# Proyecto ETL de Videojuegos

Este proyecto realiza un proceso ETL (Extracción, Transformación y Carga) desde múltiples APIs públicas del mundo del videojuego y del universo Pokémon (RAWG, GiantBomb, Pokémon Showdown) hacia una base de datos relacional MySQL alojada en AWS RDS. Los datos son estructurados, normalizados y almacenados en formato **Parquet** antes de su carga final, permitiendo análisis eficientes y escalables.

## Despliegue

1. Requisitos

* Python 3.12
* Instalar dependencias:
* pip install mysql-connector-python pymysql sqlalchemy pandas requests
* Acceso a base de datos MySQL (AWS RDS)
* API Keys:
  + RAWG 🡪 [https://rawg.io/apidocs](https://rawg.io/apidocs)
  + GiantBomb 🡪 [https://www.giantbomb.com/api/](<https://www.giantbomb.com/api/>)
  + Pokémon-moves 🡪 (<https://play.pokemonshowdown.com/data/moves.json>)
  + Pokémon-pokedex 🡪 (https://play.pokemonshowdown.com/data/pokedex.json)

1. Estructura

proyecto\_videojuegos

* Proyecto Gabriel Rodriguez.py
  + - logs/
      * + registro\_etl.log
        + carga\_global.log
    - registro\_etl.csv
    - carga\_global.csv
    - \*.parquet

1. Ejecución

Extracción y guardado local

python Proyecto Gabriel Rodriguez.py

Archivos generados

* rawg\_games.parquet
* giantbomb\_games.parquet
* pokemon\_pokedex.parquet
* pokemon\_moves.parquet

1. Carga en base de datos

python Proyecto Gabriel Rodriguez.py

1. ¿Cómo Funciona?
   * Extracción

* RAWG: Videojuegos y metadatos.
* GiantBomb: Juegos, plataformas y ratings.
* Pokémon Showdown:
  + pokedex.json: Datos de Pokémon.
  + moves.json: Información de ataques.
  + Transformación
* Conversión de fechas.
* Desanidamiento de diccionarios (baseStats, abilities, flags).
* Normalización de columnas (snake\_case, sin acentos ni símbolos).
  + Carga
* Se realiza con `SQLAlchemy`, añadiendo columnas técnicas:

|  |  |
| --- | --- |
| Campo | Descripción |
| dw\_fecha\_registro | Fecha/hora de carga |
| dw\_id\_carga | UUID único por carga |
| dw\_deleted | Marcado lógico (por defecto: False) |
| dw\_source | Fuente de origen |
| dw\_endpoint | URL consultada para la extracción |

Además, se registran las operaciones en las tablas `log\_operaciones` y `log\_etl\_fuentes `.

1. Consultas de ejemplo

-- Top 10 juegos con mayor rating en RAWG

SELECT name, rating, released

FROM rawg\_games

ORDER BY rating DESC

LIMIT 10;

-- Top Pokémon por peso

SELECT name, weightkg

FROM core\_pokemon\_pokedex

ORDER BY weightkg DESC

LIMIT 5;

1. Logs
   * Local: logs/registro\_etl.log, logs/carga\_global.log
   * CSV: registro\_etl.csv/ log\_operaciones.csv
   * BD: Tabla: log\_operaciones y log\_etl\_fuentes