**Portada:**

* **Título del documento**

**Documentación del Proyecto: Sistema de Limpieza y Almacenamiento de Datos con MySQL**

* **Autor(es)**
* **Fecha de creación**
* **Versión del documento**

** Índice:**

* **Tabla de contenido que facilite la navegación.**

** Introducción:**

**Propósito del documento.**

Este documento tiene como objetivo describir el proceso de desarrollo y las principales características del proyecto, dónde se realizará carga, limpieza y procesamiento de datos utilizando Python y Apache Spark. También detallará el propósito, el alcance y las funcionalidades que llevará al sistema implementado. Así también como las técnicas y herramientas empleadas para el desarrollo de este proyecto. Este documento servirá como guía para. Entender el flujo de trabajo y la arquitectura, los resultados esperados del sistema.

**Alcance del proyecto.**

El proyecto abarca el desarrollo de un sistema de limpieza y el procesamiento de datos masivos utilizando Python y Apache Spark. El objetivo principal es garantizar que los datos provenientes de un conjunto de archivos CSV y JSON puedan ser limpiados, transformado en pie parado para su respectivo análisis.

Los procesos claves incluyen eliminación de valores nulos, creación de columnas derivada y ejecución de cálculo, normalización de los datos y conversión de datos, Este sistema se implementa en un entorno de procesamiento distribuido capaz de manejar grandes volúmenes de datos.

**Breve descripción del sistema o producto.**

El sistema desarrollado es una solución de limpieza y procesamiento de datos basada en Apache, Spark y Python, que son herramientas orientadas a la transformación eficiente de los datos a gran escala. El proyecto se centra. En el procesamiento de un conjunto de dataset que contienen información sobre terrazas comerciales, urbanas, incluyendo de dimensiones, ubicaciones, permiso, horario y operaciones.

**El flujo de trabajo incluye las siguientes etapas:**

1. **Creación del entorno de trabajo:** Se instalarán todas las librerías y herramientas adecuadas para el desarrollo del proyecto
2. **Lectura de datos:** Se utilizará un conjunto de archivos como fuente de entrada donde se cargarán los datos con Apache Spark y convertirlo en un DataFrame distribuido.
3. **Limpieza de datos:** eliminación de filas con valores nulos. Más de 50% de sus columnas. También se identifican y se corrigen inconsistencia en los datos.
4. **Concatenación archivos:** A partir de archivos limpios y existentes, se generan uniones y concatenación de archivos
5. Procesamiento vertical y horizontal:
6. **Procesamiento vertical:** se realizan transformaciones en las columnas como normalización y limpieza de datos
7. **Procesamiento horizontal:** Se añaden nuevos registros mediante la carga de múltiples archivos y se combinan datasets.
8. **Salida de datos:** Los datos procesados se exportan y se almacenan en formato apto para su análisis posterior en bases de datos SQL.

**Requisitos:**

**Requisitos funcionales y no funcionales.**

Los requisitos funcionales definen las funciones y capacidades del sistema que debe cumplir para lograr los objetivos del proyecto.

1. Lecturas de archivos
2. Limpieza de datos
3. Transformación de datos
4. Almacenamientos de datos procesados
5. Escalabilidad del procesamiento

* **Requisitos y no funcionales.**

Los requisitos no funcionales describen las características de calidad y restricciones bajo los cuales se debe operar el sistema.

1. Rendimiento
2. Disponibilidad y confianza
3. Escalabilidad
4. Mantenibilidad
5. Seguridad
6. Compatibilidad

**Especificaciones del sistema**

**Arquitectura del Sistema:**

La arquitectura del sistema describe como los diferentes componentes interactúan y se comunican para realizar el procesamiento de data. En este proyecto, la arquitectura está basada en un modelo distribuido que utiliza Apache Spark para limpieza, transformación y análisis de datos. Como una integración hacia el sistema de almacenamiento y bases de datos.

