

Dashboard para la visualización de información de los proyectos de la empresa Servicios Empresariales

Carlos Hernando Macías Flórez

Director

Elio H. Cables Pérez, Ph.D.

Universidad Antonio Nariño
Facultad de Ingeniería de Sistemas
Especialización en Gobierno de Datos
Bogotá, Colombia
2022

DEDICATORIA

El presente trabajo de grado esta dedicado a Dios, quien me inspiró en mi formación académica. A mi familia por estar ahí conmigo y creer en mí.

Al director Ph.D. Elio H. Cables Pérez, que me brindo su conocimiento y estuvo siempre a disposición.

Carlos Macías Florez

Agradecimientos

Expreso mi agradecimiento a todas las personas que colaboraron e hicieron posible el desarrollo del trabajo de grado. Principalmente, el agradecimiento a mi director de proyecto Ph.D. Elio H. Cables Pérez, quien me orientó en todo momento, su agradecimiento por su dedicación, motivación, esfuerzo, comprensión y principalmente por el apoyo brindado durante el proceso de desarrollo y culminación de este trabajo.

Y por último, agradecer a mi familia que con su amor, comprensión, apoyo y esfuerzo, me enseñaron a nunca desistir y siempre conseguir nuestras metas y objetivos.

Índice de Contenido

1 DESCRIPCIÓN DEL PROBLEMA	2
1.1 PLANTEAMIENTO DEL PROBLEMA	3
1.2 OBJETIVOS	3
1.2.1 Objetivo General	3
1.2.2 Objetivos Específicos	3
2 MARCO REFERENCIA	4
2.1 MARCO TEÓRICO	4
2.2 ESTADO DE ARTE	6
3 METODOLOGÍA	9
4 DESARROLLO	11
5 CONCLUSIONES	40
6 Referencias	41

Índice de figuras

3-1	Metodología	9
4-1	Inicialización del ambiente en software de Anaconda	11
4-2	Ejecución del software Anaconda	12
4-3	Instalación de librería para conexión de bases de datos	12
4-4	Entorno del ambiente en el notebook de Jupyter	13
4-5	Instalación de librerías en el notebook	13
4-6	Importación librería Mysql en el notebook de Jupyter	13
4-7	Importación librería SqlServer en el notebook de Jupyter	13
4-8	Carga y visualización de archivos set de datos Comercial	20
4-9	Conexión motor de bases de datos SqlServer área Producto	21
4-10	Carga y visualización de archivos set de datos Producto	21
4-11	Conexión motor de bases de datos MySQL área Tesorería	22
4-12	Carga y visualización de archivos set de datos Tesorería	22
4-13	Arquitectura elaborada	23
4-14	Tipos de datos set de datos Comercial	23
4-15	Información set de datos Comercial	24
4-16	Datos columnas set de datos Comercial	24
4-17	Tipos de datos set de datos Producto	25
4-18	Información set de datos Producto	26
4-19	Datos columnas set de datos Producto	26
4-20	Tipos de datos set de datos Tesorería	27
4-21	Información set de datos Tesorería	28
4-22	Datos columnas set de datos Tesorería	29
4-23	Estandarizar datos de los set de datos	30
4-24	Estandarizar datos de los set de datos	30
4-25	Estandarizar datos de los set de datos	31
4-26	Estandarizar datos de los set de datos	31
4-27	Estandarizar datos de los set de datos	31
4-28	Estandarizar datos de los set de datos	32
4-29	Estandarizar datos de los set de datos	32
4-30	Cargue tres set de datos áreas Comercial, Producto y Tesorería	33
4-31	Modelo Físico Dimensional	34

4-32	Repositorio final cargado en SqlServer	36
4-33	Aplicación web sección Clientes	37
4-34	Aplicación web sección Casos	38
4-35	Aplicación web terminal ejecutando	38
4-36	Repositorio documentos del notebook	39
4-37	Repositorio aplicación web utilizada para la visualización de resultados	39



Índice de tablas

4-1	Set de Datos área Comercial Archivo Excel	15
4-2	Set de Datos área Producto SqlServer	17
4-3	Set de Datos área Tesorería SqlServer	19
4-4	Descripción dimensiones	35

Resumen

En este trabajo se explica el desarrollo de un proyecto enfocado a la empresa de Servicios Empresariales, cuando se realiza las reuniones de comité de gerencia la información de los proyectos está desactualizada y con inconsistencias presentadas, y como consecuencia se tiene decisiones no acertadas en los procesos y seguimientos de los proyectos, como también pérdida de factores de tiempo, dinero y de recursos para la empresa de Servicios de Empresariales.

El propósito de este proyecto es brindar una solución implementando un dashboard para la visualización de la información de los proyectos tal que no manifieste inconsistencias a partir de los set de datos involucrados (Comercial, Producto y Tesorería).

Palabras Claves: decisiones, inconsistencias, factores, seguimientos.

Abstract

This paper explains the development of a project focused on the Servicios Empresariales company, when the management committee meetings are held, the information on the projects is outdated and with inconsistencies presented, and as a consequence there are wrong decisions in the processes. and follow-up of the projects, as well as loss of time, money and resource factors for the Servicios Empresariales company.

The purpose of this project is to provide a solution by implementing a dashboard for displaying project information such that it does not show inconsistencies from the data sets involved (Commercial, Product and Treasury).

Keywords: decisions, inconsistencies, factors, follow.

Introducción

Debido a datos inconsistentes o no reales en las fuentes de información, con lleva a fuga o datos erróneos en las empresas y como consecuencia resultados no esperados en el desarrollo y los procesos de la organización.

En el presente trabajo se implementó un dashboard para la visualización de la información de los proyectos a partir de los datos registrados por las áreas de Comercial, Producto y Tesorería tal que no manifiesten inconsistencias, en las reuniones de comité de gerencia cuando se evalúa el estado de los proyectos se encuentra en la mayoría de veces información inconsistente y a partir de esto se presentan factores como no tener la proyección real de cada proyecto, lo que ocasiona retrasos en las entregas de los desarrollos del proyecto y al no tener los datos real de cada proyecto esto conlleva a pérdida de clientes, perdida de dinero, recursos y tiempo para la empresa de Servicios Empresariales. Por medio de la identificación de estas inconsistencias a partir de los datos se obtiene mejores tomas de decisiones en la empresa de Servicios Empresariales.

Para la solución del proyecto se realizó una implementación de análisis de información aplicando métodos de limpieza, reglas de negocio, estandarización de la información a cada set de datos involucrados en el proyecto (Comercial, Producto y Tesorería) por medio de herramientas y lenguajes de análisis de datos.

Se concluye que el desarrollo del proyecto aportó beneficios como: acceso real a la información de cada proyecto para cualquier usuario de la empresa de Servicios Empresariales, identificación de factores de perdida de información entre las áreas de Comercial, Producto y Tesorería.

1 DESCRIPCIÓN DEL PROBLEMA

Hoy en día para las organizaciones los datos son lo más valioso y a su vez información para su día a día. Para el desarrollo de este proyecto se trabajará con una organización la cual es mi objeto de estudio, la organización se dedica al desarrollo de software, por el nivel de confidencialidad no se dará su nombre y se citará de la forma de la empresa de Servicios Empresariales.

La empresa de Servicios Empresariales es una organización que se dedica a la venta de productos y servicios en áreas de Tesorería y Accionistas en entidades financieras y del sector real, en el cual por medio de sus aplicativos se ve la integración de sus productos y servicios, asegurando también temas normativos y la solución de incidentes en la organización.

Para adquirir un servicio en la empresa de Servicios Empresariales el primer paso es pasar por el área de comercial, la cual se encarga de hacer una cotización para el cliente. El cliente revisa la propuesta y si está de acuerdo o si se necesita un ajuste lo comunica al área Comercial, después de aprobado el servicio esta información se lleva a un archivo de Excel que maneja el área comercial y lo guarda en un servidor de archivos dispuesto por la empresa, esta área es la encargada de notificar al área de Producto y Tesorería este servicio/producto a implementar.

El siguiente paso es el área de Producto la cual aloja la información de este servicio, ingresando etapas, procesos, tiempos y fechas del desarrollo, en un aplicativo interno en el cual su motor de bases de datos es SqlServer, en una instancia exclusiva para este aplicativo y un servidor asignado al área de Producto, de manera simultánea el área de Tesorería ingresa la información en un aplicativo externo que lleva la parte financiera alojada en un motor de bases de datos SqlServer en una instancia diferente y un servidor compartido junto con el área de recursos humanos, donde se ingresa acuerdos de pagos y fechas de pago del servicio.

Si el cliente desea agregar o remover algo en el servicio cotizado el encargado es el área Comercial para realizar el ajuste, y después enviar esa modificación a las áreas involucradas Producto y Tesorería.

Cada semana el comité de gerencia se reúne para evaluar los estados de los proyectos, en esta revisión algunas veces cuando las áreas comparten la información no concuerda con las demás, esto genera desinformación o desconocimiento real del proyecto y esto conlleva a falencias en la entrega del servicio al cliente, ya que si no se notifica este ajuste, por ejemplo el área de Producto no desarrolla ese ajuste que se pidió ó el área de Tesorería no modifica

los pagos del cliente y se estaría cobrando de más de menos en el valor del servicio por no realizar este ajuste. Estos factores conlleva a pérdida de clientes, de tiempo, de dinero y de credibilidad en la empresa al no tener la información actualizada correctamente.

1.1. PLANTEAMIENTO DEL PROBLEMA

¿Cómo visualizar la información de los proyectos de la empresa de Servicios Empresariales tal que no manifieste inconsistencias a partir de la información de las áreas de Comercial, Producto y Tesorería?.

1.2. OBJETIVOS

1.2.1. Objetivo General

Implementar un dashboard para la visualización de la información de los proyectos a partir de los datos registrados por las áreas de Comercial, Producto y Tesorería tal que no manifiesten inconsistencias.

1.2.2. Objetivos Específicos

- Caracterizar el conjunto de herramientas tecnológicas para el análisis de datos.
- Elaborar el proceso de extracción, transformación, carga y limpieza de los set de datos de las area de Comercial, Producto y Tesorería.
- Implementar un modelo de datos que soportará la información integrada de las fuentes de datos de Comercial, Tesorería y Producto.
- Visualizar la información obtenida después del análisis de los set de datos involucrados en el proyecto.

2 MARCO REFERENCIA

2.1. MARCO TEÓRICO

Los datos recolectados/almacenados en las organizaciones se dan por medio de sus plataformas, aplicativos, fuente de información manual, de las cuales resultan las tomas de decisiones.

Bases de Datos

Las bases de datos es una colección organizada de información o datos estructurados, es un lugar en donde los datos son almacenados y organizados (Database, 2021). Una base de datos es administrada por un sistema de gestión de bases de datos (DBMS). Existen bases de datos relacionales y no relacionales, hay una estructura compuesta por columnas y filas, formada por un conjunto de tablas que permiten aumentar la consulta de datos y la eficacia del procesamiento. De las cuales se pueden realizar acciones de gestionar, acceder, actualizar, controlar y organizar fácilmente los datos (Silberschatz, 2011). Cuando se habla de relacional esto significa que los datos que están almacenados estas organizados en forma de tablas y cada una se relaciona de alguna manera (Silberschatz, 2011), esta es una representación de datos el cual es un dataset que es un conjunto de datos los cuales están tabulados, se tiene una columna donde se describe la variable de la que se está hablando o analizando y posteriormente el resto de las columnas son datos ya organizados. Estos datos pueden por un solo individuo/producto o en caso contrario se pueden ver almacenados varios, esto influye la cantidad de filas que este tenga (Balagueró, 2018).

Para la administración de estos sistemas de información se encuentran herramientas como SqlServer, Mysql, entre otras.

