integer	100%	Todos válidos	Apto para MD
Excerpt_post_id	Integer	72%	Mayoría validos	Requiere limpieza
Wiki_post_id	Integer	72%	Muchos validos	Requiere limpieza

Tabla 36. Descripción del resultado del data profiling para la tabla tags

9.4.16 votes.

Nombre del Campo	Tipo	Proporción de validos =validos/209M	Resultado	Conclusión
Id	Integer	100%	Todos válidos	Apto para MD
Creation_date	Varchar	100%	Todos válidos	Apto para MD
Post_id	Integer	100%	Todos válidos	Apto para MD
Vote_type_id	Integer	100%	Todos válidos	Apto para MD

Tabla 37. Descripción del resultado del data profiling para la tabla votes

9.5 Estándares de diseño para base de datos y programación

9.5.1 Estándares de diseño para el modelado dimensional

Estándar	Descripción	Ejemplo
Nombramiento de atributos	Todos los atributos tienen que estar en minúscula y separados por guion bajo.	view_count
Nombramiento de llave natural	Su nombre comenzara con id, luego nombre de la tabla y finalizara con nk, separados por guion bajo.	Id_user_nk
Nombramiento de llaves primarias de dimensiones y Fact tables	Su nombre comenzara con el nombre de la dimensión o Fact table finalizando con la palabra key y separado por guion bajo.	question_key
Nombramiento de llaves degeneradas	Su nombre comenzara con dd seguido del nombre de su dimensión degenerada y finalizando con key, separado por guion bajo.	dd_answer_key
Nombramiento de llave primaria de una tabla bridge	Su nombre comenzara con el nombre de la dimensión a ala que hace referencia o puente luego seguido de la palabra group y finalizando con la palabra key, separado por guion bajo.	tag_group_key
Nombramiento de dimensiones	Toda dimensión deberá de comenzar con la palabra dim seguido de su nombre, separado por guion bajo.	dim_date
Nombramiento de Fact tables	Toda Fact table deberá de comenzar con la palabra Fact seguido del nombre del proceso que representa, separado por guion bajo.	fact_done_answer

Tabla 38. Estándares de diseño para el modelado dimensional

9.5.2 Estándares de diseño para programación y documentación

Estándar	Tipo de estándar	Descripción	Ejemplo
Nombramiento de variables	Programacion	Todas las variables deben tener el estilo lowerCamelCase.	lowerCamelCase = true
Nombramiento de funciones, métodos	Programacion	Todas las variables deben tener el estilo lowerCamelCase.	def myFirstNotebook (): Int= {}

Nombramiento de clases	Programacion	Todas las clases deberán de tener el estilo UpperCamelCase.	class ScalaNotebook () { }
Idioma de documentación	Documentación	Se recomienda usar el idioma ingles para comentarios y documentación en general.	//comments in english
Comentarios	Programacion - Documentación	Se recomienda usar comentarios a nivel de línea para hacer lo más conciso posible.	//creating Fact tables question
Capa de presentación	Documentación	Todos los ETL construidos en la capa de presentación deberán de mostrar al inicio su mapeo de campos de la dimensión o fact table a construir con sus respectivas políticas de diseño.	Ver figura 17

Tabla 39. Estándares de diseño para programación y documentación

1. Description: Saves the context of the answer made.
2. Granularity: a record represents a reply post.
3. Uniqueness policy: the etl will search for responses and assign a surrogate key when this response is not stored in the dimension.
4. Invalidity policy: All fields are required.
5. SCD Policy: All fields will be Slowly Changing Dimension type one

Column name	Display name	Type	Source	Comment	Sample
answer_key	Answer Key	String	-	Surragate key generated	68d2e3f
id_answer_nk	Id natural key	Integer	stakoverflow =>post_answer=>id	Natural Key	4
last_activity_date	Last activity date	Timestamp	stakoverflow =>post_answer=>last_activity_date	-	27/06/2021
last_edit_date	Last edit date	Timestamp	stakoverflow =>post_answer=>last_edit_date	-	27/06/2021
creation_date	Creation date	Timestamp	stakoverflow =>post_answer=>creation_date	-	27/06/2021

Figura 17 Ejemplo de documentacion de capa de presentacion

9.6 Especificación de necesidades analíticas.

Nuestro modelo dimensional está basado en el dataset de StackOverflow, las necesidades que éste solventara son las siguientes:

1. ¿Cuál es el total de preguntas realizadas durante un tiempo definido?
2. ¿Cuál es el porcentaje de preguntas que han sido respondidas durante un tiempo definido?
3. ¿Cuál es el día de la semana y mes del año con mayor cantidad de preguntas y respuestas realizadas?
4. ¿Cuáles son los usuarios que tienen mayor reputación?
5. ¿Cuáles son los usuarios que han resuelto la mayor cantidad de preguntas?
6. ¿Cuáles preguntas que han tenido la mayor cantidad de visitas?
7. ¿De qué tecnologías son las preguntas que más se realizan?

8. ¿Cuáles son las preguntas mayormente marcadas como favoritas y con mayor puntaje que fueron creadas en un periodo de tiempo?
9. ¿Cómo fue el comportamiento de las preguntas y respuestas hechas durante el periodo de pandemia con respecto a años anteriores?
10. ¿Cuáles son las preguntas que han tenido mayor retroalimentación?

Estas son solo algunas de las preguntas que podríamos responder, se espera que el modelo dimensional permita a los usuarios analíticos poder contestar más tipos de preguntas.

9.7 Modelo dimensional propuesto

Luego de la aplicación del modelado dimensional ante los requerimientos iniciales presentamos a continuación el resultado:

- ✓ ***Esquema Estrella 1°***

1. **Proceso de negocio:** Pregunta Hecha.
2. **Granularidad:** 1 Fila corresponde a una pregunta hecha.
3. **Dimensiones:** question, date, time, user, tag_bridge, tag.
4. **Métricas:** answer_count, score, comment_count, revisión_count, favorite_count, view_count.

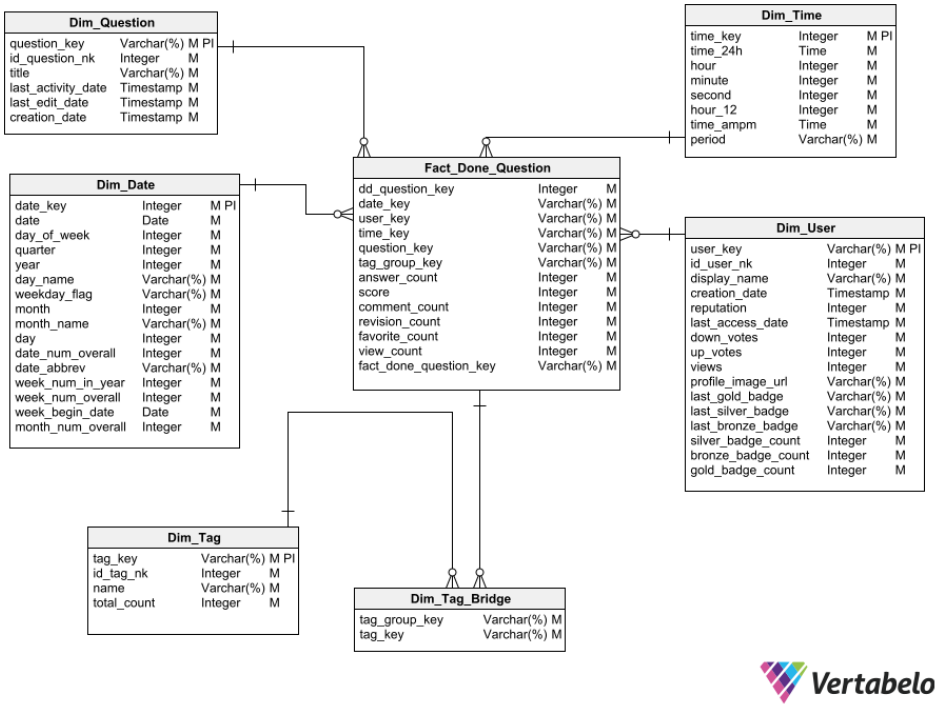

Figura 18. Modelo dimensional final para el proceso de negocio pregunta hecha

✓ ***Esquema Estrella 2°***

1. **Proceso de negocio:** Respuesta Hecha.
2. **Granularidad:** 1 Fila corresponde a una respuesta hecha a una pregunta.
3. **Dimensiones:** answer, date, time, user.
4. **Métricas:** score, comment_count, revisión_count.

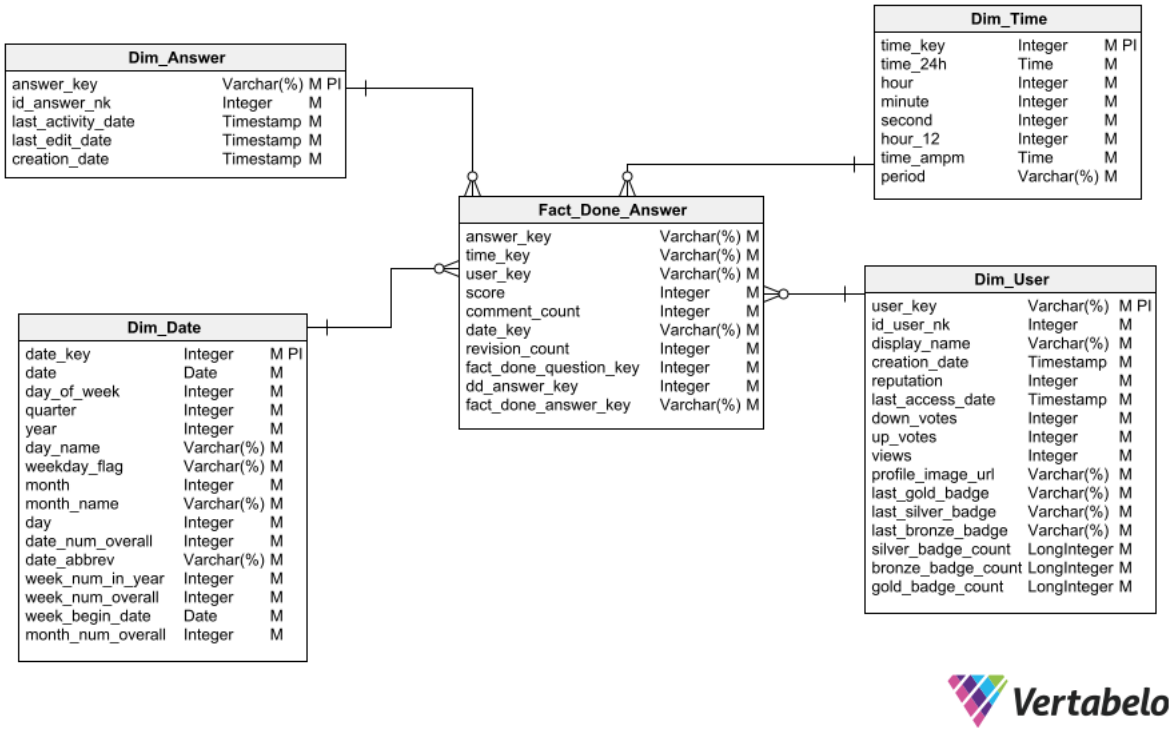

Figura 19. Modelo dimensional final para el proceso de negocio respuesta hecha

9.8 Tipos de Fact table y Dimensiones Utilizadas.

Los correspondientes elementos de los modelos dimensionales presentados anteriormente se describen a continuación:

Dimensiones Conformadas: Todas las dimensiones que se han creado son conformados, lo cual significa que todas las dimensiones son compartidas con las fact tables que necesiten del contexto que brindan las dimensiones.