1. **1-Capa de Ingesta**

* Los archivos que contienen toda la información se almacenan en un sistema local
* Se utiliza Apache Spark para leer y cargar los datos desde este sistemas de archivos hacia un dataframe distribuido

1. **Capa de procesamiento de datos (Limpieza y transformación):**

* Utilizando PySpark, los datos se limpian eliminando filas con datos faltantes o duplicados y se transforman mediante la creación de nuevas columnas derivadas.
* El sistema utiliza una serie de reglas para realizar operaciones sobre los datos como cálculos de sumas multiplicación y otros.

1. **Capa de Almacenamiento y Exportación:**

* Los datos procesados Exportan a archivos CSV limpios y se almacenan en base de datos SQL o No SQL para su posterior análisis.

**Diagrama de arquitectura.**

A continuación, se muestra un diagrama de la arquitectura que describe el flujo de datos:

Datos Crudos

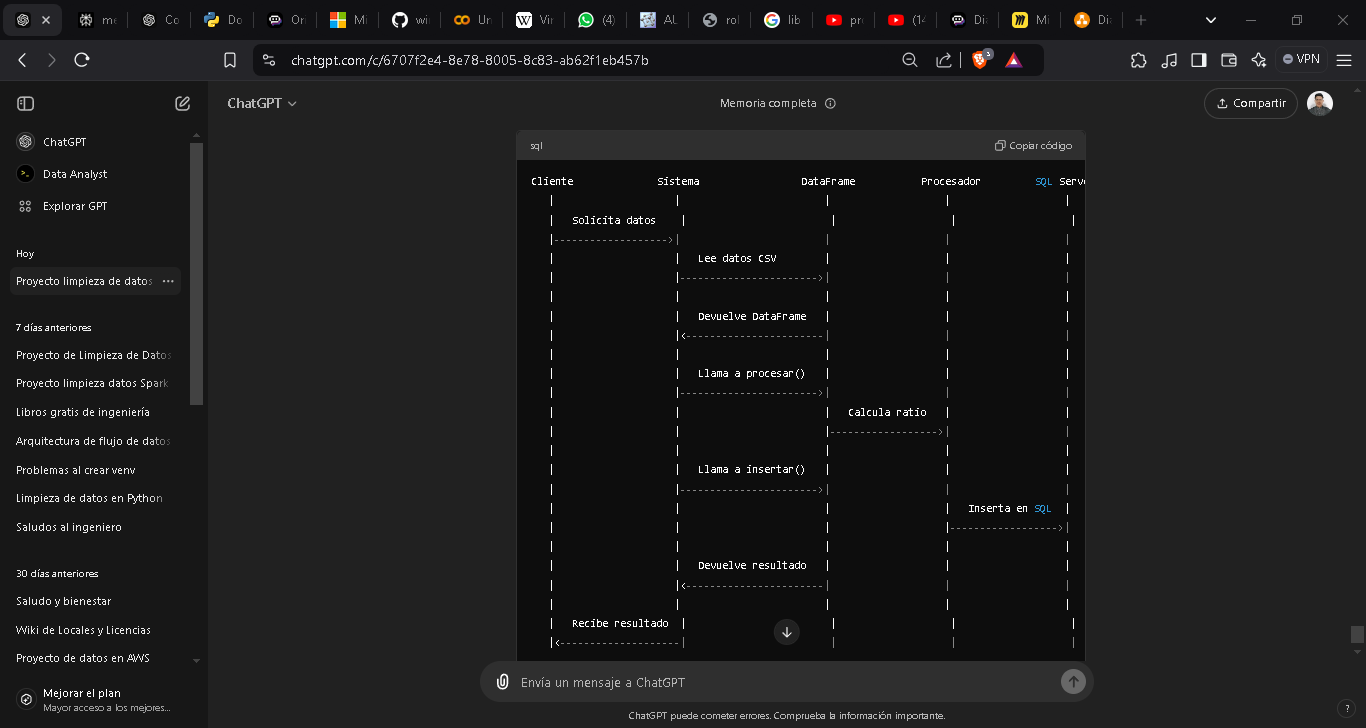
Limpieza de Datos (Script)

Transformación (Calculo de Ratios y Nuevas Columnas)

Creación de Tablas en SQL (Si no existen)

Inserción de Datos en SQL

**Diseño del Sistema:**

****Diagrama de Secuencia de inserción de los datos a SQL (clase, secuencia, etc.).

**Implementación:**

1. **Descripción del entorno de desarrollo.**

* El entorno de desarrollo para este proyecto está configurado de la siguiente manera.