SqlServer

SQL Server es un sistema que permite la administración y análisis de bases de datos, es un sistema relacional desarrollado por Microsoft (Andy Opel, 2010), el cual es utilizado para diferentes sectores de negocio como desde una tienda hasta un sistema de comercio electrónico, permitiendo gestionar varias líneas de negocio y lo mas importante el almacenamiento de datos. SQL Server se ejecuta en T-SQL (Transact -SQL), es un conjunto de extensiones de programación de Sybase y Microsoft (Andy Opel, 2010)

En el presente proyecto se utiliza ya que es muy fácil de trabajar, es el motor con el cual trabajan en la empresa de Servicios Empresariales y se toma como ejemplo de una fuente de los set de datos de las áreas involucradas.

Otra herramienta utilizada para este proyecto es Mysql.

MySql

MySQL es un sistema de administración de bases de datos relacionales y es de código abierto que maneja un modelo cliente-servidor. Mysql es RDBMS (Relational Database Management System) es un servicio que es utilizado para administrar bases de datos basadas en un modelo relacional permitiendo acciones como crear, modificar o eliminar datos (Bell, 2018).

Para el presente proyecto se utiliza ya que es muy fácil de trabajar y se toma como ejemplo de una fuente de los set de datos de las áreas que maneja la empresa de Servicios Empresariales.

Para realizar el desarrollo de una solución de un sistema de información se involucra el proceso de consolidación de datos el cuál se presenta o se realiza por medio de una bodega de datos.

Bodegas de datos

Las bodegas de datos es una solución donde se consolidan datos de diferentes orígenes y/o fuentes (Cortés Ramírez, 2019), es un tipo de base de datos que se especializa en integrar, completar, reunir y organizar información, con el fin de que pueda ser analizada y procesada fácilmente (R. Kimball., 2002), y con un objetivo de facilitar la toma de decisiones y centralizar datos que tienen una o más áreas de negocio, allí se pueden integrar con uno o más de dos data marts (almacén de datos que está orientado a un área en específica) (Golfarelli, 2009).

Con las bodegas de datos se realizan diseños tanto lógico y diseños físicos, los cuales se obtienen por modelos dimensionales, allí están aspectos principales como tablas de hechos y dimensiones (Cobos, 2006).

Esta solución de bodega de datos se realiza con procesos de carga de información desde un origen hasta un destino, a lo que después se procede con una transformación de datos para su análisis y posteriormente un cargue de datos a un destino escogido para su visualización. Además cada fuente de información de las áreas involucradas que son Comercial, Producto y Tesorería es extraída de donde esta alojada, pasa por un proceso de limpieza/calidad de los datos, es transformada para su análisis y exploración, es conocido como un proceso de ETL (Extract, Transform, Load) extraer, transformar, cargar. A esto se le aplica reglas de negocio con el objetivo de cumplir con las métricas de calidad y alojarla en un área de almacenamiento (Kakish, 2012).

Todo este proceso de análisis de datos va de la mano por medio de herramientas tecnológicas que ayudan al desarrollo de una solución.

Herramientas tecnológicas para el análisis de datos

Para el desarrollo de este proyecto se tiene en cuenta herramientas como

Python

Por medio de Python que es un lenguaje de alto nivel de programación interpretado (Python, 2021), Python es un lenguaje que es multi-paradigma como imperativo, funcional y orientado a objetos (Python 3.9, 2021). Es un lenguaje que es interpretado, que significa que no necesita de un compilador de código fuente para poder realizar su inicialización (Python 3.9, 2021), Python ofrece muchas ventajas algunas como por ejemplo curva de aprendizaje rápida, y una gran cantidad de características de funcionalidades en proyectos que son de baja escalabilidad (Python 3.9, 2021). Para el desarrollo del proyecto ejecutamos el código en el notebook para el análisis, exploración de los datos y con varias extensiones de librerías que hace que sea más compacto y se obtiene un código limpio comparado con otros lenguajes en el mercado, el uso de las librerías es variado como por ejemplo el de realizar diferentes extracción de fuentes de información ya sea que están alojadas en diferentes orígenes como un motor de bases de dato Sql-NoSql, un archivo o un repositorio. Para el proceso de integración, exploración y análisis las librerías como Pandas (Pandas, 2021) y Numpy (Numpy, 2021), esto ayuda en proporcionar tipos de n-array para los diferentes tipos de datos sobre Python, el término n-array se relaciona a un arreglo que tiene N dimensiones. Numpy maneja la persistencia, a un nivel eficiente tanto para entrada y salida de una matriz (Abraham A, 2014).

Jupyter

Es un notebook en el cual permite crear, ejecutar, compartir código y mostrar gráficas, aquí se realiza, se estructura y se ejecuta la construcción del código Python, que desde un navegador estándar permite ver el resultado del código realizado.

Angular

Se define como un framework de aplicaciones web y desarrollado en TypeScript, de código abierto, mantenido por Google, es utilizado para el mantenimiento y/o creación de aplicaciones web que manejen una sola página. (Angular,2022). Angular es organizado por medio de su patrón Modelo-Vista-Controlador y brinda ciertas características como: ser SPA (Single Page Applications), buen manejo de rutas, realización de llamadas al back-end, gestión de dependencias, realizar pruebas y entre otras (Angular,2022). Para este proyecto se utiliza ya que por medio de una página web sencilla se visualiza los resultados obtenidos.

2.2. ESTADO DE ARTE

Teniendo en cuenta que la información de cada organización al no estar centralizada conlleva a diferentes problemas en los procesos y en la toma de decisiones, todo esto por no tener planes de acción o medidas para el análisis de sus datos se encuentran trabajos de grado como

Modelo para el análisis de correlación de variables frente a la per-

cepción de seguridad ciudadana en Colombia

Es un proyecto de trabajo de grado en el cual recolectan información de diferentes fuentes de información de entidades como la Policía, Fiscalía, Datos Abiertos Colombia, Dane y Migración. A cada set de datos se le realiza un proceso de calidad de datos y de limpieza, ya que cada set tenía falencias en sus datos, información incompleta, errónea, y no concordaban la información entre estas entidades. Los set de datos fueron cargados a un motor de base de dato en donde realizaron un análisis y exploración, por medio de pl-sql y Python junto con librerías realizan todo el proceso de ETL. Después realizan un modelo de datos y posteriormente una visualización con el fin de analizar la problemática del trabajo de grado (Nieto, Fagua et al. 2021).

Por otra parte se encuentra el proyecto

Proyecto de Grado Bodegas de Datos Caso Adventure Works Cycles

Es un proyecto de trabajo de grado, donde identifican un problema en la empresa Adventure Work Cycles, lo que encuentran es que los procesos que actualmente manejan para el análisis de los datos no son óptimos para su desarrollo, debido a que este proceso lo realizan de forma manual y de una manera descentralizada en diferentes sistemas que la compañía maneja, como por ejemplo archivos de tipo Excel, textos planos, entre otros. El grupo de integrantes del proyecto identifican unos problemas, por ejemplo la información se tiene en diferentes versiones de tipo excel y esta información no esta en un lugar centralizado. Otro problema encontrado es que se duplica el esfuerzo dentro de la empresa porque hay varias personas que están trabajando sobre la misma información con las herramientas que tienen en la oficina, como consecuencia esto genera un gran tiempo en la preparación y generación de la entrega de los reportes y no hay mayor dedicación de tiempo suficiente para el análisis de los resultados.

Posteriormente la empresa en las reuniones o discusiones que programan presentan información que no concuerda con las demás fuentes, todo esto provocado por la veracidad de las cifras que se tienen, ya que cuentan con diferentes y varias versiones de esta información, lo que provoca y genera inconsistencias en los datos, menos calidad y precisión en la información que es entregada, después de que los autores identifican las fuentes de datos los procesos y el ciclo de vida que tiene la información, lo que realizan es un proceso de limpieza y estandarización de datos por medio de procesos de ETL, al tener ya la información lista para su respectivo análisis realizan un proceso de homologación de datos ya que hay varias fuentes que tienen la misma información y ya posteriormente por medio de una herramienta de BI (Business Intelligence) visualizan los resultados (Cortés Ramírez, 2019).

Prototipo de una bodega de datos para la empresa salón de automóvil

En la empresa Salón del Automóvil, una compañía que se dedica a la compra y venta de repuestos para automoviles, se encontraron problemas como la información que almacena es redundante, como no hay organización hace que esta información no sea utilizada de la mejor manera, además al generar la información como en reportes no hay una aplicación de donde la pueda sacar para realizar tomas de decisiones adecuadas para la organización y a esto se le suma que no hay un control de los datos. Es por eso que por medio del proceso de ETL y bodegas de datos se integra y se explota al máximo la información de dicha organización para un mejor desempeño y toma de decisiones (Pineda Suavita, 2011).

IMPACTO

El proyecto de grado busca implementar un dashboard de la información de la empresa de Servicios Empresariales de las fuentes de datos de las áreas involucradas Comercial, Producto y Tesorería con el fin de mitigar las inconsistencias que se generan en los set de datos y así en las reuniones de gerencia poder ver los estados de los proyectos y así mitigar factores como desconocimiento real de cada caso, además de pérdida de dinero y de tiempo de la organización todo esto ocasionado por las fuentes de información que manifiestan inconsistencias en sus datos y así poder contribuir en la toma de decisiones frente a los proyectos en la empresa de Servicios Empresariales.

COMPONENTE DE INNOVACIÓN

En la empresa de Servicios Empresariales al momento de las reuniones de los comités de gerencia, los estados de proyectos la mayoría de veces cuando están revisando el estado de cada uno o por agrupación de clientes la información presentada no concuerda ya que hay inconsistencias en los set de datos consultados, en este trabajo de proyecto de grado se plantea una solución a partir de procesos de análisis de datos con ayuda de nuevas herramientas tecnológicas que están involucradas en todo el ciclo de vida del dato, beneficiando a la empresa de Servicios de Empresariales con información real actualizada al día de los estados de los proyectos.

3 METODOLOGÍA

Una metodología es una cadena de pasos que busca desarrollar un proyecto o una investigación, aquí se describe cómo se va a desarrollar, los métodos y técnicas a utilizar ya sea para un proyecto científico, teórico ó experimental, con lo que se busca es alcanzar un resultado válido.

Para la realización de este proyecto se tuvo como referencia un estudio realizado llamado “Análisis de sentimiento y sus aplicaciones en la lucha contra el COVID-19 y las enfermedades infecciosas: una revisión sistemática” sistemática” (Alamoodi, 2021), en esta investigación se implementaron métodos y fases los cuales se utilizaron para el desarrollo del estudio, además con las definiciones realizadas por el DAMA-DMBOK se adapta para este proyecto de la empresa de Servicios Empresariales.

En la siguiente figura 3-1 se muestra las etapas que contiene la metodología a utilizar.

Figura 3-1: Metodología



Fuente: elaboración propia

A continuación se describe cada una de las etapas.

Identificación de las fuentes de datos

En esta etapa se identifica el origen de las fuentes de información de la empresa de Servicios Empresariales de las áreas Comercial, Producto y Tesorería, los datos de donde se va a obtener la información para su respectivo análisis. En cuanto al área Comercial es un archivo que esta en un formato Excel, por su parte el área de Producto es una base de dato con una instancia diferente asignada a esta área para el aplicativo interno en donde se almacena la información y por último el área de Tesorería, su fuente de datos es un aplicativo externo donde la información está en un motor de base de dato en un servidor diferente, solo compartido con el área de Recursos Humanos con una instancia exclusiva para este aplicativo.

Extracción de los datos

En esta siguiente etapa se realiza la extracción y cargue de las fuentes de información identificadas en la etapa anterior, haciendo uso de un notebook o un entorno de trabajo, con ayuda de las herramientas o librerías que se tiene para poder obtener la información para su respectivo análisis, además se hace una exploración de los set de datos, de como está estructurada y como está definida.

Pre-Procesamiento

Continuando con esta etapa se realiza un proceso de limpieza y estandarización de cada una de las fuentes de información de las áreas involucradas, también eliminando información que no es relevante para el análisis o datos erróneos.