Fact table Transaccional: Las dos fact tables son transaccionales, es decir que cada registro en la fact table equivale a una transacción realizada.

A continuación, se presenta la Matrix de bus para ver mejor la interacción de las dimensiones con las fact tables:

Procesos de negocios	DIMENSIONES						
	Answer	Date	Question	Tag	Tag_bridge	Time	User
Pregunta hecha		☑	☑		☑	☑	☑
Respuesta hecha	☑	☑				☑	☑

Tabla 40. Matrix de buz que describe la interacción entre Fact_tables y dimensiones

Proceso de Negocio: Pregunta Hecha	
Elementos	***Tipo***
Fact Done Question	Transaccional
Dim Answer	Conformada
Dim Users	Conformada
Dim Date	Conformada
Dim Time	Conformada
Dim Tag	Conformada
Dim Tag Brigde	Conformada

Tabla 41. Determinación de tipos por cada Fact table y dimensión

Proceso de Negocio: Respuesta Hecha	
Elementos	***Tipo***
Fact Done Answer	Transaccional
Dim Answer	Conformada
Dim Users	Conformada
Dim Date	Conformada

Dim Time	Conformada

Tabla 42. Matrix de buz que describe la interacción entre Fact_tables y dimensiones

9.9 Mappings por tabla

Nomenclatura del campo Source:
[Dataset]=> [Tabla] => [Campo]

Tabla 43. Nomenclatura utilizada para el mapping por tabla

9.9.1 Dim Question.

- **Descripción:** Guarda el contexto de la pregunta formulada.
- **Granularidad:** un registro representa un post de pregunta.
- **Política de unicidad:** el etl búcara las preguntas y le asignara una llave subrogada cuando esta pregunta no está almacenada en la dimensión.
- **Política de nulidad:** Todos los campos son requeridos.
- **Política de SCD:** Todos los campos serán Slowly Changing Dimension tipo uno.

Column name	Display name	Type	Source	Comment	Sample
question_key	Question key	String	-	Surragate key generated	68d2e3f
id_question_nk	Id natural key	Integer	stackoverflow =>post_question=>id	Natural Key	4
title	Title	String	stackoverflow =>post_question=>title	-	Null pointer java
last_activity_date	Last activity date	Timestamp	stackoverflow =>post_question=>last_activity_date	-	27/06/2021
last_edit_date	Last edit date	Timestamp	stackoverflow =>post_question=>last_edit_date	-	27/06/2021
creation_date	Creation date	Timestamp	stackoverflow =>post_question=>creation_date	-	27/06/2021

Tabla 44. Descripción del mapping para la dimensión question

9.9.2 Dim Answer.

- **Descripción:** Guarda el contexto de la respuesta hecha.
- **Granularidad:** un registro representa un post de respuesta.
- **Política de unicidad:** el etl búcara las respuestas y le asignara una llave subrogada cuando esta respuesta no está almacenada en la dimensión.
- **Política de nulidad:** Todos los campos son requeridos.
- **Política de SCD:** Todos los campos serán Slowly Changing Dimension tipo uno.

Column name	Display name	Type	Source	Comment	Sample
answer_key	Answer key	String	-	Surragate key generated	68d2e3f
id_answer_nk	Id natural key	Integer	stackoverflow =>post_answer=>id	Natural Key	4
last_activity_date	Last activity date	Timestamp	stackoverflow =>post_answer=>last_activity_date	-	27/06/2021
last_edit_date	Last edit date	Timestamp	stackoverflow =>post_answer=>last_edit_date	-	27/06/2021
creation_date	Creation date	Timestamp	stackoverflow =>post_answer=>creation_date		

Tabla 45. Descripción del mapping para la dimensión de answer

9.9.3 Dim user.

- **Descripción:** Guarda el contexto de un usuario.
- **Granularidad:** un registro representa a un usuario.
- **Política de unicidad: el** etl búcara los usuarios y le asignara una llave subrogada cuando esta pregunta no está almacenada en la dimensión.
- **Política de nulidad:** Todos los campos son requeridos.
- **Política de SCD:** Todos los campos serán Slowly Changing Dimension tipo uno.

Column name	Display name	Type	Source	Comment	Sample
user_key	User key	String	-	Surragate key generated	68d2e3f

Id_user_nk	Id natural key	Integer	stackoverflow =>users=>id	Natural Key	4
display_name	Display name	String	stackoverflow =>users=>display_name	-	Henry
creation_date	Creation date	Timestamp	stackoverflow =>users=>creation_date	-	27/06/2021
reputation	Reputation	Integer	bigquery:stackoverflow =>users=>reputation	-	10
last_access_date	Last access date	Timestamp	stackoverflow =>users=>last_access_date	-	27/06/2021
down_votes	Down votes	Integer	stackoverflow =>users=>down_votes	-	0
up_votes	Up votes	Integer	stackoverflow =>users=>up_votes	-	1
views	Views	Integer	stackoverflow =>users=>views	-	1
profile_image_url	Profile image url	String	stackoverflow =>users=>profile_image_url	-	https://www.gravatar.com/avatar/730d47
last_gold_badge	Last gold badge	String	stackoverflow =>badges	Calculater ETL	student
last_silver_badge	Last silver badge	String	stackoverflow =>badges	Calculater ETL	supporter
last_bronze_badge	Last bronce badge	String	stackoverflow =>badges	Calculater ETL	editor
silver_badge_count	Silver badge count	Integer	stackoverflow =>badges	Calculater ETL	2
bronze_badge_count	Bronze badge count	Integer	stackoverflow =>badges	Calculater ETL	2
gold_badge_count	Gold badge count	Integer	stackoverflow =>badges	Calculater ETL	1

Tabla 46. Descripción del mapping para la dimensión de user

9.9.4 Dim tag.

- **Descripción:** Guarda las etiquetas del post.
- **Granularidad:** un registro representa una etiqueta
- **Política de unicidad: el** etl búcara las etiquetas y le asignara una llave subrogada cuando esta pregunta no está almacenada en la dimensión.
- **Política de nulidad:** Todos los campos son requeridos.
- **Política de SCD:** Todos los campos serán Slowly Changing Dimension tipo uno.

Column name	Display name	Type	Source	Comment	Sample
tag_key	Tag key	String	-	Surragate key generated	68d2e3f
id_tag_nk	Id natural key	Integer	stackoverflow =>tags=>id	Natural Key	1
name	Name	String	stackoverflow =>tags=>name	-	JavaScript
total_count	Total count	Integer	stackoverflow =>tags=>count	-	100

Tabla 47. Descripción del mapping para la dimensión de tag

9.9.5 Dim tag_bridge

- **Descripción:** El puente guarda la relación entre la fact table y la tabla correspondiente.
- **Granularidad:** un registro representa un grupo de etiquetas para una pregunta
- **Política de unicidad:** el etl construirá el puente en base a los registros de etiquetas y preguntas y le asignará una llave de grupo de etiquetas cuando no está almacenada en el puente.
- **Política de nulidad:** Todos los campos son requeridos.

Column name	Display name	Type	Source	Comment	Sample
tag_group_key	Tag group key	String	-	Surragate key generated	68d2e3f
tag_key	Tag key	String	-	Foreign key	-

Tabla 48. Descripción del mapping para la dimensión tag_brigde

9.9.6 Dim Time.

- **Descripción:** Esta dimensión guarda datos relacionados al tiempo
- **Granularidad:** un registro representa el tiempo de un día.
- **Política de unicidad:** un registro representa el tiempo de un día.
- **Política de nulidad:** Todos los campos son requeridos.
- **Política de SCD:** Todos los campos serán Slowly Changing Dimension tipo cero.

Column name	Display name	Type	Source	Comment	Sample
time_key	Time key	Integer	-	Surragate key generated	127
time_24h	Time 24h	Time	-	-	00:01:27
hour	Hour	Integer	-	-	0
minute	Minute	Integer	-	-	1
second	Second	Integer	-	-	27
hour_12	Hour 12	Integer	-	-	12
time_ampm	AM-PM	Time	-	-	12:01:27 a.m
Period	Period	String	-	-	morning

Tabla 49. Descripción del mapping para la dimensión time

9.9.7 Dim Date.

- **Descripción:** Esta dimensión guarda datos relacionados al dia
- **Granularidad:** un registro representa un día del año
- **Política de unicidad:** un registro representa un día del año
- **Política de SCD:** Todos los campos utilizaran SCD cero.