1. **Lenguaje de programación:**

* Python es el lenguaje principal utilizado para desarrollar, escribir de limpieza y transformación.
* Spark principal componente para el procesamiento de datos a grandes volumen.

1. **Entorno de Desarrollo Integrado (IDE).**

* Se utilizó visual Studio Code (VSCode) para el desarrollo del Código en Python aprovechando su integración con entornos virtuales.

1. **Entorno virtual.**

* El proyecto utiliza un entorno virtual de Python (venv) para gestionar las dependencias y paquetes de manera aislada, asegurando que las versiones de las librerías utilizadas no interfieran con otras aplicaciones.
* Los paquetes principales instalados en el enfermo mental incluyen:
* Pyspark Para el procesamiento de transformación de datos con Apache Spark.
* Pymsql y pyodbc Para conectarse a la base de datos es el QSL o MySql para consultas necesarias.

1. **Configuración de bases de datos**

* SQL Server 2019 es la versión del servidor de la base de datos utilizada para almacenar los datos limpios. La conexión se establece mediante los drivers adecuados configurando el servidor con usuarios y contraseñas necesarias para acceder a la instancia.

1. **Ejecución de Pipeline scripts .bat**

* El scripts principal es (app.bat) donde se ejecuta en la línea de comando, activando todo el entorno y ejecutando la aplicación.
* Este entorno garantiza que el proyecto sea portable, controlado y fácil de replicar en otros equipos o entorno de producción

**Documentacion Tecnica**

**Herramientas y tecnologías utilizadas**

**1. Creación del Entorno Virtual (venv)**

Para aislar las dependencias del proyecto, creamos un entorno virtual utilizando venv en el directorio del proyecto. Esto nos permitió tener un entorno controlado y evitar conflictos con otras instalaciones globales de Python en el sistema

Pasos:

1-Navegamos al directorio donde se encuentra el proyecto:

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

2-Creamos el entorno virtual con el siguiente comando:

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Esto creó un nuevo directorio llamado venv, que contiene los archivos del entorno virtual.

3-Activamos el entorno virtual (para sistemas Windows):

Texto

Descripción generada automáticamente

**2. Archivo requirements.txt**

Creamos el archivo requirements.txt para especificar las dependencias que se deben instalar en el entorno virtual. Este archivo contenía las librerías necesarias para el proyecto, tales como pyspark, mysql-connector-python, y pandas.

**Contenido del archivo requirements.txt:**

**Interfaz de usuario gráfica, Texto

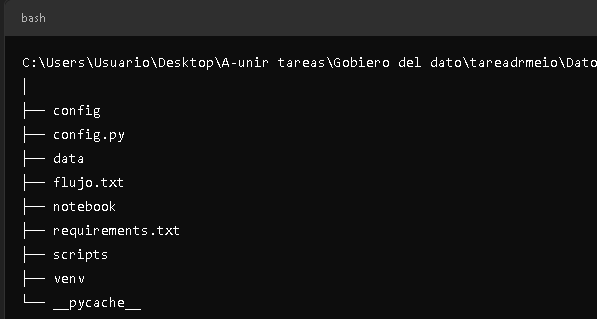
Descripción generada automáticamente con confianza media**

**Instalación de Dependencias:**

Una vez que el entorno virtual estuvo activo, ejecutamos el siguiente comando para instalar todas las dependencias listadas en el archivo requirements.txt:

**Estructura del Directorio del Proyecto**

Después de crear el entorno y ejecutar los requerimientos, el proyecto debería tener la siguiente estructura:

****

* **config/:** Contiene configuraciones adicionales.
* **config.py:** Archivo de configuración principal que centraliza credenciales y rutas.
* **data/:** Directorio donde se almacenan los datos en bruto.
* **flujo.txt:** Documento que describe el flujo de trabajo.
* **notebook/:** Directorio para los Jupyter notebooks si fuera necesario.
* **requirements.txt**: Lista de dependencias.
* **scripts/:** Contiene scripts Python para la extracción, limpieza y almacenamiento de datos.
* **venv/:** El entorno virtual donde se instalaron las dependencias.
* **pycache/:** Directorio generado automáticamente para almacenar los archivos compilados de Python.

**Manual de ejecución del Pipeline de limpieza**

Debes tener configurado un entorno virtual (venv) en Python con las dependencias necesarias instaladas, incluyendo pyspark y pymssql/pyodbc.

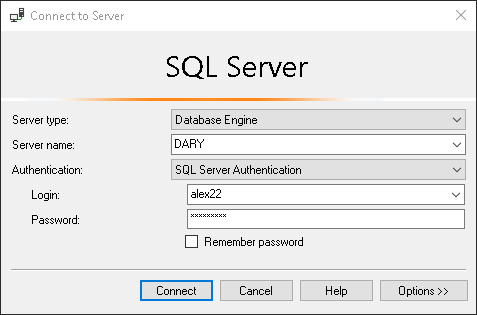
Como ya tenemos listo el entorno realizaremos los siguientes pasos:

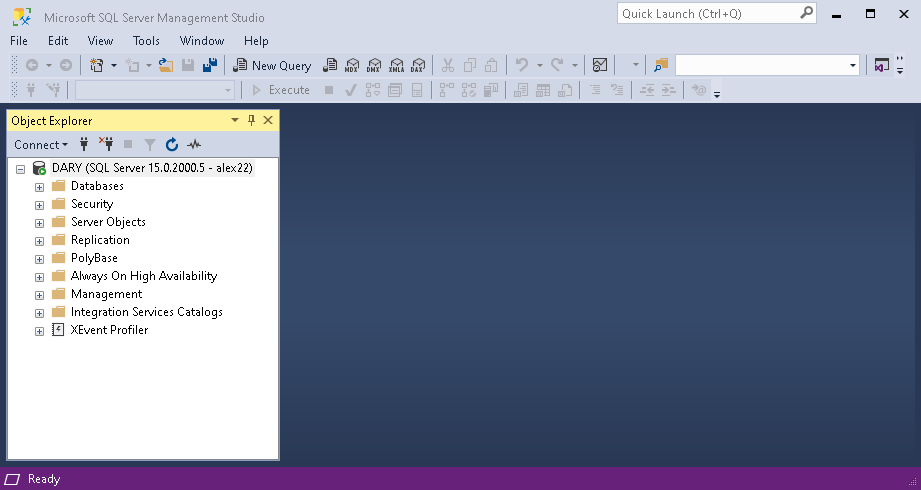
* 1. Texto

     Descripción generada automáticamenteDebemos estar en la ruta correcta donde creamos nuestro entorno
  2. Texto

     Descripción generada automáticamenteIngresamos el siguiente comando para poder trabajar dentro del entorno y preparar el proceso de limpieza   
     # .\venv\Scripts\activate  
       
       
       
       
       
     Podemos observar que ingreso correctamente al entorno .
  3. **Acceso a la Base de Datos SQL Server**:

Nos aseguramos de tener las credenciales correctas para conectarte a la base de datos de SQL Server donde se almacenarán los datos limpios.