Procesamiento

A continuación en esta etapa se realiza la integración de las fuentes de cada área para su respectivo análisis, y posteriormente se realiza un proceso de modelado de los datos y un diseño lógico del data warehouse.

Vizualización de los Resultados

Por último en esta etapa se mostrará el resultado de la información obtenida en un aplicación web sencilla elaborada por el área de desarrollo, la cuál es realizada por medio del framework de Angular, los datos obtenidos o mostrados en la página web se realiza consumiendo el repositorio construido alojado en el motor de bases de datos.

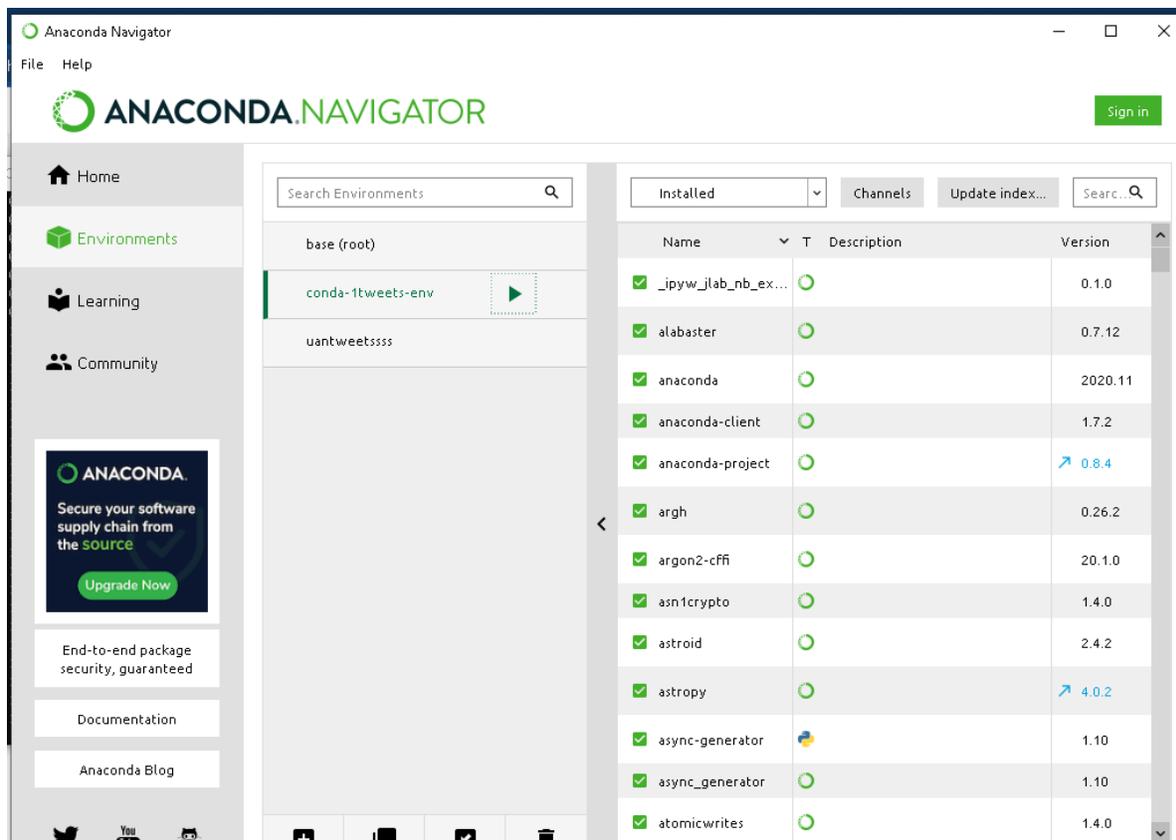
4 DESARROLLO

Para el desarrollo del proyecto se configura primero el ambiente, en el cual por medio de Anaconda se crea el entorno y por terminal se crea el entorno del notebook ver figuras 4-1 y 4-2 y Jupyter notebook el cual es utilizado por cada etapa de la metodología.

Posterior en la terminal de Anaconda-Jupyter para el ambiente se instala las librerías para conexión con motores de bases de datos, ver figura 4-3.

Se crea la estructura del proyecto y se crea el notebook como se muestra en la figura 4-4.

Figura 4-1: Inicialización del ambiente en software de Anaconda



Fuente: imagen tomada del software de Anaconda

Figura 4-2: Ejecución del software Anaconda

```

C:\WINDOWS\system32\cmd.exe - jupyter.exe notebook --notebook-dir="C:\Users\carpr\OneDrive\Documents\ProyectoEspecializacion2022-\SetDa...
[I 20:46:35.578 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 20:48:36.272 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 20:50:35.578 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 20:52:36.280 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 20:54:35.578 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 20:56:35.578 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 20:58:36.307 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:00:35.610 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:02:35.586 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:04:35.584 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:06:35.586 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:08:35.581 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:10:36.282 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:12:35.583 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:14:36.275 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:16:35.596 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:18:35.593 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:20:36.283 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:22:35.581 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:24:35.583 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:28:36.280 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:30:36.279 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:32:36.276 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:34:36.276 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:42:36.302 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:44:43.810 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:46:35.713 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:48:35.946 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
[I 21:58:36.306 NotebookApp] Saving file at /ProyectoGradoGD2022_I.ipynb
  
```

Fuente: imagen tomada de la consola de windows

Figura 4-3: Instalación de librería para conexión de bases de datos

```

C:\WINDOWS\system32\cmd.exe

(conda-1tweets-env) C:\Users\carpr>pip3 install pymysql
Collecting pymysql
  Downloading PyMySQL-1.0.2-py3-none-any.whl (43 kB)
  |-----| 43 kB 40 kB/s
Installing collected packages: pymysql
Successfully installed pymysql-1.0.2

(conda-1tweets-env) C:\Users\carpr>
  
```

Fuente: imagen tomada del terminal de windows

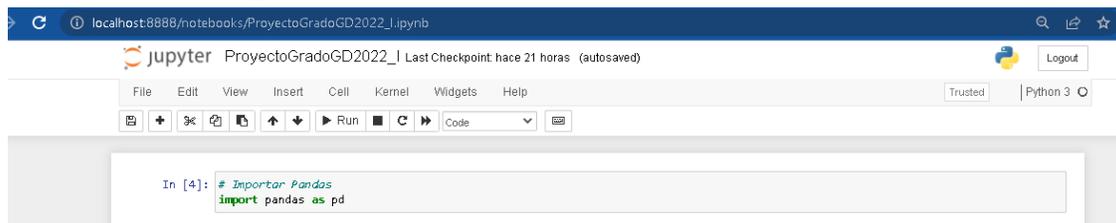
Figura 4-4: Entorno del ambiente en el notebook de Jupyter



Fuente: imagen tomada del ambiente de Jupyter

Después se realiza las importaciones de librerías necesarias para el proyecto, como lo son Pandas, como se muestra en la figura 4-5, que son utilizadas por cada etapa de la metodología y la importación de conexión de bases de datos, para MySql ver figura 4-6 y SqlServer ver figura 4-7.

Figura 4-5: Instalación de librerías en el notebook



Fuente: imagen tomada del ambiente de Jupyter

Figura 4-6: Importación librería Mysql en el notebook de Jupyter



Fuente: imagen tomada del ambiente de Jupyter

Figura 4-7: Importación librería SqlServer en el notebook de Jupyter



Fuente: imagen tomada del ambiente de Jupyter

Ya después de haber configurado el ambiente se procede con el desarrollo del proyecto de cada etapa de la metodología.

Identificación de las fuentes de datos

En la empresa de Servicios Empresariales los set de datos a trabajar se encuentran en las áreas de Comercial, Producto y Tesorería. En las reuniones gerenciales que realiza la empresa de Servicios Empresariales ellos revisan información de historial tanto trimestral y mensual, a esto también revisan casos por cada Cliente y casos individual dependiendo las prioridades que tengan, para el desarrollo del proyecto se va a escoger los set de datos con un historial de seis meses cada uno, que comprende entre las fechas de 1 de Septiembre de 2021 hasta el 28 de febrero de 2022.

El primer set de datos es administrado por el área Comercial el cuál esta elaborado en un archivo Excel, este archivo contiene toda la información de las cotizaciones que se le ha realizado a los clientes, productos y/o servicios (cotizados - adquiridos - ejecutados - finalizados - no ejecutados), los funcionarios de esta área gerente comercial, asesores comerciales y analista comercial son los que administran este archivo y son los encargados de ingresar, modificar información que se encuentra allí. En la tabla **4-1** se muestra una breve descripción de los campos del set de datos del área Comercial.

Tabla 4-1: Set de Datos área Comercial Archivo Excel

Set de Datos Comercial (Excel)			
Num	Nombre_Tecnico	Tipo_Dato	Descripción
1	caso	int	Consecutivo se asigna al día (YYMMDD+dia)
2	servicio_producto	varchar	Servicio o Producto adquirido por el cliente
3	descripcion_caso	varchar	Descripción del servicio o producto
4	nit_cliente	varchar	Identificación del cliente
5	nombre_cliente	varchar	Nombre del cliente
6	nro_etapa	int	Número de la etapa
7	nombre_etapa	varchar	Nombre de la etapa
8	descripcion_etapa	varchar	Descripción de la etapa
9	responsable_etapa	varchar	Responsable de la etapa
10	nro_proceso	int	Numero del proceso
11	proceso	varchar	Nombre del proceso
12	entregables_proceso	varchar	Entregables del proceso
13	caso_proceso	int	Caso proceso
14	descripcion_caso_proceso	varchar	Descripción del caso proceso
15	fecha_inicio_planeada	date	Fecha que se estima que inicia el proyecto
16	fecha_final_planeada	date	Fecha que se estima que finaliza el proyecto
17	dias_consultor_funcional	int	Días consultor funcional en el proyecto
18	dias_consultor_tecnico	int	Días consultor técnico en el proyecto
19	dias_gerente	int	Días gerente en el proyecto
20	total_dias_proceso	int	Total días proceso
21	dias_estimados	int	Días estimados en el proyecto
22	valor_dia_consultoria	int	Valor día consultoria
23	valor_consultoria	int	Valor total consultoria (Técnico + Funcional)
24	valor_dia_gerencia	int	Valor día gerencia
25	valor_gerencia	int	Valor total gerencia
26	horas_vendidas	int	Horas vendidas
27	acuerdo_pago	int	Tipo de acuerdo de pago
28	id_lider_funcional	int	Líder funcional
29	nombre_lider_funcional	varchar	Nombre líder funcional
30	id_lider_tecnico	int	Líder técnico
31	nombre_lider_tecnico	varchar	Nombre líder técnico
32	id_lider_cliente	int	Líder cliente
33	nombre_lider_cliente	varchar	Nombre líder cliente
34	valor_total_proyecto	int	Valor total del proyecto
35	estado_proyecto	varchar	Estado proyecto
36	observaciones_proyecto	varchar	Observaciones del proyecto

El siguiente set de datos a utilizar es manejado por el área de producto, esta información es enviada primero por el gerente Comercial al gerente de Producto y él es el encargado de ingresar la información a un aplicativo interno quienes lo administran el gerente de Producto y los desarrolladores para ingresar o modificar información, esta aplicativo interno maneja una base de datos SqlServer para alojar la información el cuál es exclusivo para el aplicativo y el área, para el desarrollo del proyecto se realiza una consulta al motor de bases de datos y se trae los campos a utilizar que maneja esta área los cuales se pueden ver en la tabla **4-2** en una breve descripción realizada.