Column name	Display name	Type	Source	Comment	Sample
date_key	Date key	Integer	-	Surragate key generated	20130101
date	Date	Date	-	-	01/01/2013
day_of_week	Day of week	Integer	-	-	1
Quarter	Quarter	Integer	-	-	2
Year	Year	Integer	-	-	2013

day_name	Day name	String	-	-	Thursday
weekday_flag	Weekday flag	String	-	-	Weekday
month	Month	Integer	-	-	1
month_name	Month name	String	-	-	January
day	Day	Integer	-	-	1
date_num_overall	Date num overall	Integer	-	-	2
date_abbrev	Date abbrev	String	-	-	Wed
week_num_in_year	Week num in year	Integer	-	-	2
week_num_overall	Week num overall	Integer	-	-	2
week_begin_date	Week begin date	Date	-	-	01/01/2021
month_num_overall	Month num overall	Integer	-	-	2

Tabla 50. Descripción del mapping para la dimensión time

9.9.8 Fact_Done_Question

- **Descripción:** contiene todos los eventos que ocurren en el proceso de negocio de la pregunta formulada.
- **Granularidad:** un registro representa una pregunta realizada
- **Política de unicidad:** El Etl construirá un registro en la fact table basado en las nuevas preguntas que se realicen en StackOverflow.
- **Política de nulidad:** Todos los campos son requeridos.
- **Política de ausencia de contexto:** Cuando una dimensión no aplique a una fila de la fact, se definirá una llave foránea para indicar la ausencia de datos de la misma.

Column name	**Display name**	**Type**	**Source**	**Comment**	**Sample**
dd_question_key	Dd Question key	Integer	Dim_question=>id_question_nk	**-**	**4**
time_key	Time key	Integer	Dim_Time=>time_key	Foreign key pointing to Dim_Time	-

date_key	Date key	Integer	Dim_Date=>date_key	Foreign key pointing to Dim_Date	-
user_key	User key	String	Dim_User=>user_key	Foreign key pointing to Dim_User	-
question_key	Question key	String	Dim_Question=>question_key	Foreign key pointing to Dim_Question	-
tag_group_key	Tag group key	String	Dim_Tag_Bridge=>tag_group_key	Foreign key pointing to Dim_Tag_Bridge	-
answer_count	Answer count	Integer	stackoverflow =>post_question=>answer_count	-	2
view_count	View count	Integer	stackoverflow =>post_question=>view_count	-	10
score	Score	Integer	stackoverflow =>post_question=>score	-	5
comment_count	Comment count	Integer	stackoverflow =>post_question=>comment_count	-	4
revision_count	Revision count	Integer	stackoverflow =>post_history	Calculater ETL	1
favorite_count	Favorite count	Integer	stackoverflow =>post_question=>favorite_count	-	2
fact_done_question_key	Fact done question key	String	-	Primary key generated to fact_done_question	abd-gr7

Tabla 51. Descripción del mapping para la Fact_done_question

9.9.9 Fact_Done_Answer

- **Descripción:** Contiene todos los eventos que ocurren en el proceso de negocio respuesta hecha
- **Granularidad:** un registro representa una respuesta
- **Política de unicidad:** El Etl construirá un registro en la fact table basado en las nuevas respuestas a las preguntas hechas en StakOverflow.
- **Política de nulidad:** Todos los campos son requeridos.
- **Política de ausencia de contexto:** Cuando una dimensión no aplique a una fila de la fact, se definirá una llave foránea para indicar la ausencia de datos de la misma.

Column name	Display name	Type	Source	Comment	Sample
time_key	Time key	Integer	Dim_Time=>time_key	Foreign key pointing to Dim_Time	-
date_key	Date key	Integer	Dim_Date=>date_key	Foreign key pointing to Dim_Date	-
user_key	User key	String	Dim_User=>user_key	Foreign key pointing to Dim_User	-
score	Score	Integer	stackoverflow =>post_answer=>score	-	5
comment_count	Comment count	Integer	stackoverflow =>post_answer=>comment_count	-	4
revision_count	Revision count	Integer	stackoverflow =>post_history	Calculater ETL	1
fact_done_question_key	Fact done question key	String	Fact_Done_Question=>fact_done_question_key	Foreing Key pointing to fact_done_question_key	ab-cef
fact_done_answer_key	Fact done answer key	String	-	Primary key generated to Fact_Done_Answer	hi-jkl
dd_answer_key	DD answer key	Integer	Dim_Answer=>id_answer_nk	-	7

Tabla 52. Descripción del mapping para la Fact_done_answer

9.10 Arquitectura y selección de herramientas para la construcción del Data Lakehouse implementada en el proyecto.

9.10.1 Arquitectura.

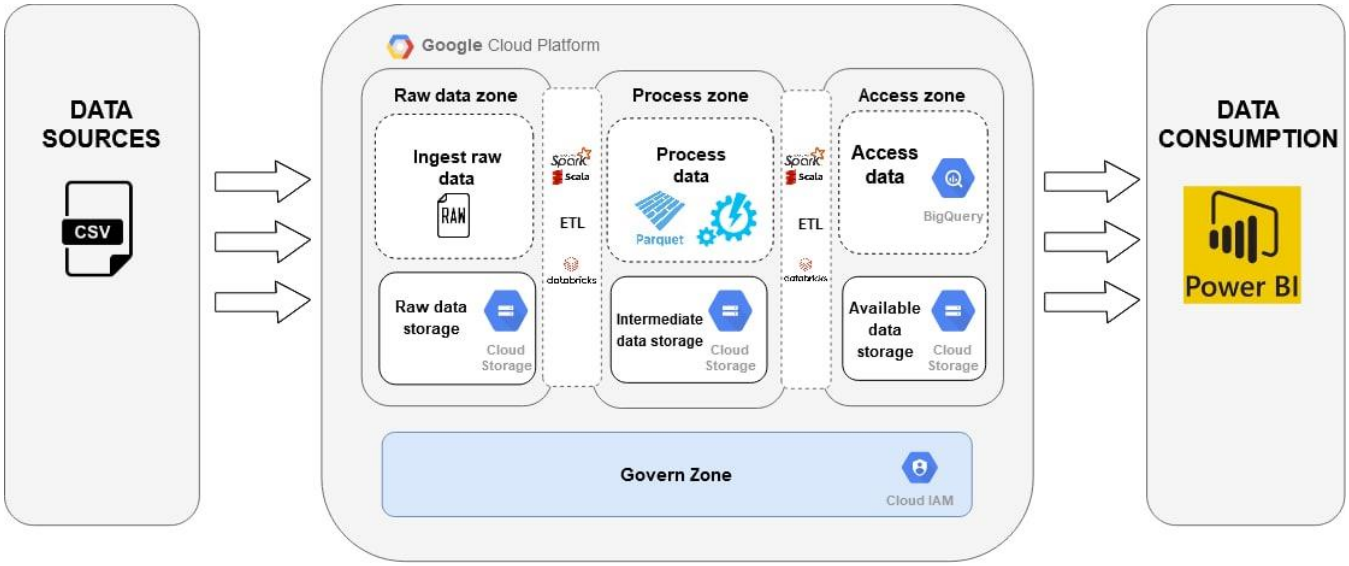

Figura 20. Arquitectura del Data-Lake implementado

9.10.2 Descripción de sus componentes.

Componente	Descripción
Data Source	Constituyen las principales fuentes de datos de las cuales se alimenta al data lake. En términos generales son múltiples fuentes de datos, para el proyecto se utilizó un dataset como única fuente de datos en formato csv.
Raw data zone	Capa inicial de la arquitectura en la que los datos son ingestados sin transformaciones y conservando su formato nativo, es decir los datos almacenados están almacenados tal cual como los genera la fuente de datos.
Process zone	Capa secundaria de la arquitectura que nos permite almacenar todos los datos que han sufrido ya sea un pre procesamiento o un procesamiento final, para que estos datos sean utilizados para la correspondiente construcción de las dimensiones y Fact tables.
Access zone	Capa final de la arquitectura en la que se almacenan los datos listos para ser consumidos, es decir se almacena data perteneciente a las respectivas dimensiones y Fact tables que conformaran el modelo dimensional final, esta capa permite a los usuarios utilizar la data para múltiples propósitos.
Govern zone	Constituye uno de los principales elementos de la arquitectura ya que permite aplicar un conjunto de reglas, políticas para poder administrar correctamente los datos dentro del data lake, logrando con esto la seguridad de tener todos los datos lo suficientemente ordenados. La zona de gobierno es aplicada a nivel de las tres capas.
Data consumption	Área final donde encontraremos los correspondientes dashboards consumiendo los datos para dar respuesta a los requerimientos analíticos del negocio, cabe

	destacar que en esta región los usuarios analíticos se encontraran consumiendo los datos a través de diferentes herramientas para la construcción de los reportes.

Tabla 53. Descripción de los componentes de la arquitectura del Data-Lake implementado

9.10.3 Descripción de los productos seleccionados para la arquitectura.

a. Cloud Storage.

Google Cloud Storage es un servicio de almacenamiento de archivos en línea restful para almacenar y acceder a datos en la infraestructura de Google cloud Platform. El servicio combina el rendimiento y la escalabilidad de la nube de Google con capacidades avanzadas de seguridad y uso compartido.

Figura 21. Logo de Google Cloud Storage

b. **Apache Spark**.

Es una plataforma que ha sido diseñada específicamente para el procesamiento de grandes cantidades de datos, es decir datos provenientes de diferentes fuentes de datos, con una amplia variedad, volumen y velocidad.

Figura 22. Logo de Apache Spark.

c. Scala.

Scala es un lenguaje de programación multi-paradigma diseñado para expresar patrones comunes de programación en forma concisa, elegante y con tipos seguros. Integra sutilmente características de lenguajes funcionales y orientados a objetos.

Figura 23. Logo de Scala

d. Databricks.

Databricks es una herramienta de ingeniería de datos basada en la nube líder en la industria que se utiliza para procesar y transformar cantidades masivas de datos y explorar los datos a través de modelos de aprendizaje automático.

Figura 24. Logo de databricks

e. Cloud IAM.

El servicio Google Cloud Identity and Access Management (IAM) permite crear y administrar permisos para los recursos de Google Cloud. Cloud IAM unifica el control de acceso para los servicios de Google Cloud en un solo sistema y presenta un conjunto coherente de operaciones.

Figura 25. Logo de Google Cloud IAM

f. **BigQuery**.