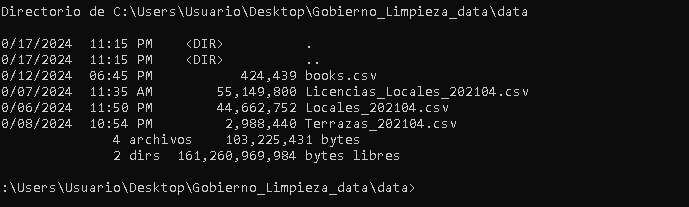




Las credenciales donde se guardarán los archivos limpios son correctas

* 1. **Archivos CSV de Entrada**:

Los archivos deben estar ubicado en la ruta correcta para que el script pueda leerlo.



Preparación de los scripts para el proceso de limpieza

* 1. Creamos un archivo config donde estén las rutas de los archivos a limpiar y las credenciales de la bases de datos

Texto

Descripción generada automáticamente

Usamos un archivo de configuración porque es una práctica recomendada en el desarrollo de software que mejora la seguridad, mantenibilidad, facilidad de uso y separación de responsabilidades. Es esencial para gestionar las credenciales y configuraciones necesarias para que el proyecto funcione correctamente.

* 1. Un reloj digital en la pantalla

     Descripción generada automáticamente con confianza mediaLimpieza y transformación de los datos   
     -Funciones clave utilizadas en el proceso de limpieza y transformación de los datos   
     Carga de datos

Filtrados de nulos

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamenteNormalizar columnas

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza mediaCalcular Ratio

Texto

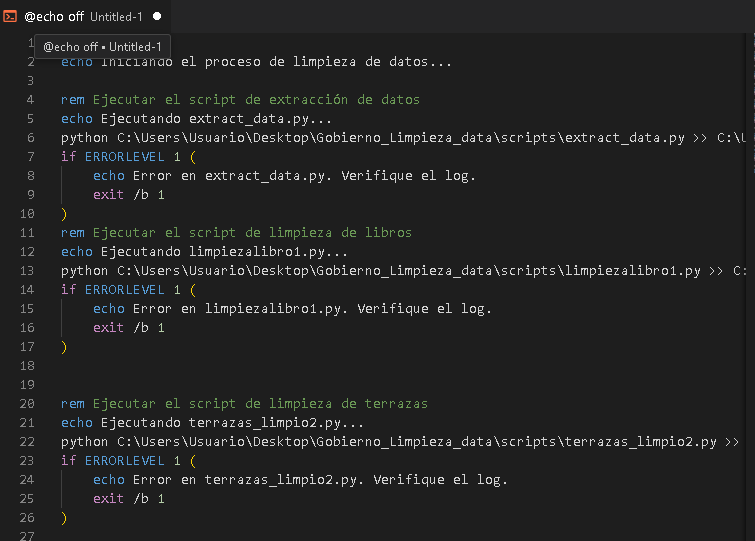
Descripción generada automáticamente con confianza mediaConvertir horas en formato adecuado

* 1. **Resultados Esperados**

Un conjunto de datos limpio y normalizado que puede ser utilizado para análisis posteriores o ser insertado en una base de datos SQL Server.

* 1. **Ejecución del Pipeline**

Se utiliza el archivo por lotes**run\_app.bat**

El archivo run\_app.bat es un script de Windows que se utiliza para ejecutar automáticamente el pipeline de carga, limpieza y procesamiento de datos. Este archivo permite a los usuarios iniciar todo el proceso con un solo comando, simplificando la ejecución del proyecto.   


* 1. Ejecución del pipiline
     1. Texto

        Descripción generada automáticamenteIngresamos al entorno del proyecto
     2. **Texto

        Descripción generada automáticamente**Ingresamos a la ruta donde se encuentra nuestro archivo **run\_app.bat**
     3. **Texto

        Descripción generada automáticamente con confianza media**Dentro de la ruta ejecutamos el archivo **run\_app.bat**
     4. Texto

        Descripción generada automáticamenteEl proceso se ejecutó correctamente   
        Interfaz de usuario gráfica, Texto

        Descripción generada automáticamente
     5. Interfaz de usuario gráfica, Texto, Aplicación

        Descripción generada automáticamenteVerificamos los datos se cargaron correctamente