Tabla 4-2: Set de Datos área Producto SqlServer

Set de Datos Producto (SqlServer)			
Num	Nombre_Tecnico	Tipo_Dato	Descripción
1	caso	int	Consecutivo se asigna al día (YYMMDD+dia)
2	servicio_producto	varchar	Servicio o Producto adquirido por el cliente
3	descripcion_caso	varchar	Descripción del servicio o producto
4	nit_cliente	varchar	Identificación del cliente
5	nombre_cliente	varchar	Nombre del cliente
6	nro_etapa	int	Número de la etapa
7	nombre_etapa	varchar	Nombre de la etapa
8	descripcion_etapa	varchar	Descripción de la etapa
9	responsable_etapa	varchar	Responsable de la etapa
10	id_proceso	int	Id del proceso
11	nro_proceso	int	Número del proceso
12	proceso	varchar	Nombre del proceso
13	entregables_proceso	varchar	Entregables del proceso
14	id_responsable_proceso	int	Id del responsable del proceso
15	responsable_proceso	varchar	Nombre responsable del proceso
16	estado_proceso	varchar	Estado del proceso
17	id_actividad	int	Id de la actividad
18	nombre_actividad	varchar	Nombre de la actividad
19	horas_actividad	int	Horas actividad
20	responsable_actividad	varchar	Responsable de la actividad
21	caso_proceso	int	Caso proceso
22	descripcion_caso_proceso	varchar	Descripción del caso proceso
23	fecha_inicio_planeada	date	Fecha que se estima que inicia el proyecto
24	fecha_final_planeada	date	Fecha que se estima que finaliza el proyecto
25	fecha_inicio	date	Fecha que inicia el proyecto en Producto
26	fecha_final	date	Fecha que finaliza el proyecto en Producto
27	fecha_final_actual	date	Fecha final actual que esta el proyecto
28	dias_desfase	int	Días de desfase que tiene el proyecto
29	horas_vendidas	int	Horas vendidas del proyecto
30	horas_estimadas	int	Horas estimadas del proyecto
31	horas_ejecutadas	int	Horas ejecutadas en el proyecto
32	id_lider_funcional	int	Líder funcional
33	nombre_lider_funcional	varchar	Nombre líder funcional
34	id_lider_tecnico	int	Líder técnico
35	nombre_lider_tecnico	varchar	Nombre líder técnico

Y por último el set de datos de Tesorería, este lleva toda la información contable de los servicios y/o productos adquiridos por los clientes es manejado por los funcionarios gerente Financiero y los analistas contables el cuál es llevado por un aplicativo externo y guarda la información en un motor de bases de datos SqlServer, de allí es generado la consulta de los campos a utilizar para el proyecto como se muestran en la tabla **4-3** en una descripción realizada.

Tabla 4-3: Set de Datos área Tesorería SqlServer

Set de Datos Tesoreria (SqlServer)			
Num	Nombre_Tecnico	Tipo_Dato	Descripción
1	caso	int	Consecutivo se asigna al día (YYMMDD+dia)
2	servicio_producto	varchar	Servicio o Producto adquirido por el cliente
3	descripcion_caso	varchar	Descripción del servicio o producto
4	nit_cliente	varchar	Identificación del cliente
5	nombre_cliente	varchar	Nombre del cliente
6	nro_etapa	int	Número de la etapa
7	nombre_etapa	varchar	Nombre de la etapa
8	descripcion_etapa	varchar	Descripción de la etapa
10	id_proceso	int	Id del proceso
11	nro_proceso	int	Número del proceso
12	proceso	varchar	Nombre del proceso
13	responsable_proceso	varchar	Nombre responsable del proceso
14	estado_proceso	varchar	Estado del proceso
15	caso_proceso	int	Caso proceso
16	descripcion_caso_proceso	varchar	Descripción del caso proceso
17	fecha_inicio_planeada	date	Fecha que se estima que inicia el proyecto
18	fecha_final_planeada	date	Fecha que se estima que finaliza el proyecto
19	fecha_inicio	date	Fecha que inicia el proyecto en Producto
20	fecha_final	date	Fecha que finaliza el proyecto en Producto
21	fecha_final_actual	date	Fecha final actual que esta el proyecto
22	horas_vendidas	int	Horas vendidas del proyecto
23	horas_estimadas	int	Horas estimadas del proyecto
24	horas_ejecutadas	int	Horas ejecutadas en el proyecto
25	costo_planeado	int	Costo planeado en el proyecto
26	costo_ejecutado	int	Costo ejecutado en el proyecto
27	valor_consultoria	int	Valor consultoria del proyecto
28	valor_gerencia	int	Valor gerencia en el proyecto
29	id_acuerdo_pago	int	Tipo de acuerdo de pago
30	nombre_acuerdo_pago	varchar	Nombre de acuerdo de pago
31	observaciones_proyecto	varchar	Observaciones del proyecto

Fuente: elaboración propia

Extracción de los datos

Después en esta etapa se realiza el cargue de las fuentes de información de Servicios Empresariales de las áreas de Comercial Producto y Tesorería al notebook y mostrando algunos registros, ver figuras 4-8, 4-10 y 4-12.

En las figuras 4-9 y 4-11 se realiza la conexión tanto el motor de bases de datos SqlServer set de datos área de Producto y Mysql set de datos área de Tesorería.

Figura 4-8: Carga y visualización de archivos set de datos Comercial

```

In [4]: # Importar Pandas
import pandas as pd

In [9]: # Leemos la data del Excel y la guardamos
df_setComercial12 = pd.read_excel(r'C:\Users\carpr\OneDrive\Documentos\ProyectoEspecializacion2022-1\SetDatos\Ambiente\Archi...

In [17]: # Muestra de los datos
df_setComercial12.head()
    
```

	caso	servicio_producto	descripcion_caso	nif_cliente	cliente	nro_etapa	nombre_etapa	descripcion_etapa	responsable_etapa	nro_proceso	...
0	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	1	Definición Alcance	Definición Alcance	Ivan G. Avellaneda	1	...
1	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	1	Definición Alcance	Definición Alcance	Ivan G. Avellaneda	2	...
2	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	1	Definición Alcance	Definición Alcance	Ivan G. Avellaneda	3	...
3	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	2	Desarrollo	Desarrollo	Oscar Rivas	1	...
4	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	2	Desarrollo	Desarrollo	Oscar Rivas	2	...

5 rows x 36 columns

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-9: Conexión motor de bases de datos SqlServer área Producto

```
In [16]: server = 'LAPTOP-5LQND9AB\SUCHAC739'
database = 'ProductoSet'
username = 'sa'
password = 'sa'
cnxn = pyodbc.connect("DRIVER={SQL Server};SERVER="+server+";DATABASE="+database+";UID="+username+";PWD="+password)
cursor = cnxn.cursor()
# select 26 rows from SQL table to insert in dataframe.
query = "SELECT * FROM [ProductoSet].[dbo].[Producto$];"
df_setProducto2 = pd.read_sql(query, cnxn)
# print(df_setProducto2.head(5))
df_setProducto2.head()
```

Out[16]:

	caso	servicio_producto	descripcion_caso	nit_cliente	cliente	nro_etapa	nombre_etapa	descripcion_etapa	entregable_etapa	responsable_et
0	21090101.0	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827.0	Aeronautica Civil	1.0	Definición Alcance	Definición Alcance	PS171130	Ivan G. Avellan
1	21090101.0	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827.0	Aeronautica Civil	1.0	Definición Alcance	Definición Alcance	PP171130	Ivan G. Avellan
2	21090101.0	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827.0	Aeronautica Civil	1.0	Definición Alcance	Definición Alcance	PA171130	Ivan G. Avellan
3	21090101.0	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827.0	Aeronautica Civil	2.0	Desarrollo	Desarrollo	PA170228	Oscar Ri
4	21090101.0	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827.0	Aeronautica Civil	2.0	Desarrollo	Desarrollo	PS170228	Oscar Ri

5 rows x 36 columns

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-10: Carga y visualización de archivos set de datos Producto

```
In [23]: # Muestra de Los datos del set de datos de Producto
df_setProducto2.head()
```

Out[23]:

	caso	servicio_producto	descripcion_caso	nit_cliente	cliente	nro_etapa	nombre_etapa	descripcion_etapa	entregable_etapa	responsable_etapa	...	fe
0	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	1	Definición Alcance	Definición Alcance	PS171130	Ivan G. Avellaneda	...	
1	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	1	Definición Alcance	Definición Alcance	PP171130	Ivan G. Avellaneda	...	
2	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	1	Definición Alcance	Definición Alcance	PA171130	Ivan G. Avellaneda	...	
3	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	2	Desarrollo	Desarrollo	PA170228	Oscar Rivas	...	
4	21090101	Atención de Incidentes	Atención de Incidentes Aeronautica Civil	860035827	Aeronautica Civil	2	Desarrollo	Desarrollo	PS170228	Oscar Rivas	...	

5 rows x 36 columns

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-11: Conexión motor de bases de datos MySql área Tesorería

```
In [13]: conn=pymysql.connect(host='localhost',port=int(3306),user='root',passwd='[REDACTED]',db='tesoreria')
df_setTesoreria2=pd.read_sql_query("select * from archivosettesoreria",conn)
print(df_setTesoreria2)
```

caso	servicio_producto	descripcion_caso	nit_cliente
0	21090101	Atención de Incidentes	860035827
1	21090101	Atención de Incidentes	860035827
2	21090101	Atención de Incidentes	860035827
3	21090101	Atención de Incidentes	860035827
4	21090101	Atención de Incidentes	860035827
...
7384	21121701	Problemas	860035843
7385	21121701	Problemas	860035843
7386	21121701	Problemas	860035843
7387	21121701	Problemas	860035843
7388	21121701	Problemas	860035843

cliente	nro_etapa	nombre_etapa	descripcion_etapa
Aeronautica Civil	1	Definición Alcance	Definición Alcance

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-12: Carga y visualización de archivos set de datos Tesorería

```
In [31]: # Muestra de los datos del set de datos de Tesorería
df_setTesoreria2.tail()
```

```
Out[31]:
```

caso	servicio_producto	descripcion_caso	nit_cliente	cliente	nro_etapa	nombre_etapa	descripcion_etapa	id_proceso	nro_proceso	...	horas_v
98284	21090103	Atención de Incidentes Servicios	860035829	Alianza Valores	2	Desarrollo	Desarrollo	1683042	2	...	
98285	21090103	Atención de Incidentes Servicios	860035829	Alianza Valores	2	Desarrollo	Desarrollo	1683043	3	...	
98286	21090103	Atención de Incidentes Servicios	860035829	Alianza Valores	2	Desarrollo	Desarrollo	1683044	4	...	
98287	21090103	Atención de Incidentes Servicios	860035829	Alianza Valores	2	Desarrollo	Desarrollo	1683045	5	...	
98288	21090103	Atención de Incidentes Servicios	860035829	Alianza Valores	3	Pruebas	Pruebas	1683046	1	...	

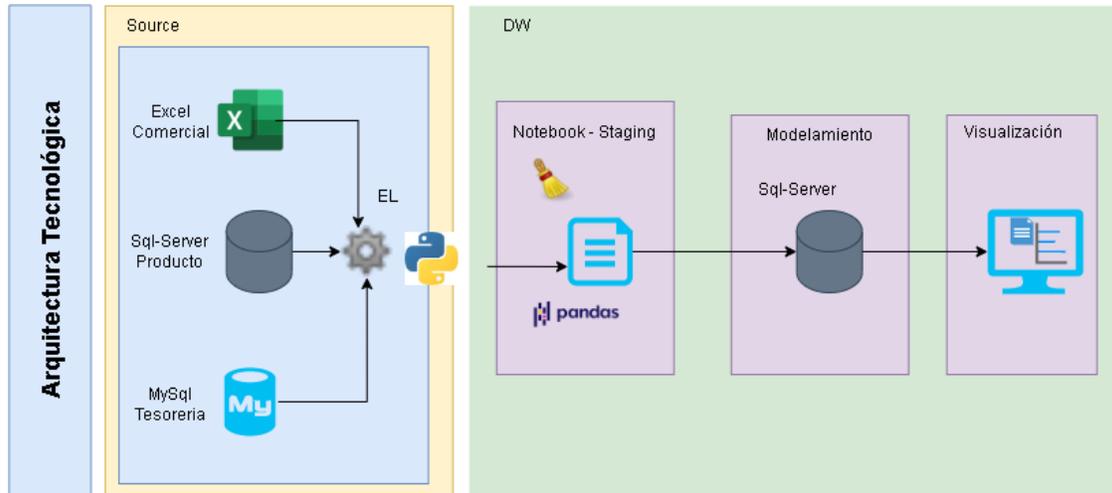
5 rows x 30 columns

Fuente: imagen tomada del ambiente de Jupyter

En la figuras 4-8, 4-10 y 4-12, se observa los registros cargados y una pequeña visualización de los datos, nombre de las columnas y la información cargada.