BigQuery **es** un almacén de datos de Google de bajo coste y totalmente administrado que permite extraer analíticas de petabytes de datos. **Es** autónomo, por lo que no **es** necesario gestionar ninguna infraestructura ni contar con un administrador de bases de datos.

Figura 26. Logo de Google BigQuery

g. Power BI.

Es una herramienta potencial dentro del mercado que permite la implementación de visualizaciones para poder construir los correspondientes dashboards, es super versátil ya que se integra con diferentes tipos de fuentes de datos en diferentes tecnologías.

Figura 27. Logo de Power BI

9.11 Descripción de estructura de los ETL implementados.

Los ETL´s implementados fueron construidos con el lenguaje de programación Scala, utilizando la tecnología de Apache Spark, dichos ETLs fueron desplegados en la plataforma de Databricks en su versión Community.

Los ETL's fueron divididos en tres etapas:

- **Raw layer o raw zone**: Etapa de cargado de datos.
- **Staging layer o process zone**: Etapa de transformacion de los datos
- **Presentation layer o Access zone**: Etapa de refinamiento de datos.

En Databricks se creó la siguiente estructura de carpetas para almacenar los ETLs.

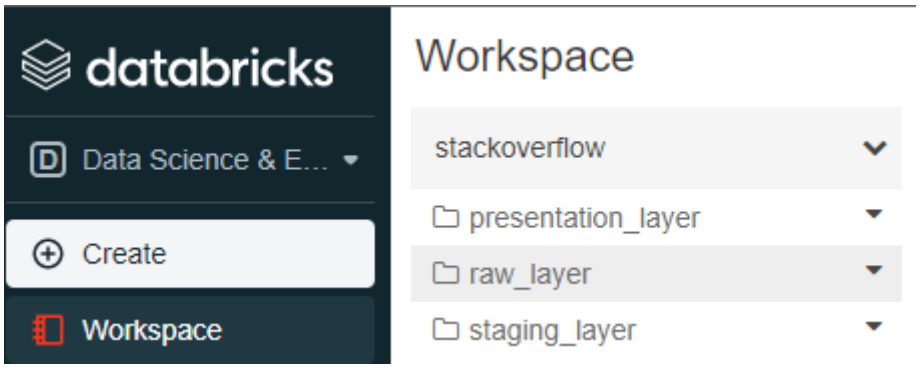

Figura 28. División de etapas en Databricks

Como destino para la data transformada en los ETLs se creó un Bucket en Cloud Storage con la siguiente división de carpetas:

Nombre	Tamaño	Tipo
presentation-layer/	—	Carpeta
raw-layer/	—	Carpeta
staging-layer/	—	Carpeta

Figura 29. Bucket en Cloud Storage

Raw layer

En esta etapa se desarrollaron los ETLs para la carga y limpieza de los datos, como fuente de entrada se los tenían archivos CSVs almacenados en un bucket de S3 en Amazon Web Services y como destino de esos datos ya limpiados se tenia un bucket con sus respectivas particiones en Google Cloud Platform (GCP), especificamente en Cloud Storage. Ver Anexo 1: Raw Tag ETL para más detalles de un ETL de la capa Raw.

Figura 30. Databricks, Lista de ETLs en raw Layer

Nombre	Tamaño	Tipo
badges.parquet/	—	Carpeta
comments.parquet/	—	Carpeta
post_answer.parquet/	—	Carpeta
post_history.parquet/	—	Carpeta
posts_moderator_nomination.parqu	—	Carpeta
posts_orphaned_tag_wiki.parquet/	—	Carpeta
posts_questions.parquet/	—	Carpeta
posts_tag_wiki_excerpt.parquet/	—	Carpeta
posts_wiki_placeholder.parquet/	—	Carpeta
stackoverflow_posts.parquet/	—	Carpeta
tags.parquet/	—	Carpeta
users.parquet/	—	Carpeta
votes.parquet/	—	Carpeta

Figura 31. Cloud Storage, Data escrita por los ETLs de Raw layer

Staging Layer

En esta etapa se procesan los datos encaminados a crear dimensiones, por lo cual se hacen combinaciones de tablas de la raw layer para crear nuevas tablas que contengan los campos de las dimensiones propuestas. La lectura y escritura de datos se realizaron en un bucket en Cloud Storage.

Ver Anexo 2: Staging Tag ETL para más detalles de un ETL de la capa Staging.

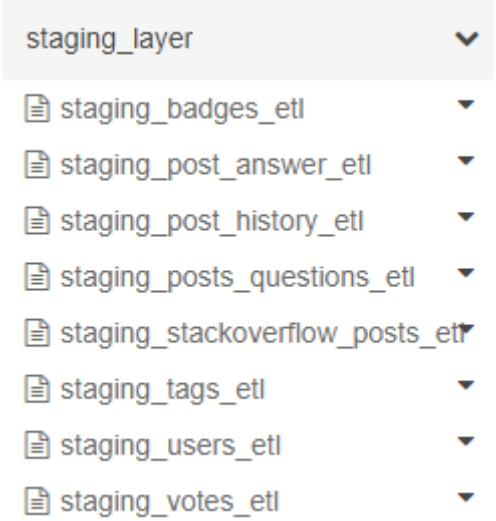

Figura 32. Databricks, Lista de ETLs en Staging Layer

Nombre	Tamaño	Tipo
badges.parquet/	—	Carpeta
post_answer.parquet/	—	Carpeta
post_history.parquet/	—	Carpeta
posts_questions.parquet/	—	Carpeta
stackoverflow_posts.parquet/	—	Carpeta
tag_bridge.parquet/	—	Carpeta
tags.parquet/	—	Carpeta
users.parquet/	—	Carpeta
votes.parquet/	—	Carpeta

Figura 33. Cloud Storage, Data escrita por ETLs de Staging Layer

Presentation Layer

En esta última etapa se forman los esquemas estrellas, se crean las llaves subrogadas de las dimensiones y llaves compuestas de las fact tables. La escritura de estos modelos dimensionales fue en la carpeta de presentation layer en el bucket designado en Cloud Storage.

Ver Anexo 3: Presentation Tag ETL para más detalles de un ETL de la capa Presentation.

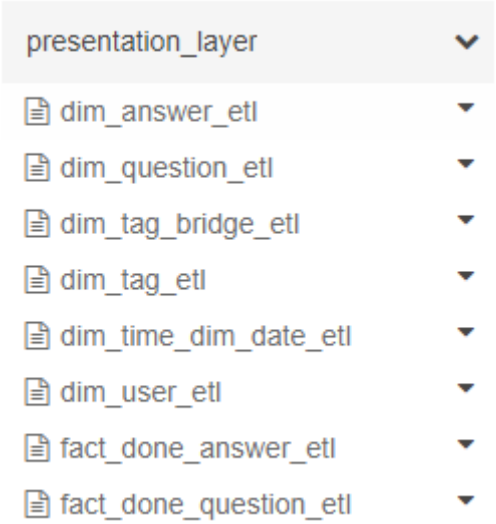

Figura 34. Databricks, Lista de ETLs de presentation layer

Nombre	Tamaño	Tipo
dim_answer.parquet/	—	Carpeta
dim_date.parquet/	—	Carpeta
dim_question.parquet/	—	Carpeta
dim_tag.parquet/	—	Carpeta
dim_tag_bridge.parquet/	—	Carpeta
dim_time.parquet/	—	Carpeta
dim_user.parquet/	—	Carpeta
fact_done_answer.parquet/	—	Carpeta
fact_done_question.parquet/	—	Carpeta

Figura 35. Cloud Storage, Dimensiones y fact tables escrita por los ETLs de presentation layer

Big Query

Como paso fundamental para que los datos estén listos para ser consumidos, Google Cloud Platform provee de Big Query.

Por tanto, las dimensiones y fact tables almacenados en Cloud Storage se migraron a Big Query para ser accedidas con facilidad a través de herramientas externas de análisis o visualización de datos.

9.12 Selección de la herramienta para la construcción de visualizaciones.

Dentro del mercado existen un enorme abanico de herramientas que nos permite implementar o construir visualizaciones para poder formar el correspondiente dashboard. Pero no todas pueden satisfacer las potenciales necesidades que se desean implementar. La elección de una con respecto a otra dependerá inicialmente si las herramientas nos permiten implementar lo que deseamos construir y en segundo lugar como punto más importante es verificar y comprobar el grado de versatilidad con la que cuenta la herramienta para poder integrarse super bien con otros ecosistemas de tecnológicas diferentes.

Es por ello que para poder construir los correspondientes dashboard que nos permitirán contestar de manera concreta las preguntas analíticas que se plantearon inicialmente hemos seleccionado una herramienta de Microsoft el cual es Power BI.

Power BI es una herramienta super potente que se integra muy bien con otras fuentes de datos en diferentes tecnologías, esto la hace una herramienta super versátil, dentro de ese abanico de fuentes de datos. Es importante mencionar que coexiste super bien con nuestro almacén de datos oncloud el cual es BigQuery, a través del cual vamos a poder desde Power BI a través del modo de consumo de datos DirectQuery conectarnos directamente a los datos almacenados y poder alimentar a las correspondientes visualizaciones que conforman el dashboard. A continuación, se muestra una figura ilustrativa a acerca del área de trabajo del cual nos provee la herramienta seleccionada como tal.

