|  |
| --- |
| SISTEMA DE BI SOBRE EL CLIMA EN NAVARRA |
| PROCESO ETL |
| FECHA DE ENTREGA:  7 diciembre  ANÁLISIS DE APLICACIONES EMPRESARIALES  IÑIGO AZNÁREZ GIL |



# ÍNDICE

1. Introducción…………………………………………………………………... Pág. 3
2. Arquitectura del proceso……………………………………………….. Pág. 4
3. Proceso ETL…………………………………………………………………… Pág. 5-7
   1. Extracción…………………………………………………………….. Pág. 5
   2. Transformación……………………………………………………. Pág. 6
   3. Carga…………………………………………………………………….. Pág. 7
4. Anexo: Capturas de ejecución………………………………………… Pág. 8-9

# INTRODUCCIÓN

El proceso ETL (Extracción, Transformación y Carga) se utiliza para obtener datos climáticos históricos de estaciones automáticas en Navarra, transformarlos a un formato estándar y cargarlos en una base de datos relacional para su posterior análisis. Este documento detalla cada paso del proceso, incluyendo las operaciones de extracción, transformación y carga.

# ARQUITECTURA DEL PROCESO

Lenguaje de programación: El proceso ETL se desarrolla utilizando Python, lo que permite manejar tareas de automatización, manipulación de datos y conexión con bases de datos de manera eficiente.

Base de datos: Los datos procesados se almacenan en MySQL, utilizando la biblioteca pymysql para establecer la conexión y realizar operaciones SQL.

Directorio de trabajo: El proceso organiza los datos en las siguientes carpetas:

* *datos\_climatologicos*: Carpeta destinada a almacenar los archivos CSV descargados desde las fuentes originales. Estos archivos contienen datos climáticos en bruto provenientes de las estaciones automáticas.
* *datos\_transformados*: Carpeta donde se guardan los archivos procesados y estandarizados tras la etapa de transformación. Estos datos están listos para cargarse en la base de datos.

Fuentes de datos: <https://meteo.navarra.es/estaciones/descargardatos.cfm>

Este sitio proporciona enlaces a los datos históricos de estaciones automáticas.

Los datos se encuentran en formato CSV, organizados por años y asociados a cada estación climática.

# PROCESO ETL

# EXTRACCIÓN

Objetivo: Descargar datos CSV de estaciones automáticas correspondientes a los años 2006-2023.

Pasos del proceso:

1. Obtener estaciones automáticas:
   1. Se realiza una solicitud GET a la página de estaciones y se analiza el HTML con BeautifulSoup.
   2. Se filtran las estaciones automáticas (icono estacionautomatica.gif).
   3. Se extraen el ID y el nombre de cada estación.
2. Descargar archivos CSV:
   1. Para cada estación, se accede a la página específica y se recopilan los enlaces a los archivos CSV.
   2. Se descargan únicamente los archivos correspondientes a los años 2006-2023.
   3. Los archivos se renombran con un identificador único y se almacenan en el directorio *datos\_climatologicos*.

Errores manejados:

* Página no accesible.
* Ausencia de archivos CSV para una estación.
* Años fuera del rango especificado.

Funciones: script descargar\_datos\_transformar.py

* *obtener\_estaciones\_automaticas()*: Obtiene una array con los Ides de las estaciones automáticas para diferenciarlas de las manuales.
* *descargar\_datos():* Descarga los csvs de las estaciones automáticas que cumplen con la condición de contener datos de los últimos 18 años.

# TRANSFORMACIÓN:

Objetivo: Estandarizar los datos descargados y preparar un formato adecuado para la carga en la base de datos.

Operaciones realizadas:

1. Formato de fecha:
   1. Convertir las fechas al formato YY/MM/DD usando la función *transformar\_fecha().*
2. Reemplazar caractéres:
   1. Reemplazar “,” por “.” de manera que tenemos todos los valores decimales con puntos.
   2. Reemplazar “;” por “,” para tener todos los campos delimitados por comas.
3. Generación de archivos transformados:
   1. Guardar los datos procesados en la carpeta *datos\_transformados*.

Tras realizar la transformación ejecutar el script comprobar\_csv.py para verificar que todas las filas contienen el mismo número de campos que la cabecera.

Errores manejados:

* Formato de fecha no válido.
* Arhivos mal formateados.

Funciones: script descargar\_datos\_transformar.py

* *transformar\_fecha():* Transforma la fecha al formato correcto
* *transformar\_datos():* Realiza las modificaciones necesarias a los csvs de la carpeta *datos\_climatologicos*  y los almacena en el directorio *datos\_transformados.*

CARGA

Objetivo: Insertar los datos transformados en la base de datos MySQL.

Pasos del proceso:

1. Conexión a la base de datos: Se establece conexión con el servidor MySQL mediante pymysql.
2. Preparación de la base de datos:
   1. Vaciar las tablas existentes para evitar duplicados.
   2. Crear las tablas necesarias: LK\_COMARCAS, LK\_MUNICIPIOS, LK\_FECHAS y DT\_DATOS\_CLIMATICOS.
   3. Insertar datos de referencia (comarcas y municipios).
3. Carga de datos:
   1. Leer los archivos CSV transformados.
   2. Insertar registros en las tablas correspondientes (LK\_FECHAS y DT\_DATOS\_CLIMATICOS).

Errores manejados:

* Fallos en la conexión a la base de datos.
* Violaciones de claves foráneas.

Funciones: archivo crear\_tablas\_insertar.py

* *conectar\_bd() :* Realiza la conexión a la base de datos.
* *vaciar\_tablas():* Desactiva claves foráneas y vacia el contenido de las tablas para su posterior inserción y evitar duplicados.
* *crear\_tablas():* Borra las tablas por si existén para evitar errores y las vuelve a crear.
* *insertar\_comarcas():* Inserta las comarcas de Navarra.
* *insertar\_municipios():* Inserta los municipios de Navarra cada uno con el identificador de su comarca correspondiente.
* *insertar\_fechas():* Inserta las fechas desde el 01/01/2006 hasta el 31/12/2023.
* *procesar\_csv():* Procesa el contenido del archivo csv para obtener los campos que queremos insertar en la base de datos.
* *insertar\_datos\_climáticos():* Inserta los datos climáticos de los csvs transformados.

# ANEXO: CAPTURAS DE EJECUCIÓN

Texto

Descripción generada automáticamentedescargar\_datos\_transformar.py:

Texto

Descripción generada automáticamentecrear\_tablas\_insertar.py:

Texto

Descripción generada automáticamente