Para el desarrollo del proyecto la arquitectura realizada se muestra en la figura 4-13

Figura 4-13: Arquitectura elaborada



Fuente: elaboración propia

Pre-Procesamiento

En esta etapa se realiza una breve exploración de cada set de datos tanto del área Comercial, Producto y Tesorería, mostrando descripción, tipos de datos cargados en cada dataframe, información de cada set y la visualización de las columnas con el fin de identificarlas, ver figuras 4-14, 4-15, 4-16, 4-17, 4-18, 4-19, 4-20, 4-21 y 4-22

Figura 4-14: Tipos de datos set de datos Comercial

```
In [18]: # Tipos de datos de las columnas del set de datos de Comercial
df_setComercial2.dtypes

Out[18]: caso                int64
servicio_producto          object
descripcion_caso           object
nit_cliente                int64
cliente                    object
nro_etapa                  int64
nombre_etapa               object
descripcion_etapa          object
responsable_etapa         object
nro_proceso                int64
proceso                    object
entregables_proceso       object
caso_proceso               float64
descripcion_caso_proceso   object
fecha_inicio_planeada     object
fecha_fin_planeada        object
dias_consultor_funcional  float64
dias_consultor_tecnico    float64
dias_gerente               float64
tota_dias_proceso         float64
dias_estimados             float64
valor_dia_consultoria     int64
valor_consultoria         float64
valor_dia_gerencia        int64
valor_gerencia            float64
horas_vendidas            float64
acuerdo_pago              object
id_lider_funcional        float64
nombre_lider_funcional    object
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-15: Información set de datos Comercial

```
In [19]: # Información general del set de datos de Comercial
df_setComercial2.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 112500 entries, 0 to 112499
Data columns (total 36 columns):
#   Column                                Non-Null Count  Dtype
---  ---
0   caso                                   112500 non-null  int64
1   servicio_producto                     112500 non-null  object
2   descripcion_caso                       112500 non-null  object
3   nit_cliente                            112500 non-null  int64
4   cliente                                112500 non-null  object
5   nro_etapa                              112500 non-null  int64
6   nombre_etapa                           112500 non-null  object
7   descripcion_etapa                      112500 non-null  object
8   responsable_etapa                     112500 non-null  object
9   nro_proceso                            112500 non-null  int64
10  proceso                                 112500 non-null  object
11  entregables_proceso                   112500 non-null  object
12  caso_proceso                          13500 non-null   float64
13  descripcion_caso_proceso               13500 non-null   object
14  fecha_inicio_planeada                 112500 non-null   object
15  fecha_fin_planeada                    111300 non-null   object
16  dias_consultor_funcional              110100 non-null   float64
17  dias_consultor_tecnico                 111600 non-null   float64
18  dias_gerente                           111900 non-null   float64
19  tota_dias_proceso                     109500 non-null   float64
20  dias_estimados                        112500 non-null   float64
21  valor_dia_consultoria                  112500 non-null   int64
22  valor_consultoria                      112500 non-null   float64
23  valor_dia_gerencia                     112500 non-null   int64
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-16: Datos columnas set de datos Comercial

```
In [20]: # Columnas del set de datos de Comercial
df_setComercial2.columns

Out[20]: Index(['caso', 'servicio_producto', 'descripcion_caso', 'nit_cliente',
'cliente', 'nro_etapa', 'nombre_etapa', 'descripcion_etapa',
'responsable_etapa', 'nro_proceso', 'proceso', 'entregables_proceso',
'caso_proceso', 'descripcion_caso_proceso', 'fecha_inicio_planeada',
'fecha_fin_planeada', 'dias_consultor_funcional',
'dias_consultor_tecnico', 'dias_gerente', 'tota_dias_proceso',
'dias_estimados', 'valor_dia_consultoria', 'valor_consultoria',
'valor_dia_gerencia', 'valor_gerencia', 'horas_vendidas',
'acuerdo_pago', 'id_lider_funcional', 'nombre_lider_funcional',
'id_lider_tecnico', 'nombre_lider_tecnico', 'id_lider_cliente',
'nombre_lider_cliente', 'valor_total_proyecto', 'estado_proyecto',
'observaciones_proyecto'],
dtype='object')
```

```
In [22]: # Leeemos datos de Las otras areas involucradas Producto y Tesoreria
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-17: Tipos de datos set de datos Producto

```
In [24]: # Tipos de datos de Las columnas del set de datos de Producto
df_setProducto2.dtypes
```

```
Out[24]: caso                int64
servicio_producto          object
descripcion_caso           object
nit_cliente                int64
cliente                    object
nro_etapa                  int64
nombre_etapa               object
descripcion_etapa          object
entregable_etapa           object
responsable_etapa         object
id_proceso                 int64
nro_proceso                int64
proceso                    object
entregables_proceso        object
id_responsable_proceso     int64
responsable_proceso        object
estado_proceso             object
id_actividad               int64
nombre_actividad           object
horas_actividad            float64
responsable_actividad      object
caso_proceso               float64
descripcion_caso_proceso   float64
fecha_inicio_planeada      object
fecha_fin_planeada         object
fecha_inicio               object
fecha_final                object
fecha_final_actual         object
dias_desfase               int64
horas_vendidas             float64
horas_estimadas            float64
horas_ejecutadas           float64
id_lider_funcional         float64
nombre_lider_funcional     object
id_lider_tecnico           float64
nombre_lider_tecnico       object
dtype: object
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-18: Información set de datos Producto

```
In [25]: # Información general del set de datos de Producto
df_setProducto2.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 107519 entries, 0 to 107518
Data columns (total 36 columns):
#   Column                Non-Null Count  Dtype
---  ---
0   caso                  107519 non-null  int64
1   servicio_producto    107519 non-null  object
2   descripcion_caso     107519 non-null  object
3   nit_cliente          107519 non-null  int64
4   cliente              107519 non-null  object
5   nro_etapa            107519 non-null  int64
6   nombre_etapa         107519 non-null  object
7   descripcion_etapa    107519 non-null  object
8   entregable_etapa     107519 non-null  object
9   responsable_etapa    107519 non-null  object
10  id_proceso            107519 non-null  int64
11  nro_proceso           107519 non-null  int64
12  proceso              107519 non-null  object
13  entregables_proceso  107519 non-null  object
14  id_responsable_proceso 107519 non-null  int64
15  responsable_proceso   107519 non-null  object
16  estado_proceso       107519 non-null  object
17  id_actividad          107519 non-null  int64
18  nombre_actividad     107519 non-null  object
19  horas_actividad      107519 non-null  float64
20  responsable_actividad 107505 non-null  object
21  caso_proceso         0 non-null      float64
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-19: Datos columnas set de datos Producto

```
In [26]: # Columnas del set de datos de Producto
df_setProducto2.columns

Out[26]: Index(['caso', 'servicio_producto', 'descripcion_caso', 'nit_cliente',
'cliente', 'nro_etapa', 'nombre_etapa', 'descripcion_etapa',
'entregable_etapa', 'responsable_etapa', 'id_proceso', 'nro_proceso',
'proceso', 'entregables_proceso', 'id_responsable_proceso',
'responsable_proceso', 'estado_proceso', 'id_actividad',
'nombre_actividad', 'horas_actividad', 'responsable_actividad',
'caso_proceso', 'descripcion_caso_proceso', 'fecha_inicio_planeada',
'fecha_fin_planeada', 'fecha_inicio', 'fecha_final',
'fecha_final_actual', 'dias_desfase', 'horas_vendidas',
'horas_estimadas', 'horas_ejecutadas', 'id_lider_funcional',
'nombre_lider_funcional', 'id_lider_tecnico', 'nombre_lider_tecnico'],
dtype='object')
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-20: Tipos de datos set de datos Tesorería

```
In [32]: # Tipos de datos de Las columnas del set de datos de tesorería
df_setTesoreria2.dtypes
```

```
Out[32]: caso                int64
servicio_producto           object
descripcion_caso            object
nit_cliente                 int64
cliente                     object
nro_etapa                   int64
nombre_etapa                object
descripcion_etapa           object
id_proceso                  int64
nro_proceso                 int64
proceso                     object
responsable_proceso         object
estado_proceso              object
caso_proceso                float64
descripcion_caso_proceso    float64
fecha_inicio_planeada       object
fecha_fin_planeada          object
fecha_inicio                object
fecha_final                 object
fecha_final_actual          object
horas_vendidas              float64
horas_estimadas             float64
horas_ejecutadas            float64
costo_planeado              float64
costo_ejecutado             float64
valor_consultoria           float64
valor_gerencia              float64
id_acuerdo_pago             int64
nombre_acuerdo_pago         object
observaciones_proyecto      object
dtype: object
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-21: Información set de datos Tesorería

```
In [33]: # Información general del set de datos de Tesorería
df_setTesoreria2.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 98289 entries, 0 to 98288
Data columns (total 30 columns):
#   Column                                Non-Null Count  Dtype
---  ---                                -
0   caso                                  98289 non-null  int64
1   servicio_producto                    98289 non-null  object
2   descripcion_caso                     98289 non-null  object
3   nit_cliente                           98289 non-null  int64
4   cliente                               98289 non-null  object
5   nro_etapa                             98289 non-null  int64
6   nombre_etapa                         98289 non-null  object
7   descripcion_etapa                    98289 non-null  object
8   id_proceso                           98289 non-null  int64
9   nro_proceso                           98289 non-null  int64
10  proceso                               98289 non-null  object
11  responsable_proceso                  98289 non-null  object
12  estado_proceso                       98289 non-null  object
13  caso_proceso                         0 non-null      float64
14  descripcion_caso_proceso             0 non-null      float64
15  fecha_inicio_planeada                98289 non-null  object
16  fecha_fin_planeada                   97241 non-null  object
17  fecha_inicio                         98289 non-null  object
18  fecha_final                           98287 non-null  object
19  fecha_final_actual                   98289 non-null  object
20  horas_vendidas                       98289 non-null  float64
21  horas_estimadas                      98289 non-null  float64
22  horas_ejecutadas                     98289 non-null  float64
23  costo_planeado                       98289 non-null  float64
24  costo_ejecutado                      98289 non-null  float64
25  valor_consultoria                    98289 non-null  float64
26  valor_gerencia                       98289 non-null  float64
27  id_acuerdo_pago                      98289 non-null  int64
28  nombre_acuerdo_pago                  98289 non-null  object
29  observaciones_proyecto               94097 non-null  object
dtypes: float64(9), int64(6), object(15)
memory usage: 22.5+ MB
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-22: Datos columnas set de datos Tesorería

```
In [34]: # Columnas del set de datos de Tesorería
df_setTesoreria2.columns

Out[34]: Index(['caso', 'servicio_producto', 'descripcion_caso', 'nit_cliente',
               'cliente', 'nro_etapa', 'nombre_etapa', 'descripcion_etapa',
               'id_proceso', 'nro_proceso', 'proceso', 'responsable_proceso',
               'estado_proceso', 'caso_proceso', 'descripcion_caso_proceso',
               'fecha_inicio_planeada', 'fecha_fin_planeada', 'fecha_inicio',
               'fecha_final', 'fecha_final_actual', 'horas_vendidas',
               'horas_estimadas', 'horas_ejecutadas', 'costo_planeado',
               'costo_ejecutado', 'valor_consultoria', 'valor_gerencia',
               'id_acuerdo_pago', 'nombre_acuerdo_pago', 'observaciones_proyecto'],
              dtype='object')
```

Fuente: imagen tomada del ambiente de Jupyter

Si es necesario se cambia el tipo de formato de algunas de las columnas para poder trabajarla.