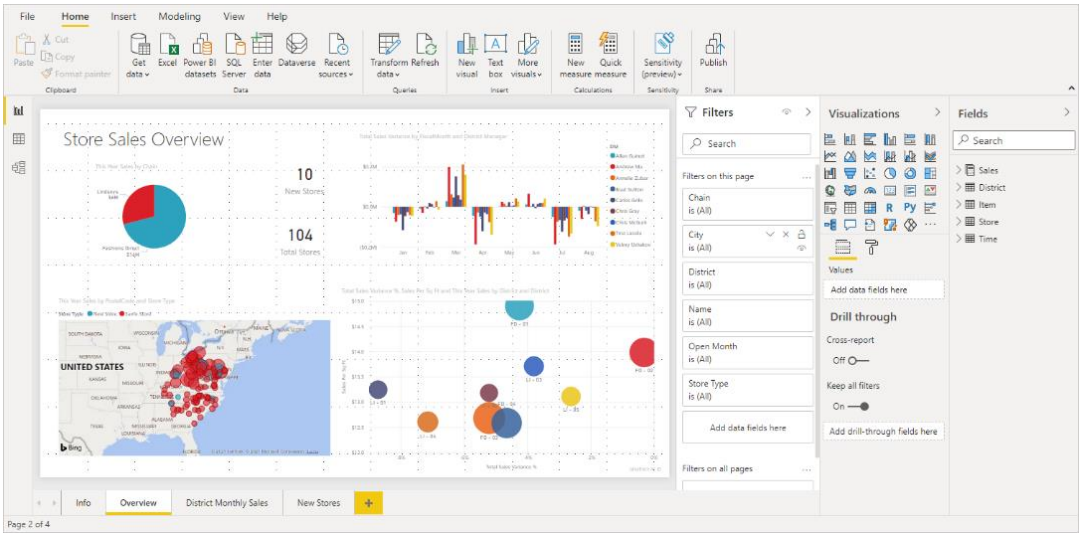

Figura 36. Vista de Informe el cual permite la construcción de dashboards en Power BI

9.13 Desarrollo de dasboards para la resolución de las necesidades analíticas.

9.13.1 ¿Cuál es el total de preguntas realizadas durante un tiempo definido?

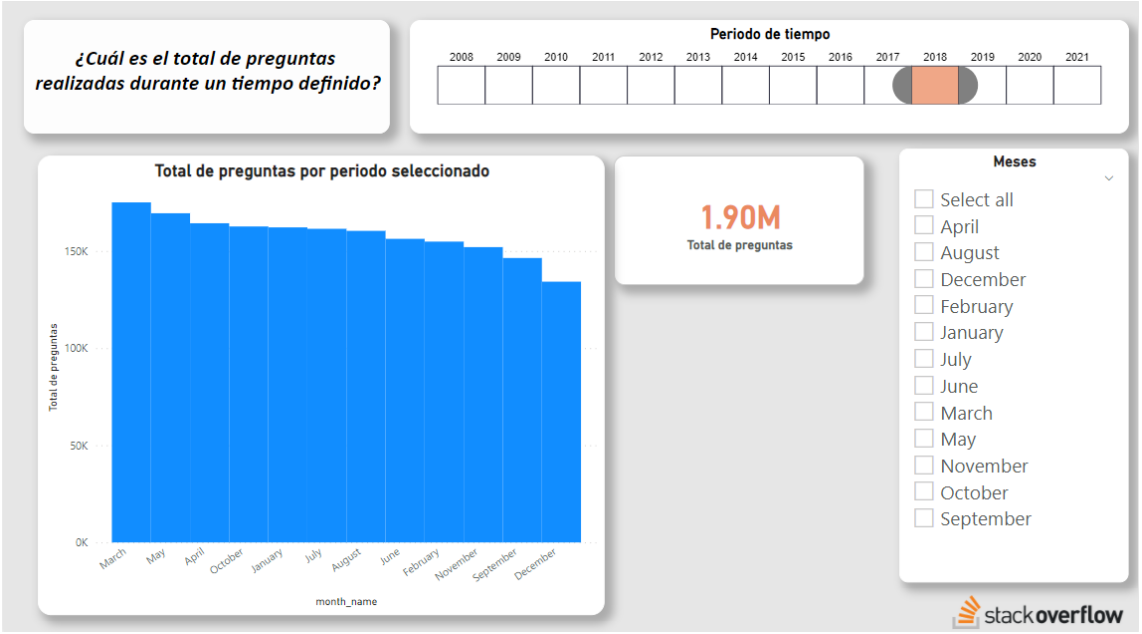

Figura 37. Dashboard resultante que responde ala pregunta ¿Cuál es el total de preguntas realizadas durante un tiempo definido?

Descripción:

Este dashboard nos permite seleccionar un periodo de tiempo, ya sea el año y meses deseados. Una vez definido el periodo de tiempo, el grafico nos muestra el total de preguntas realizadas durante ese periodo.

El grafico tiene habilitado una jerarquía de tiempo, por lo que nos permite ver el total de preguntas ya sea por año, trimestre, mes o día. Para la imagen mostrada, el grafico nos muestra el total de preguntas realizadas por mes para el año 2018.

9.13.2 ¿Cuál es el porcentaje de preguntas que han sido respondidas durante un tiempo definido?

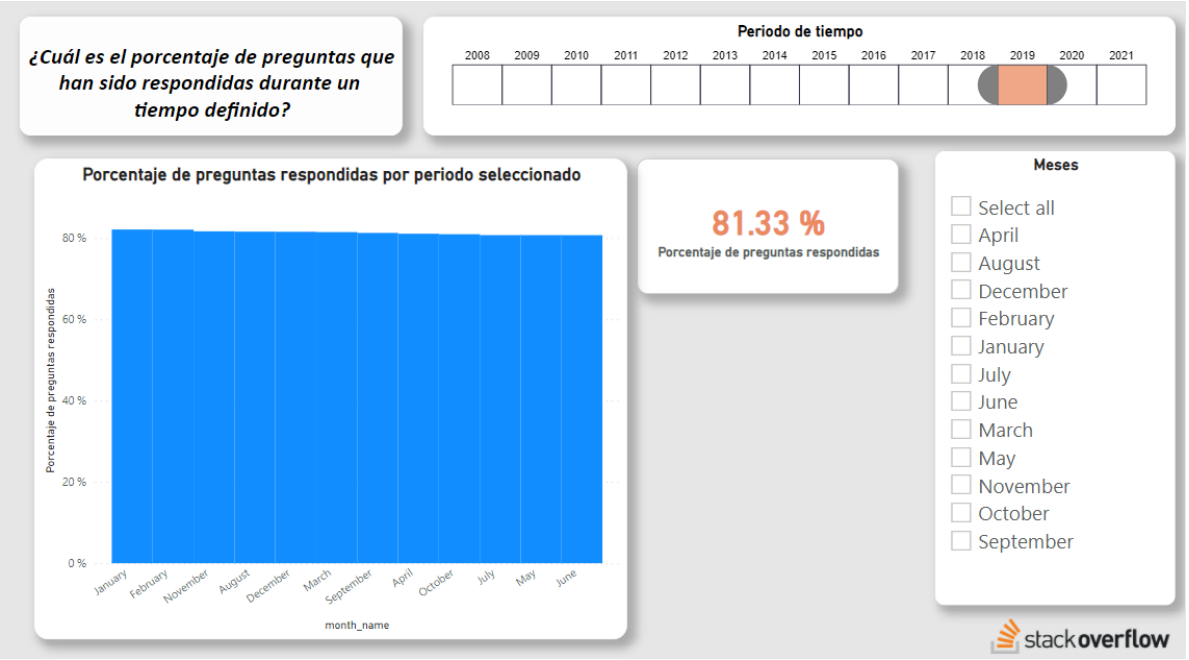

Figura 38. Dashboard resultante que responde ala pregunta ¿Cuál es el porcentaje de preguntas que han sido respondidas durante un tiempo definido?

Descripción:

Este dashboard de igual manera nos permite seleccionar un periodo de tiempo, ya sea un año, varios años y los meses deseados. Una vez definido el periodo de tiempo, el grafico nos muestra el porcentaje de preguntas que han sido respondidas en ese periodo seleccionado. Es decir que, si para un tiempo T se realizan 100 preguntas, pero solo 80 son contestadas, el grafico nos mostraría un porcentaje del 80% para ese tiempo T.

El grafico también tiene habilitado una jerarquía de tiempo, por lo que nos permite ver el porcentaje de preguntas respondidas por año, trimestre, mes o día. Para la imagen mostrada, el grafico nos muestra porcentaje de preguntas respondidas por mes para el año 2019.

9.13.3 ¿Cuál es el día de la semana y el mes del año con mayor cantidad de preguntas y respuestas realizadas?

Figura 39. Dasboard resultante que responde ala pregunta ¿Cuál es el día de la semana y el mes del año con mayor cantidad de preguntas y respuestas realizadas?

Descripción:

En la figura 39 se observa los gráficos que comparan las preguntas hechas contra las respuestas hechas, por lo cual se puede ver cuál es el mes y el día de la semana que más actividad hubo en la plataforma de Stackoverflow, además se ven tarjetas informativas que nos dicen el total de respuestas, de preguntas y el porcentaje de preguntas respondidas durante el periodo de tiempo definido.

Para entender mejor los gráficos de dispersión se tiene que: entre más a la derecha esta un punto mayor cantidad de preguntas y entre más arriba mayor cantidad de respuestas, por lo que el mejor punto debe estar situado arriba a la derecha y el peor abajo a la izquierda.

9.13.4 ¿Cuáles son los usuarios que tienen mayor reputación?

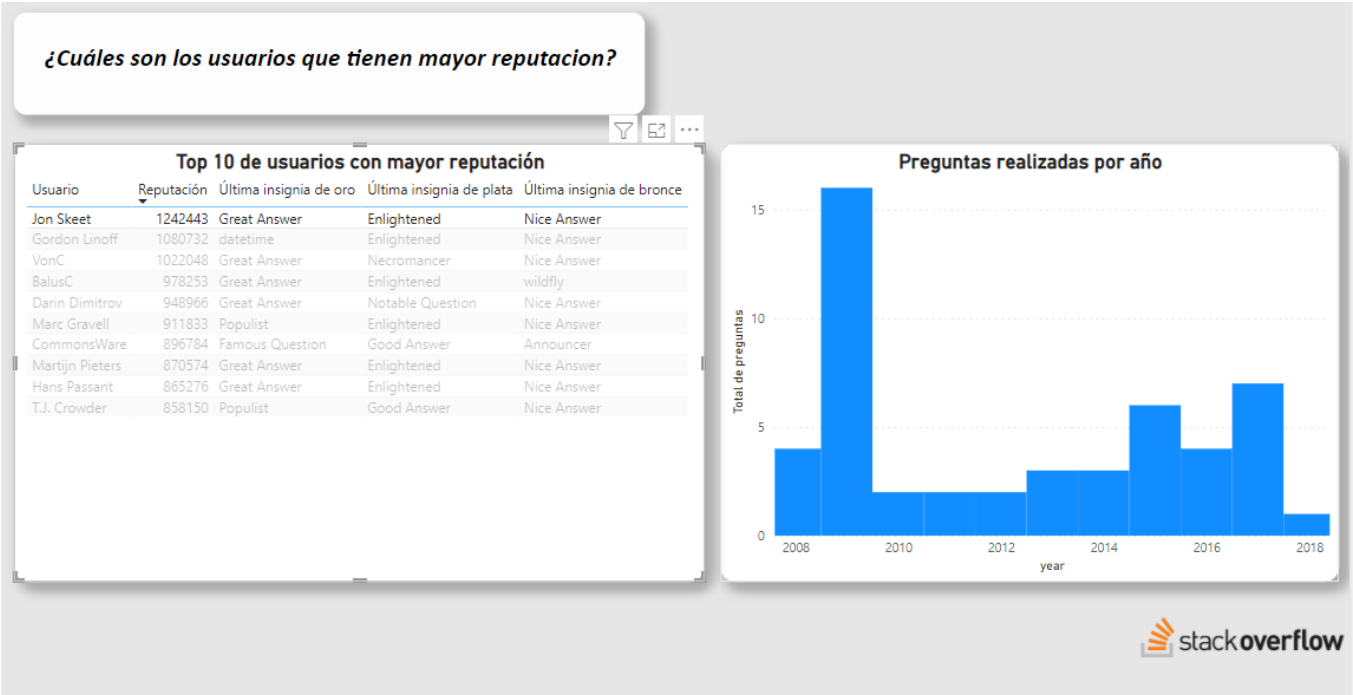

Figura 40. Dasboard resultante que responde a la pregunta ¿Cuáles son los usuarios que tienen mayor reputación?

Descripción:

Este dashboard nos muestra a los 10 usuarios que tienen mayor reputación, la última insignia de oro, plata y bronce que ha ganado, de igual manera al seleccionar a un usuario del top 10, automáticamente la visualización de ''preguntas realizadas por año'' es actualizado, de tal manera que se pueden observar las preguntas que ese usuario ha realizado durante los años.

9.13.5 ¿Cuáles usuarios han resuelto mayor cantidad de preguntas?

Figura 41. Dasboard resultante que responde a la pregunta ¿Cuáles usuarios han resuelto mayor cantidad de preguntas?

Descripción:

En este dashboard se puede establecer un periodo de tiempo para saber cuáles son los usuarios que más han aportado respuestas para resolver las preguntas hechas. Se puede filtrar por año o años y por meses.

En el gráfico de barras están listados de mayor a menor aportación los nombres de los usuarios, a su derecha están algunas medidas interesantes como lo son: El total de respuestas hechas, los usuarios que han aportado una o más respuestas y un promedio de cuantas preguntas responden los usuarios.

En el caso específico de la figura 41 nos muestra los usuarios que más respondieron preguntas durante el año 2019, y se puede apreciar en el gráfico de barras la cantidad de respuestas que cada usuario aporto.

9.13.6 ¿Cuáles preguntas han tenido la mayor cantidad de visitas?

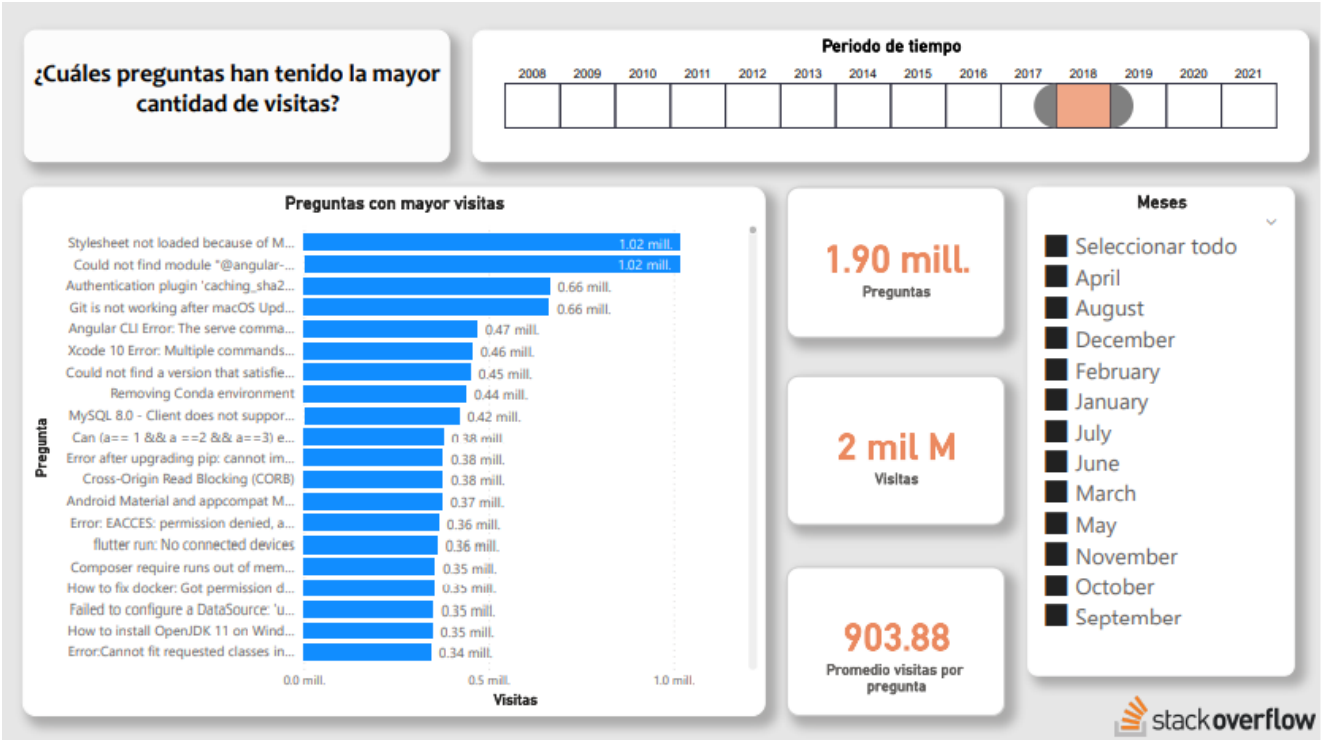

Figura 42. Dasboard resultante que responde a la pregunta ¿Cuáles preguntas han tenido la mayor cantidad de visitas?

Descripción:

Este dashboard nos muestra las preguntas que fueron hechas en un periodo definido ya sea por años o meses y que ahora tienen la mayor cantidad de visitas, y como medidas interesantes nos muestra la cantidad de preguntas totales hechas, el total de visitas obtenidas y el promedio de visitas por preguntas durante el periodo de tiempo definido.

9.13.7 ¿De qué tecnologías son las preguntas que más se realizan?

Figura 43. Dasboard resultante que responde a la pregunta ¿De qué tecnologías son las preguntas que más se realizan?

Descripción:

El dashboard nos presenta el top 10 de las tecnologías que más se preguntaron en un periodo de tiempo definido, estas tecnologías fueron la tendencia en la plataforma de Stackoverflow durante ese periodo. Como datos adicionales nos muestran el total de preguntas realizadas, el total de tecnologías abordadas y un promedio de cuantas etiquetas de tecnologías por pregunta.

9.13.8 ¿Cuáles son las preguntas mayormente marcadas como favoritas y con mayor puntaje que fueron creadas en un periodo de tiempo?

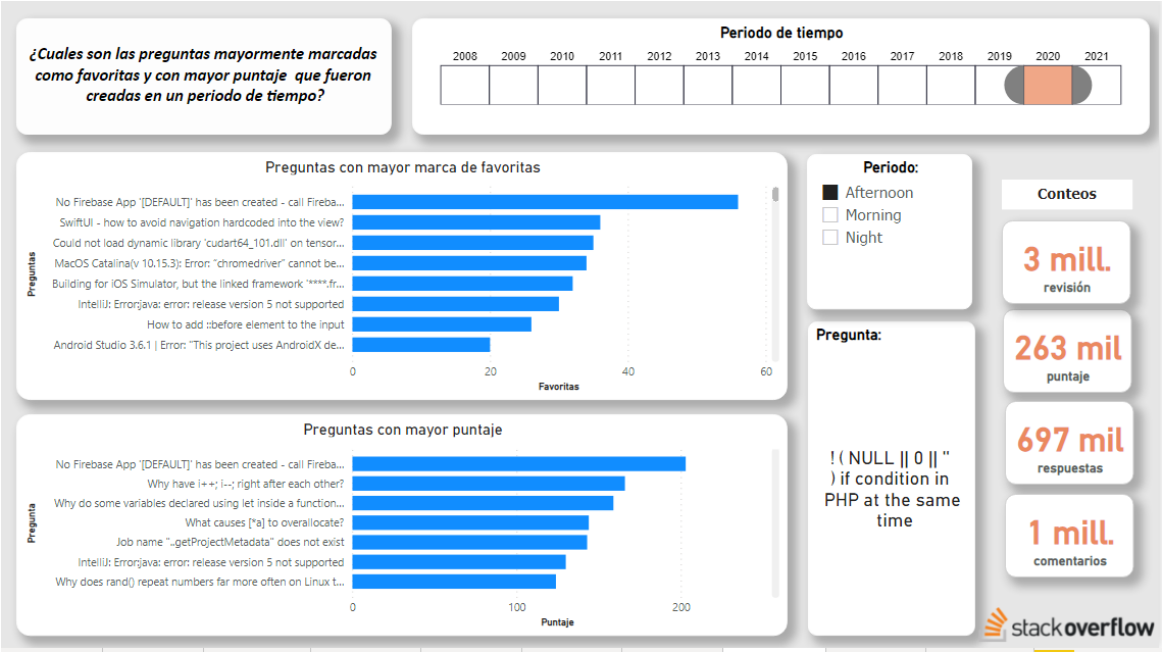

Figura 44. Dasboard resultante que responde a la pregunta ¿Cuáles son las preguntas mayormente marcadas como favoritas y con mayor puntaje que fueron creadas en un periodo de tiempo?