En cada set de datos tanto del área Comercial, Producto y Tesorería se encuentra datos que no concuerdan, datos nulos, datos erróneos y mal ingresados.

Con cada área se realizó una reunión con el encargado para revisar cada columna del set de datos, esto con el fin de realizar reglas y normas para estandarizar la información datos erróneos, datos vacíos, valores por defecto, por ejemplo si en un campo nombre_campo su dato es vacío o nulo se aplicaba la regla definida se actualizaba con un dato estándar o se realizaba la concatenación de algunos otros campos para no dejar ese campo vacío.

Uno de esos ejemplos en un escenario de datos erróneos fue aplicado a la columna cliente; habían formas en donde un cliente estaba de diferentes maneras ingresado (ver figuras 4-23 y 4-24), un cliente estaba de diferentes maneras escrito lo que se hizo fue dejar un nombre estándar aplicado en las reglas previamente elaboradas por cada área. Esto se hizo para cada set de datos con las reglas establecidas.

Figura 4-23: Estandarizar datos de los set de datos

```
In [40]: # Procedemos a limpiar y a Estandarizar información de cada uno de los sets
df_setComercial2['cliente'].unique()

Out[40]: array(['Aeronautica Civil', 'Alianza Fiduciaria', 'Alianza Valores',
               'Allianz', 'Almacafé', 'AV Villas', 'Banco AV Villas',
               'BancoAV Villas', 'BancoAV-Villas', 'AXA Colpatría',
               'AXA Banco Colpatría', 'Banco Colpatría', 'Banco Colpatría AXA ',
               'Banco AXA Colpatría', 'Banco Scotiabank',
               'Banco Agrario de Colombia', 'Banco W', 'Grupo Bancolombia',
               'Banco Colombia', 'BancoColombia', 'Grupo Banco Colombia',
               'Grupo Bancolombia Empresas', 'Bancolombia', 'BBVA Colombia',
               'Caxdac', 'Colfondos', 'Colpensiones', 'Colsubsidio', 'Compensar',
               'Corficolombiana', 'Davivienda', 'Ecopetrol S.A.',
               'Edegel S.A.A. (Perú)', 'EE.PP.M.', 'FAMSA',
               'Fiduciaria Colpatría', 'Fiducoldex'], dtype=object)

In [41]: # Estandarizamos datos
df_setComercial2['cliente'].replace({'Grupo Bancolombia': 'Bancolombia', 'Banco Colombia': 'Bancolombia', 'BancoColombia': 'Bancolombia'})

In [42]: #Revisamos los datos despues de realizar la actualización
df_setComercial2['cliente'].unique()

Out[42]: array(['Aeronautica Civil', 'Alianza Fiduciaria', 'Alianza Valores',
               'Allianz', 'Almacafé', 'AV Villas', 'Banco AV Villas',
               'BancoAV Villas', 'BancoAV-Villas', 'AXA Colpatría',
               'AXA Banco Colpatría', 'Banco Colpatría', 'Banco Colpatría AXA ',
               'Banco AXA Colpatría', 'Banco Scotiabank',
               'Banco Agrario de Colombia', 'Banco W', 'Bancolombia',
               'BBVA Colombia', 'Caxdac', 'Colfondos', 'Colpensiones',
               'Colsubsidio', 'Compensar', 'Corficolombiana', 'Davivienda',
               'Ecopetrol S.A.', 'Edegel S.A.A. (Perú)', 'EE.PP.M.', 'FAMSA',
               'Fiduciaria Colpatría', 'Fiducoldex'], dtype=object)
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-24: Estandarizar datos de los set de datos

```
In [43]: # Procedemos a limpiar y a Estandarizar información de cada uno de los sets
df_setTesoreria2['cliente'].unique()

Out[43]: array(['Aeronautica Civil', 'Alianza Fiduciaria', 'Alianza Valores',
               'Allianz', 'Almacafé', 'AV Villas', 'Banco AV Villas',
               'BancoAV Villas', 'BancoAV-Villas', 'AXA Colpatría',
               'AXA Banco Colpatría', 'Banco Colpatría', 'Banco Colpatría AXA ',
               'Banco AXA Colpatría', 'Banco Scotiabank',
               'Banco Agrario de Colombia', 'Banco W', 'Grupo Bancolombia',
               'Banco Colombia', 'BancoColombia', 'Grupo Banco Colombia',
               'Grupo Bancolombia Empresas', 'Bancolombia', 'BBVA Colombia',
               'Caxdac', 'Colfondos', 'Colpensiones', 'Colsubsidio', 'Compensar',
               'Corficolombiana', 'Davivienda', 'Ecopetrol S.A.',
               'Edegel S.A.A. (Perú)', 'EE.PP.M.', 'FAMSA',
               'Fiduciaria Colpatría', 'Fiducoldex'], dtype=object)

In [44]: df_setTesoreria2['cliente'].replace({'Banco AV Villas': 'AV Villas', 'BancoAV Villas': 'AV Villas', 'BancoAV-Villas': 'AV Villas'})

In [45]: #Revisamos los datos despues de realizar la actualización
df_setTesoreria2['cliente'].unique()

Out[45]: array(['Aeronautica Civil', 'Alianza Fiduciaria', 'Alianza Valores',
               'Allianz', 'Almacafé', 'AV Villas', 'AXA Colpatría',
               'AXA Banco Colpatría', 'Banco Colpatría', 'Banco Colpatría AXA ',
               'Banco AXA Colpatría', 'Banco Scotiabank',
               'Banco Agrario de Colombia', 'Banco W', 'Grupo Bancolombia',
               'Banco Colombia', 'BancoColombia', 'Grupo Banco Colombia',
               'Grupo Bancolombia Empresas', 'Bancolombia', 'BBVA Colombia',
               'Caxdac', 'Colfondos', 'Colpensiones', 'Colsubsidio', 'Compensar',
               'Corficolombiana', 'Davivienda', 'Ecopetrol S.A.',
               'Edegel S.A.A. (Perú)', 'EE.PP.M.', 'FAMSA',
               'Fiduciaria Colpatría', 'Fiducoldex'], dtype=object)
```

Fuente: imagen tomada del ambiente de Jupyter

Otro escenario era por ejemplo valores por defecto, para la columna "nombre_lider_funcional" del set de datos Producto habian datos vacios, NaN, y valores numericos ver figura 4-25, lo que

se hizo fue revisar las reglas establecidas por el área y aquí cada cliente tiene un asignado un líder funcional lo que se hizo fue buscar en el set de datos de Producto las filas que no tuvieran asignado un líder funcional ver figura 4-26 y se reemplazaba ese campo por el nombre del líder funcional asignado, después de haber realizado esta modificación ya los datos se encontraban actualizados y acordes a las reglas establecidas ver figura 4-27

Figura 4-25: Estandarizar datos de los set de datos

```
In [100]: # Buscar datos vacios ó NaN en los set de datos
df_setProducto2['nombre_lider_funcional'].unique()

Out[100]: array(['Yudy Tatiana Prieto', nan, 'Alexandra Cano', 'Arnold Peña B',
'Ivan G. Avellaneda', 'Oscar Rivas', 'Katherine Santamaria',
'Andrea Scarpetta', 'Wildiman Rodriguez', 0], dtype=object)
```

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-26: Estandarizar datos de los set de datos

```
In [114]: # Buscamos los valores que estan vacios o tiene un valor NaN para poder ser reemplazados, se muestra caso y cliente
# para saber que tipo de id de lider funcional le pertenece
df_setProducto2[~df_setProducto2.nombre_lider_funcional.isin(['Yudy Tatiana Prieto', 'Alexandra Cano', 'Arnold Peña B',
'Ivan G. Avellaneda', 'Oscar Rivas', 'Katherine Santamaria',
'Andrea Scarpetta', 'Wildiman Rodriguez'])][['caso', 'nit_cliente', 'id_lider_funcional', 'nombre_lider_funcional']]
```

```
Out[114]:
```

	caso	nit_cliente	id_lider_funcional	nombre_lider_funcional
1	21090101	860035827	NaN	NaN
14	21090101	860035827	NaN	NaN
34	21090103	860035829	NaN	NaN
50	21090106	860035830	NaN	NaN
68	21090107	860035831	NaN	NaN
119	21090109	860035833	NaN	NaN
129	21090110	860035834	NaN	NaN
162	21090202	860035836	NaN	NaN
192	21090204	860035838	NaN	NaN
205	21090205	860035839	NaN	NaN
247	21090208	860035842	NaN	NaN

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-27: Estandarizar datos de los set de datos

```
In [130]: # Buscar datos vacios ó NaN en los set de datos
df_setProducto2['nombre_lider_funcional'].unique()

Out[130]: array(['Yudy Tatiana Prieto', 'Alexandra Cano', 'Arnold Peña B',
'Ivan G. Avellaneda', 'Oscar Rivas', 'Katherine Santamaria',
'Andrea Scarpetta', 'Wildiman Rodriguez'], dtype=object)
```

Fuente: imagen tomada del ambiente de Jupyter

Otro ejemplo es el de datos vacíos y la regla dice que se debe concatenar o actualizar con la unión de uno o dos o más campos, ver figura 4-28 el campo observaciones_proyecto esta vacío o tiene un dato que no concuerda del set de datos, aquí lo que se hace es unir la información de otros campos para poder identificarlo o asignarle un valor, se unió la información de otros campos que son proceso + estado_proceso + responsable_proceso y como resultado ver figura 4-29.

Figura 4-28: Estandarizar datos de los set de datos

```
In [166]: # Valores por defecto o que esten vacios
df_setTesoreria2[~df_setTesoreria2.observaciones_proyecto.isin(['Estado revisado', 'Estado desarrollo', 'Estado finalizado'])][
```

caso	proceso	estado_proceso	fecha_fin_planeada	fecha_final	observaciones_proyecto	
164	21090202	Incidentes	ejecución	2021-11-10 00:00:00	2021-11-25 00:00:00	NaN
165	21090203	Definición Alcance	cerrado	2021-09-26 00:00:00	2021-10-23 00:00:00	NaN
166	21090203	Objetivos de Alcance	cerrado	2021-10-02 00:00:00	2021-10-29 00:00:00	NaN
167	21090203	Cierre	iniciar	2021-10-05 00:00:00	2021-11-01 00:00:00	NaN
168	21090203	Definición	ejecución	2021-10-08 00:00:00	2021-11-04 00:00:00	NaN
...
98106	22010501	Pruebas	iniciar	2022-02-27 00:00:00	2022-03-19 00:00:00	NaN
98107	22010501	Cierre	ejecución	2022-03-02 00:00:00	2022-03-22 00:00:00	NaN
98108	22010501	Pruebas	cerrado	2022-03-05 00:00:00	2022-03-25 00:00:00	NaN
98109	22010501	Cierre	iniciar	2022-03-08 00:00:00	2022-03-28 00:00:00	NaN
98110	22010501	Versionamiento	ejecución	2022-03-11 00:00:00	2022-03-31 00:00:00	NaN

4192 rows x 6 columns

Fuente: imagen tomada del ambiente de Jupyter

Figura 4-29: Estandarizar datos de los set de datos