Descripción:

El dashboard nos da a conocer de manera clara cuales son las preguntas que mayormente han sido marcadas como favoritas y con mayor puntaje, analizadas desde un periodo de tiempo, en este caso para el año 2020, periodo de tiempo marcado por el aparecimiento de la pandemia. Básicamente el reporte realiza el análisis a través de la presentación de dos gráficos de barras horizontales, el primero de ellos representa todas aquellas preguntas mayormente marcadas como favoritas y la segunda muestra las preguntas con mayor puntaje.

Además, se presentan elementos adicionales que permiten incrementar la capacidad analítica del reporte tales como la segmentación del periodo en el que se realizan las puntuaciones o marcajes sobre las preguntas, presentación directa de la pregunta con mayor puntaje o marcaje de favorito. Si seleccionamos sobre los gráficos de barra las preguntas presentadas, además se presentan una serie de contadores que permiten dar a conocer la cantidad de revisiones, puntajes, respuestas y comentarios recibidos para la pregunta seleccionada.

9.13.9 ¿Comó fue el comportamiento de las preguntas y respuestas hechas durante el periodo de pandemia con respecto a años anteriores?

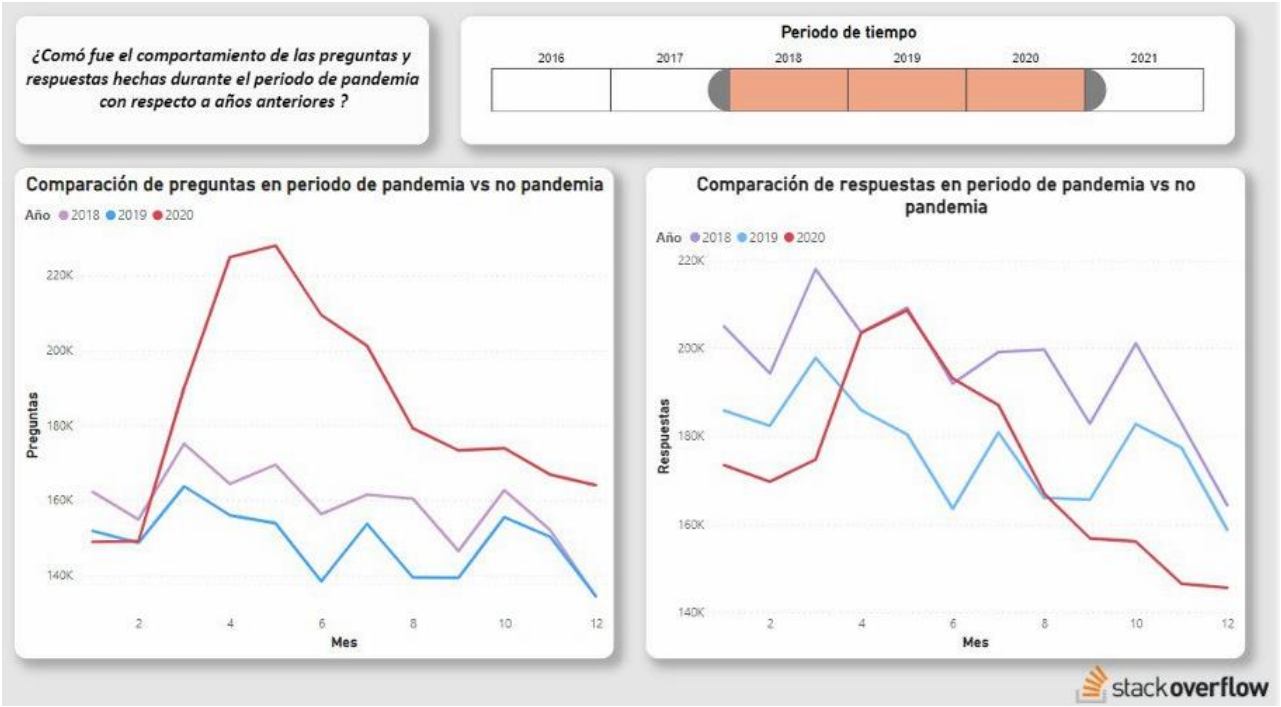

Figura 45. Dasboard resultante que responde a la pregunta ¿Comó fue el comportamiento de las preguntas y respuestas hechas durante el periodo de pandemia con respecto a años anteriores?

Descripción:

El dashboard nos da a conocer de manera concreta el comportamiento que se produjo con respecto a la creación de preguntas y respuestas durante el periodo de cuarentena el cual es para el año 2020, este comportamiento se ha comparado con respecto a años anteriores, tal como lo muestra la figura. Se tomó como referencia el periodo desde 2018 -2020, dicho periodo determina claramente en los gráficos de líneas como la generación de preguntas y respuestas tuvo su punto más alto en los primeros meses en los que las naciones implementaban sus respectivas cuarentenas, así como también el declive interesante de las preguntas y respuestas en los meses posteriores de ese periodo de tiempo.

A través de la interacción con los gráficos podremos observar datos relevantes tales como la cantidad de preguntas o respuestas hechas para cada año en análisis, así como también el número de mes y su correspondiente nombre. De tal forma que nos permitirá interactuar seleccionando sobre los gráficos en aquellos meses que sean de interés analítico tanto para responder la pregunta analítica como también para posteriores interrogantes que puedan surgir.

9.13.10 ¿Cuáles son las preguntas que han tenido una mayor retroalimentación?

Figura 46. Dasboard resultante que responde a la pregunta ¿Cuáles son las preguntas que han tenido una mayor retroalimentación?

Descripción:

El dashboard nos da a conocer de manera clara cuales son las preguntas que han tenido mayor retroalimentación durante un periodo. El reporte analiza la pregunta en base a dos perspectivas, la primera perspectiva se muestra a través del primer grafico de barras horizontal el cual muestra las preguntas con mayor cantidad de respuestas recibidas. El segundo grafico nos muestra las preguntas con mayor cantidad de comentarios, esto es debido a que la mayor retroalimentación que las preguntas pueden tener es a través de los comentarios y respuestas recibidas.

Adicionalmente se presentan elementos visuales tales como el segmentador de periodos en los que se realizan los comentarios y respuestas a las preguntas, además si seleccionamos en los gráficos de barras una de las preguntas a través de la tarjeta visual nos mostrara el nombre de la pregunta con mayor cantidad de respuestas y comentarios. Finalmente, el reporte está dotado con una serie de contadores que nos permite visualizar de manera directa la cantidad de revisiones, puntaje, respuestas y comentarios recibidos, nos arrojaran datos concretos si seleccionamos preguntas concretas el respectivo grafico como tal.

9.14 Repositorio de Github y prueba en vivo de reportes del proyecto

A través del enlace se podrá visualizar la versión técnica final del proyecto en un plano profesional con la versión más actual:

Repositorio en Github

A través del siguiente enlace se podrá visualizar una prueba en vivo de los correspondientes reportes anteriormente mostrados, cabe destacar que para poder visualizarlos se necesitará ingresar con la correspondiente cuenta institucional de la Universidad de El salvador.

Prueba en vivo de reportes del proyecto

9.15 Conclusiones.

1. Con el desarrollo de los modelos dimensionales para los procesos de negocio de preguntas y respuestas hechas, permitirá a la comunidad de usuarios de la plataforma Stack Overflow, poder conocer de primera mano el estado actual de la plataforma, en cuanto a patrones de comportamiento en la generación de preguntas y respuestas en periodos de tiempo específicos, patrones de comportamiento de los usuarios al momento de interactuar dentro de la plataforma, patrones relacionados al nivel de retroalimentación que las preguntas reciben, tecnologías mayormente consultadas durante periodos de tiempo determinados, entre otros variables de análisis que resultan de los procesos de negocio establecidos.

2. El desarrollo una solución de Big data, le permitirá a la comunidad de Stack Oveflow contar con un Data Lakehouse, que de manera directa dará solución a los principales requerimientos analíticos, que condujeron al desarrollo del proyecto. Es importante mencionar que los datos finales además podrán ser aplicados en soluciones de ciencia de datos, aplicación de Machine Learning, Inteligencias de Negocio, es decir que el modelo final está dotado de la capacidad para poder alimentar procesos superiores de análisis de datos, incrementando así el potencial de la solución.

3. La creación de reportes que dan solución concreta a los requerimientos de negocio, permite entender realmente el enorme aporte analítico, que los datos procesados le puedan dar a las organizaciones, las innumerables inquietudes, preguntas o necesidades que pueden surgir en el tiempo. Ya que un reporte permite transmitir de manera directa los hallazgos encontrados en los datos, hallazgos que pueden incrementar el éxito de la toma de decisiones estratégica.

9.16 Bibliografía.

- Arias, Á. (2015). *Computación en la Nube: 2ª Edición*. IT Campus Academy.
- Ariyansyah, M. F. H., Ridwan, A. Y., & Andreswari, R. (2016). Developing Business Intelligence System Based On Data Warehouse Using Pentaho For Procurement Process With Business Dimensional Life-cycle Methodology (case Study: Perum Bulog Divre Jabar). *eProceedings of Engineering*, *3*(2).
- Buyya, R., Calheiros, R. N., & Dastjerdi, A. V. (2016). Big Data Principles and Paradigms. Elsevier Gezondheidszorg.
- Foote, K., 2021. *A Brief History of Analytics - DATAVERSITY*. [online] DATAVERSITY. Available at: <https://www.dataversity.net/brief-history-analytics/#> [Accessed 15 January 2022].
- Guamán, M. A. A., Vaca, M. J. N., & Yuquilema, J. F. B. (2018). Mapeo sistematico de literatura de un data lake. *Revista mktDescubre-ESPOCH FADE*, (11), 50-66.
- Kimball, R., & Ross, M. (2011). *The data warehouse toolkit: the complete guide to dimensional modeling*. John Wiley & Sons.
- Ramos, S. (2016). Data Warehouse, data marts y modelos dimensionales. *Un pilar fundamental para la toma de decisiones. Albatera: SolidQ*.
- Ridwan, A. Y. (2015). Designing a multidimensional data warehouse for procurement processes analysis using business dimensional lifecycle method. In *Proceeding of The 8th International Seminar on Industrial Engineering and Management (ISIEM)*
- Rivadera, G. R. (2010). La metodología de Kimball para el diseño de almacenes de datos (Data warehouses). *Cuadernos de Ingeniería*, (5), 56-71.
- Segal, Troy. "Big Data." *Investopedia*, 5 July 2019, www.investopedia.com/terms/b/big-data.asp

9.17 Glosario de términos.

A

Análisis descriptivo: En el análisis descriptivo como su nombre lo indica se analizan y se contestan todas las preguntas acerca de los eventos que han ocurrido en un periodo de tiempo determinado.