```
In [184]: # Valores por defecto o que esten vacios
df_setTesoreria2[~df_setTesoreria2.observaciones_proyecto.isin(['Estado revisado', 'Estado desarrollo', 'Estado finalizado'])][
```

caso	proceso	estado_proceso	fecha_fin_planeada	fecha_final	observaciones_proyecto	
164	21090202	Incidentes	ejecución	2021-11-10 00:00:00	2021-11-25 00:00:00	Incidentes-ejecución-Arnold Peña B
165	21090203	Definición Alcance	cerrado	2021-09-26 00:00:00	2021-10-23 00:00:00	Definición Alcance-cerrado-Ivan G. Avellaneda
166	21090203	Objetivos de Alcance	cerrado	2021-10-02 00:00:00	2021-10-29 00:00:00	Objetivos de Alcance-cerrado-Ivan G. Avellaneda
167	21090203	Cierre	iniciar	2021-10-05 00:00:00	2021-11-01 00:00:00	Cierre-iniciar-Ivan G. Avellaneda
168	21090203	Definición	ejecución	2021-10-08 00:00:00	2021-11-04 00:00:00	Definición-ejecución-Oscar Rivas
...
98106	22010501	Pruebas	iniciar	2022-02-27 00:00:00	2022-03-19 00:00:00	Pruebas-iniciar-Oscar Rivas
98107	22010501	Cierre	ejecución	2022-03-02 00:00:00	2022-03-22 00:00:00	Cierre-ejecución-Oscar Rivas
98108	22010501	Pruebas	cerrado	2022-03-05 00:00:00	2022-03-25 00:00:00	Pruebas-cerrado-Katherine Santamaria
98109	22010501	Cierre	iniciar	2022-03-08 00:00:00	2022-03-28 00:00:00	Cierre-iniciar-Katherine Santamaria
98110	22010501	Versionamiento	ejecución	2022-03-11 00:00:00	2022-03-31 00:00:00	Versionamiento-ejecución-Wildman Rodriguez

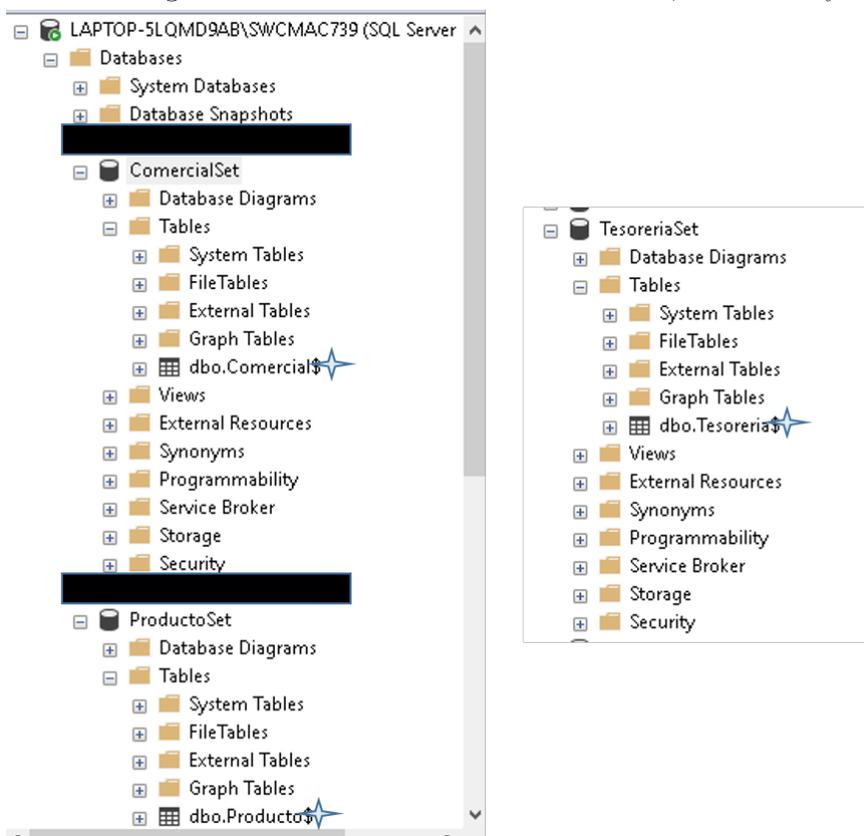
4192 rows x 6 columns

Fuente: imagen tomada del ambiente de Jupyter

Procesamiento

Para esta etapa por medio de Jupyter se envía cada set de datos de cada área a un motor de bases de datos SQLServer con el fin de integrar los tres set y dejar un solo repositorio final de la información, en la figura 4-30 se ve el cargue de los tres de datos (ComercialSet, ProductoSet, TesoreriaSet).

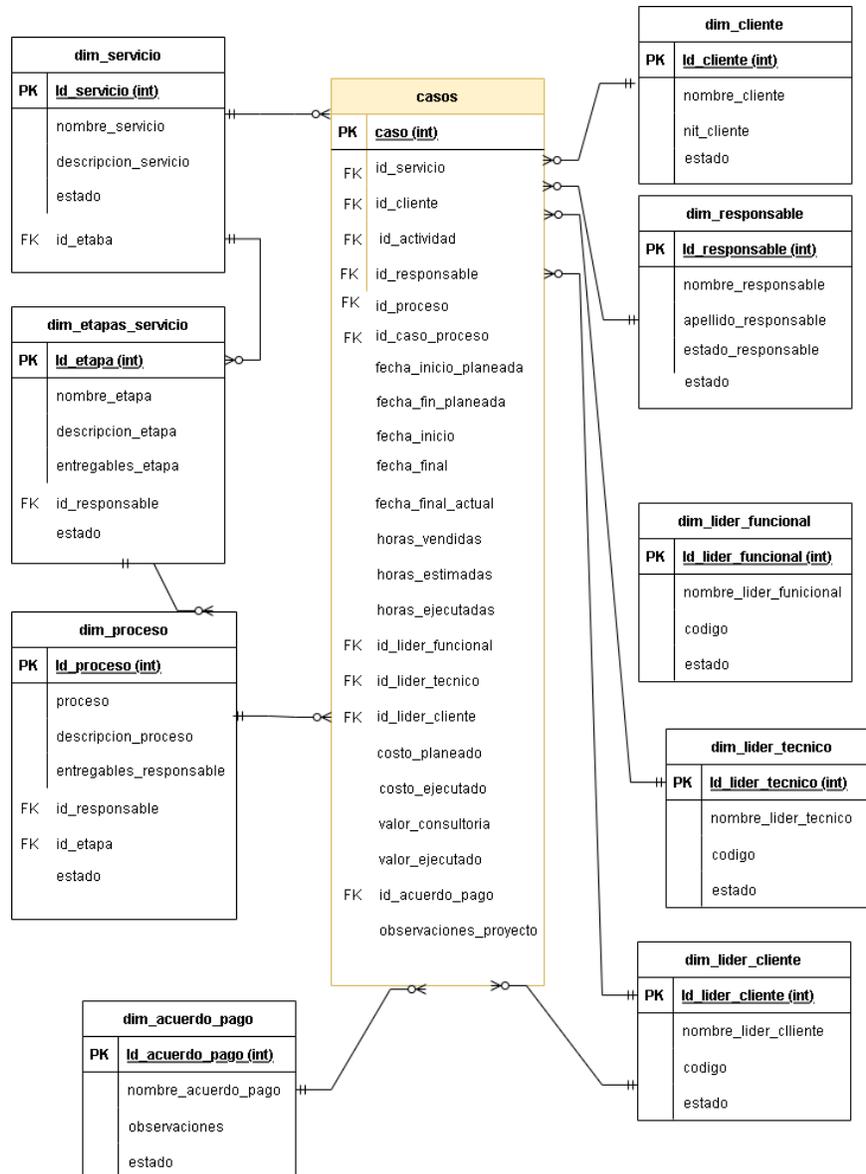
Figura 4-30: Cargue tres set de datos áreas Comercial, Producto y Tesorería



Fuente: imagen tomada del motor de bases de datos

Después se almacena el repositorio final en un motor de bases de datos SqlServer, pero antes se crea el modelo físico dimensional de la base de datos que va a contener la información ya procesada (ver figura 4-31).

Figura 4-31: Modelo Físico Dimensional



Fuente: elaboración propia

En diseño y desarrollo del modelo físico dimensional de nuestro DWH tendrá el siguiente detalle:

Dimensiones

El modelo está compuesto por 9 dimensiones, siete tipo 1 y dos dimensiones tipo 2 en la tabla 4-4 se describe la información.

Tabla 4-4: Descripción dimensiones

Nombre del objeto	Descripción	Tipo de objeto
DIM.SERVICIO	Esta dimensión corresponde a los tipos de servicio que hay en la Servicios Empresariales	Dimensión tipo 1
DIM.ETAPAS_SERVICIO	Dimensión que aloja las etapas del servicio	Dimensión tipo 1
DIM.PROCESO	Dimensión que tiene toda la información correspondiente a los procesos del servicio	Dimensión tipo 2
DIM.ACUERDO_PAGO	Esta dimensión contiene los tipos de acuerdos de pago que tiene la empresa para sus clientes	Dimensión tipo 1
DIM.LIDER_CLIENTE	Dimensión que contiene la información correspondiente a líderes de cada cliente	Dimensión tipo 1
DIM.LIDER_FUNCIONAL	Contiene la información de líderes funcionales asignados a cada cliente	Dimensión tipo 1
DIM.LIDER_TECNICO	Dimensión que contiene la información de líderes asignados a cada cliente	Dimensión tipo 1
DIM.RESPONSABLE	Esta dimensión corresponde a la información del consultor asignado para los proceso	Dimensión tipo 1
DIM.CLIENTE	En esta dimensión esta la información de los clientes que tienen algún servicio o producto con la empresa	Dimensión tipo 2

Fuente: elaboración propia

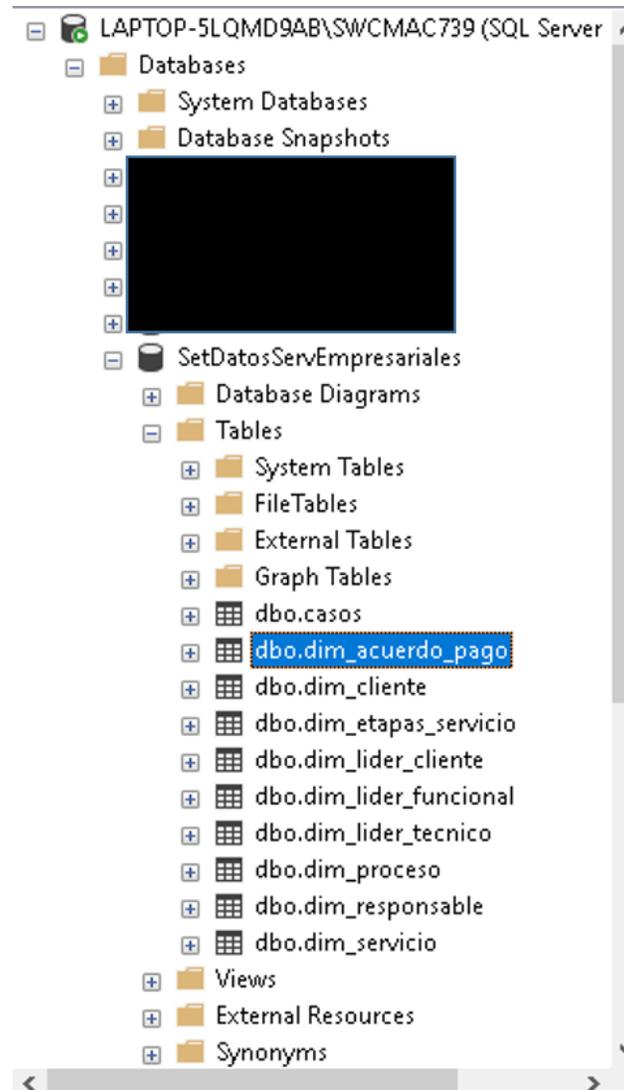
Hechos

El modelo cuenta con una tabla de hechos de Casos, la cual se compone de 24 columnas, 10 llaves foráneas que a su vez componen la llave primaria de la tabla de Casos.

Para realizar la creación del modelo de la base de datos se hace por medio de sentencias sql se crea los scripts, sentencias .ddl y para crear la base de datos, tablas y los respectivos campos de la tabla, con los respectivos constraint para persistir la información.

En la figura 4-32 se realiza el cargue al repositorio y el destino es un motor de bases de datos SqlServer.

Figura 4-32: Repositorio final cargado en SqlServer



Fuente: imagen tomada del motor de bases de datos

Posteriormente ya está lista la información para ser consumida o analizada según se requiera, allí estará para que la empresa de Servicios Empresariales ya sea por medio de herramientas de visualización de BI (Business Intelligence) lo pueda consultar. En la siguiente etapa se muestra la visualización de los datos

Vizualización de los Resultados

Para ver el resultado del proyecto se va a realizar por medio de una aplicación web elaborada

en el framework de Angular hecha por el área de desarrollo de la empresa de Servicios Empresariales en donde estará desplegada en un servidor interno de la empresa donde se podrá ver en la red local en donde cada colaborador ingresara por medio de una url, consumirá el servicio del motor de bases de datos para ver los resultados obtenidos, como por ejemplo a nivel general información por cada cliente (ver figura 4-33) y al darle click en la opción del icono del ojo podrá ver todos los casos de ese cliente que escogió, además información de cada uno de los casos (ver figura 4-34), en la figura 4-35 se muestra la terminal ejecutándose correctamente de forma local para la compilación de la página web.

Figura 4-33: Aplicación web sección Clientes

The screenshot shows a web browser at localhost:4200/clientes. The page title is 'Clientes'. There is a search bar with the placeholder text 'Cliente'. Below the search bar is a table with the following data:

Cliente	Casos en ejecución	Casos cerrados	Estado	actions
Bancolombia	8	4	Activo	
Caxdæc	5	3	Activo	
Compensar	4	12	Activo	
Ecopetrol S	2	0	Activo	
Colfondos	1	0	Activo	
Corficolombiana	3	4	Activo	
BBVA Colombia	1	3	Activo	
AV Villas	6	3	Activo	
Colsubsidio	5	9	Activo	
Banco Agrario de Colombia	14	12	Activo	

Fuente: imagen tomada del ambiente web

Figura 4-34: Aplicación web sección Casos

Caso	Servicio	Cliente	Observaciones	Estado	actions
22011205	Atención de Incidentes	Allianz	Estado revisado	Ejecución	
21120102	Requerimiento de ley	Caxdao	Estado desarrollo	Cerrado	
21120805	Capacitación	FAMSA	Estado revisado	Cerrado	
21090201	Versionamiento	Ecopetrol S.A.	Estado desarrollo	Ejecución	
22020210	Consultoría Funcional	Davivienda	Estado desarrollo	Ejecución	
21090207	Consultoría Técnica	Corficolombiana	Estado desarrollo	Ejecución	
21090101	Contrato Soporte	BEVA Colombia	Estado revisado	Iniciar	
21121701	Atención de Incidentes	AV Villas	Estado revisado	Cerrado	
21090106	Capacitación	Colsubsidio	Estado desarrollo	Ejecución	
21090103	Atención de Incidentes	AV Villas	Cierre-cerrado-Ivan G. Avellaneda	Ejecución	

Fuente: imagen tomada del ambiente web

En este dashboard (figuras 4-33 y 4-34) se muestra toda la información corregida y actualizada de las inconsistencias encontradas de las fuentes de información de las áreas de Comercial, Producto y Tesorería, por lo que le proporciona al comité de gerencia la información actualizada para realizar los seguimientos de los proyectos.

Figura 4-35: Aplicación web terminal ejecutando