Análisis de diagnóstico: En el análisis de diagnóstico básicamente se busca determinar las principales causas de eventos que han ocurrido en el tiempo.

Análisis predictivo: En el análisis predictivo se busca predecir los eventos en base a patrones, tendencias y excepciones encontradas en los datos.

Análisis prescriptivo: En el análisis prescriptivo se busca determinar las mejores acciones a seguir en base a los hallazgos del análisis predictivo.

B

Business Intelligente: Inteligencia empresarial, conjunto de estrategias, aplicaciones, datos, productos, tecnologías y arquitectura técnicas, los cuales están enfocados a la administración y creación de conocimiento sobre el medio, a través del análisis de los datos existentes en una organización

Big data: Conjunto de datos provenientes de múltiples fuentes de datos con gran volumen, variedad y velocidad, de estos datos se buscará explotarlos es decir darle el correspondiente sentido para que las organizaciones puedan tomar sus decisiones basadas en sus datos.

Biquery: Es un almacén de datos sin servidor totalmente administrado que permite un análisis escalable en petabytes de datos.

C

CSV: Tipo de documento en formato abierto sencillo para representar datos en forma de tabla, en las que las columnas se separan por comas y las filas por saltos de línea

Cluster: Conjunto de elementos computaciones que trabajan coordinadamente para lograr el objetivo principal, el cual es el procesamiento de grandes volúmenes de datos.

D

Dataset: Es una colección o grupo de datos relacionados.

Data Análisis: Es el proceso de examinación de datos para encontrar métricas, relaciones, patrones y tendencias.

Data Mart: Es una versión específica del almacén de datos (data warehouse) centrados en un tema o un área de negocio dentro de una organización. Son subconjuntos de datos con el propósito de ayudar a que un área específica dentro del negocio pueda tomar mejores decisiones.

Data Science: Ciencia de datos, es un campo interdisciplinario que involucra métodos científicos, procesos y sistemas para extraer conocimiento o un mejor entendimiento de datos en sus diferentes formas, ya sea estructurados o no estructurados.

Dasboard: Documento que contienen un conjunto de visualizaciones que interaccionan entre si, de tal manera que se desea presentar los principales hallazgos que son encontrados en los datos.

Databricks: herramienta cloud usada para procesar y realizar transformaciones sobre Big Data. También permite explorar estos datos usando modelos de inteligencia artificial. Está basada en Apache Spark.

Dataprep: Servicio inteligente de datos en la nube que te permite examinar, limpiar y preparar datos de forma visual para analizarlos y crear modelos de aprendizaje automático.

DirecQuery: Modo de consumo directo a bases de datos on cloud en el que la data se refrescara automáticamente por lo que personas trabajaran con la versión más actual de los datos.

Data Warehouse (DW): Almacén de datos, es una colección de datos orientada a un determinado ámbito, integrado, no volátil y variable en el tiempo, que ayuda a la toma de decisiones en la entidad en la que se utiliza. Se usa para realizar informes, análisis de datos, y se considera un componente fundamental de la inteligencia empresarial.

Dimensiones: Las dimensiones son tablas de datos que forman parte de los componentes de un modelo dimensional, en el que se busca describir el ¿Quién?, ¿Qué?, ¿Donde?, ¿Cuándo?,¿Como?, de tal forma que se permita almacenar el contextual textual que acompaña al evento de negocio que la Fact table se encuentra registrando.

Dataframe: Es una tabla de datos columnar, que habita en memoria, inmutable y que es particionada para ser distribuida a cada uno de los nodos ejecutores del cluster de Spark.

E

ETL: Es un proceso concreto que permite la correspondiente extracción, transformación y carga de los datos con los que nos encontremos trabajando, en el mundo de la ingeniería de datos toman especial relevancia ya que son los procesos a través de los cuales se realiza toda la lógica de procesamiento sobre los datos.

Executors: Unidad virtual con recursos computaciones que permiten llevar a cabo las correspondientes tareas que el nodo driver planifica sobre las porciones de datos que el mismo asigna a cada uno de los nodos ejecutores que conforman el cluster de Spark.

F

Fact table: Tabla de datos que pertenece a los elementos que conforman un modelo dimensional, siendo la misma el corazón del modelo ya que permite registrar los eventos del proceso de negocio al que pertenece, para medir el evento registrado calcula métricas, teniendo en su interior solamente llaves foráneas y métricas.

G

Google Cloud Platform: Es un conjunto de servicios de cómputo, almacenamiento, redes, macrodatos, aprendizaje automático e Internet de las cosas (IoT), así como herramientas de administración, seguridad y desarrollo en la nube.

Github: Es una plataforma de alojamiento de código para el control de versiones y la colaboración entre los miembros del proyecto.

M

Machine Learning (ML): Aprendizaje automático, es un subcampo de las ciencias de la computación y una rama de la inteligencia artificial, cuyo objetivo es desarrollar técnicas que permitan que las computadoras aprendan.

Metricas: medición numérica de un evento del proceso de negocio, potencializa la capacidad analítica del modelo ya que arrojan datos propios del evento que la Fact table registro, siendo la misma la principal fuente de alimentación para la construcción de visualizaciones.

O

OLAP (Online analytical processing): Procesamiento analítico en línea, es un método informático que permite a los usuarios extraer y consultar datos de manera fácil y selectiva para analizarlos desde diferentes puntos de vista.

P

Planeación del Proyecto: Es la primera fase o etapa del ciclo de vida de desarrollo de proyectos de datawarehouse en el que se determina el propósito del proyecto, sus objetivos específicos y el alcance del mismo, los principales riesgos y una aproximación inicial a las necesidades de información.

Power BI: Es una herramienta dentro del mercado que nos permite implementar cualquier tipo de visualizaciones e informes, cualquier tipo de lógica de transformación sobre los correspondientes datos, estando este en su versión desktop, services y mobile.

Perfilado de datos: Proceso de revisión de fuentes de datos, entendimiento de sus estructuras, contenido y relaciones.

Parquet: Formato de almacenamiento por defecto que es utilizado por spark , al ser este un formato columnar nos brinda todas las ventajas que el almacenamiento columnas nos ofrece , entre algunas de las ventajas que se aprovechan es la reducción significativa del storage comparado con otros formatos tradicionales , el tiempo de ejecución de queries es super rápido , se lee menos cantidad de datos para contestar a una querie etc.

S

Selección del producto: Es una de las etapas del ciclo de vida de desarrollo de proyectos, básicamente similar a una lista de compras para seleccionar productos que encajen en el marco del plan.

StackOverflow: Plataforma que se basa en una comunidad de usuarios que se encargan de gestionar la realización de preguntas y respuestas a los usuarios que desean resolver sus principales problemáticas que se han encontrado en su dia a dia laboral.

Spark: Plataforma perteneciente a los productos de apache, diseñada y construida específicamente para el procesamiento de grandes cantidades de datos.

Spark Sql: Es un módulo de Spark, el cual esta específicamente diseñado para el procesamiento de big data estructurada.

9.18 Anexos.

Anexo 1: Raw Tag ETL

A continuación, se presentará el ETL de la tabla Tag en la capa Raw:

```
import org.apache.spark.sql.functions._

val location = "s3://idt115-stackoverflow/dataprep/pm15007/tags.csv/"
val bucketName = "idt-stackoverflow"
val layerName = "raw-layer"
val tableName = "tags.parquet"
val destination = s"gs://$bucketName/$layerName/$tableName"

 //Reading tag table
val dirtyTable = spark.read
  .option("sep", ",")
  .option("header", false)
  .option("inferSchema", true)
  .csv(location)
// Set correct names to columns
val tags = dirtyTable
  .withColumnRenamed("_c0", "id")
  .withColumnRenamed("_c1", "tag_name")
  .withColumnRenamed("_c2", "count")
  .withColumnRenamed("_c3", "excerpt_post_id")
  .withColumnRenamed("_c4", "wiki_post_id")
//verify ids
val validTags = tags.filter(col("id").isNotNull)
//Writing tags table
validTags.write
  .option("compression", "snappy")
  .option("header", true)
  .mode("overwrite")
  .parquet(destination);
```

Anexo 2: Staging Tag ETL

```
import org.apache.spark.sql.functions._

val bucketName = "idt-stackoverflow"
val originLayer = "raw-layer"
val detinantionLayer = "staging-layer"
val tableName = "tags.parquet"
val origin = s"gs://$bucketName/$originLayer/$tableName"
val destination = s"gs://$bucketName/$detinantionLayer/$tableName"
//Reading raw tag table
val rawTags = spark.read.option("inferSchema", "true").parquet(origin)
//Drop unused columns
val tags = rawTags.drop("excerpt_post_id").drop("wiki_post_id")
//Writing staging tag table
tags.write
  .option("compression", "snappy")
  .option("header", true)
  .mode("overwrite")
  .parquet(destination);
```

Anexo 3: Presentation Tag ETL

```
import org.apache.spark.sql.functions._

val bucketName = "idt-stackoverflow"
val originLayer = "staging-layer"
val detinantionLayer = "presentation-layer"
val tableName = "tags.parquet"
val dimName = "dim_tag.parquet"
val origin = s"gs://$bucketName/$originLayer/$tableName"
val destination = s"gs://$bucketName/$detinantionLayer/$dimName"
//Reading staging tag table
val tags = spark.read.option("inferSchema", "true").parquet(origin)
//Renamed columns and adding tag key
val dim_tag = tags.withColumn("tag_key", expr("uuid()"))
                  .withColumnRenamed("count", "total_count")
                  .withColumnRenamed("id", "id_tag_nk")
                  .withColumnRenamed("tag_name", "name")
//Writing Tag Dimension
dim_tag.write
  .option("compression", "snappy")
  .option("header", true)
  .mode("overwrite")
  .parquet(destination);
```