```

ngcc (worker)
Date: 2022-05-05T03:33:52.518Z - Hash: 455e6a4f69cb96250122
6 unchanged chunks
chunk (main) main.js, main.js.map (main) 57.9 kB [initial] [rendered]
Time: 522ms
: Compiled successfully.

Date: 2022-05-05T03:36:16.916Z - Hash: 86a88088e533185fa1ae
6 unchanged chunks
chunk (main) main.js, main.js.map (main) 58 kB [initial] [rendered]
Time: 814ms
: Compiled successfully.

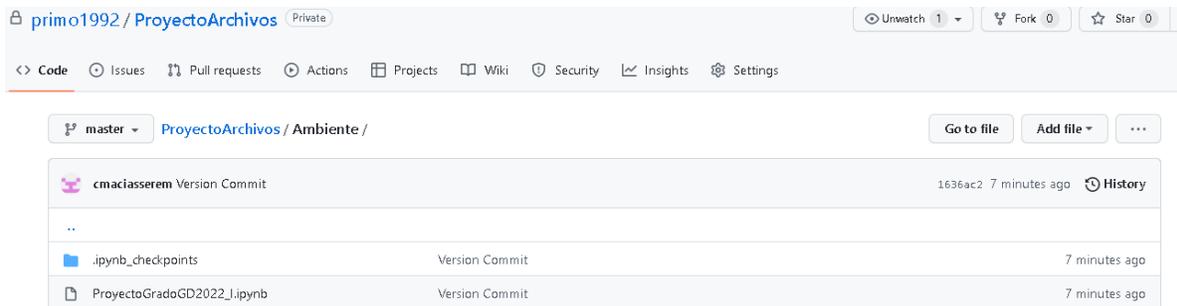
Date: 2022-05-05T03:38:41.585Z - Hash: 4da10a6515a3dba67f82
6 unchanged chunks
chunk (main) main.js, main.js.map (main) 58 kB [initial] [rendered]
Time: 815ms
: Compiled successfully.

Date: 2022-05-05T03:40:03.394Z - Hash: 99ca304792266dac549e
6 unchanged chunks
chunk (main) main.js, main.js.map (main) 58.1 kB [initial] [rendered]
Time: 728ms
: Compiled successfully.
    
```

Fuente: imagen tomada de la consola de gitbash

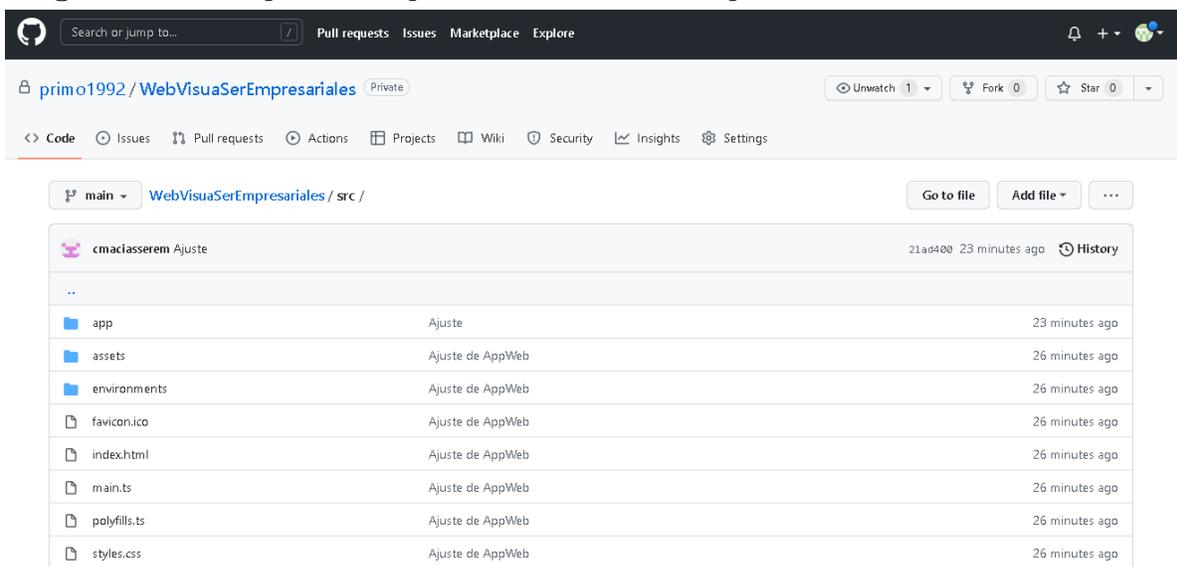
Por otra parte este proyecto tanto los notebooks, scripts y documentación está alojado en un repositorio donde hay creados tres ramas diferentes que son master, develop y documentación, en master encontramos el código finalizado de todo el proyecto del notebook, en la rama develop fue la evolución de todo el código y por último en la rama documentación esta todo lo relacionado con los scripts utilizados para el proyecto ver figura 4-36. También en la figura 4-37 se muestra el repositorio de la aplicación web sencilla elaborada para la visualización de los resultados.

Figura 4-36: Repositorio documentos del notebook



Fuente: imagen tomada del ambiente de GitHub

Figura 4-37: Repositorio aplicación web utilizada para la visualización de resultados



Fuente: imagen tomada del ambiente de GitHub

5 CONCLUSIONES

En el desarrollo del proyecto y la solución se demostró las falencias de cada set de datos de las áreas involucradas tanto Comercial, Producto y Tesorería implementando un dashboard con la solución de los proyectos tal que no manifiesten inconsistencias.

Además se realizó un proceso de análisis de datos de cada fuente de información de las áreas involucradas

Se implementó un modelo de datos para el repositorio final obtenido después de realizar el análisis a cada set de datos y realizando posteriormente la integración a un solo set de datos.

Para un trabajo a futuro se plantea alojar el repositorio final en una base de datos no relacional para la consulta por medio de una aplicación móvil, con el fin de tener la información a la mano en la organización.

6 Referencias

Angular (24 de febrero 2022). About Angular.

<https://angular.io/docs> - <https://angular.io/guide/architecture>

Database (2021). About database. <https://www.oracle.com/mx/database/what-is-database/>

Abraham, A., Pedregosa, F., Eickenberg, M., Gervais, P., Mueller, A., Kossaifi, J., Gramfort, A., Thirion, B y Varoquaux, G. (2014). Machine learning for neuroimaging with scikit-learn. Parte delantera. Neuroinform.

Alamoodi, A., Zaidan, B., Zaidan, A., Albahri, O., Mohammed, K., Malik, R., Almaahdi, E., Chyad, M., Tareq, Z., Albahri, A., Hameed, H. y Alaa, M. (2021). Sentiment analysis and its applications in fighting COVID-19 and infectious diseases: A systematic review.

Andy Opel, R. S. (2010). Fundamentos de SQL. McGraw-Hill.

Balagueró, T. (2018). ¿Qué son los datasets y los dataframes en el Big Data?

<https://www.deustoformacion.com/blog/programaciondiseno-web/que-son-datasets-dataframes-big-data>

Bell, C. (2018). Introducing the MySQL 8 Document Store. Apress.

Cobos, C., Muñoz, J., Mendoza, M., Acosta Muñoz, L. y Gómez, L. (2006). Bodega de datos y olap en unicauca virtual. pages 99–109.

Cortés Ramírez, J. D., Gélvez Navas, D. L. y Ordóñez Gil, L A. (2019). Proyecto de grado bodegas de datos caso adventure works cycles.

Golfarelli, M. (2009). Data warehouse design: modern principles and methodologies.

Kakish, K., Kraft, T. A. (2012). Etl evolution for real-time data warehousing. proceedings of the conference on information systems applied research.

Mysql (07 de febrero 2022). About Mysql. <https://www.hostinger.es/tutoriales/que-es-mysql>

Nieto Fagua, E. A., Yepes Guzmán, M. C. (2021). Modelo para el análisis de correlación de variables frente a la percepción de seguridad ciudadana en Colombia.

NumPy (2021). About NumPy. <https://numpy.org/about/>

Pandas (2021). About Pandas. <https://pandas.pydata.org/about/>

Pineda Suavita, A. C., Torres Gonzales, W. H. (2011). Prototipo de una bodega de datos para la empresa salón de automóvil.

Python (06 de abril 2022). Definición Python. <https://docs.python.org/3/tutorial/index.html>

R. Kimball., M. Ross. (2002). The data warehouse toolkit: the complete guide to dimensional modeling.

Roldan Pinzón, D. E. (29 de julio 2017). Diseño de una guía general para construir una bodega de datos del área de ventas de una empresa.

<https://repository.unilibre.edu.co/handle/10901/11042>

Silberschatz, A., Korth, H.F. y Sudarshan, S. (2011). Database System Concepts, 6th Edition. McGraw Hill.

What's New In Python documentation, (2021).

<https://docs.python.org/3/whatsnew/3.9.html>