

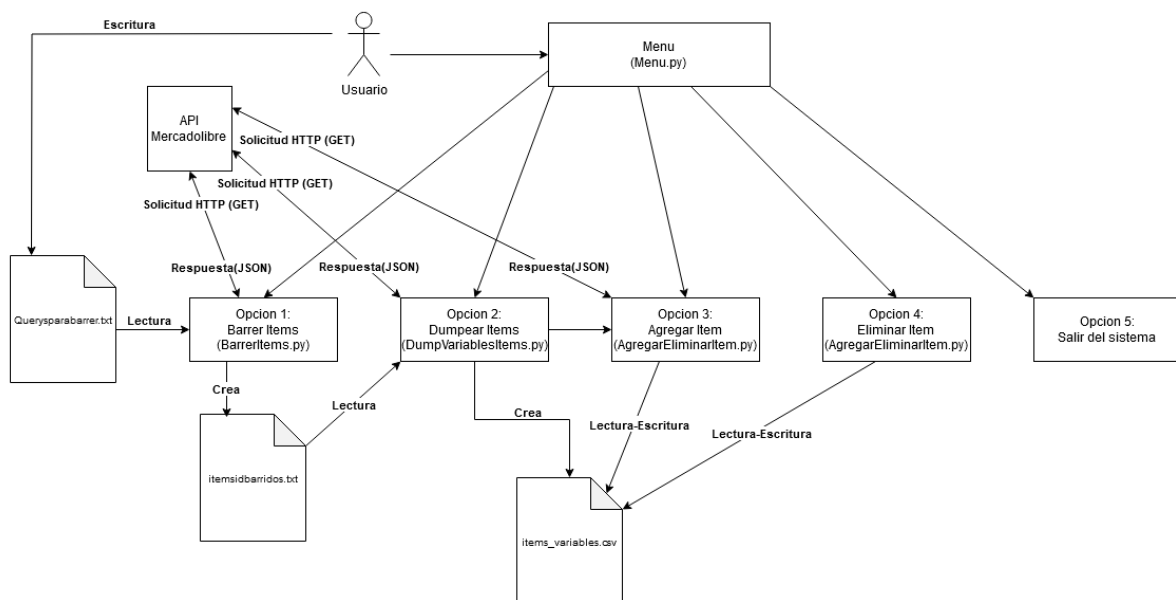
1. Descripción de la Solución

Este proyecto consiste en un menú interactivo que permite realizar consultas y gestionar ítems de Mercado Libre a través de su API pública. El programa tiene las siguientes funcionalidades:

1. **Barrer Ítems:** Realiza consultas al API para obtener un listado de ítems según las queries proporcionadas por el usuario y guarda sus IDs en un archivo de texto.
2. **Dumpear Ítems:** Extrae las variables necesarias de los ítems utilizando sus IDs y las almacena en un archivo CSV.
3. **Agregar Ítems:** Permite agregar un ítem al archivo CSV proporcionando su ID, sin duplicados.
4. **Eliminar Ítems:** Permite eliminar un ítem del archivo CSV proporcionando su ID.

El sistema está compuesto por tres scripts principales que realizan tareas específicas y un menú interactivo que permite al usuario ejecutar cada una de estas tareas.

2. Diagrama de Alto Nivel



3. Documentación de los Pasos

Paso 1: Barrer Ítems desde Query (Opcion 1 del menú)

- El usuario proporciona las queries que desea ejecutar en el archivo "Querysparabarrer.txt".
- Se ejecuta la función `barrer_items` que realiza las consultas a la API de Mercado Libre.
- Los IDs de los ítems obtenidos se guardan en el archivo "itemsidbarridos.txt" luego de haber chequeado que no haya duplicados.

Paso 2: Dumpear Variables de los Ítems al CSV (Opcion 2 del menú)

- Se ejecuta la función `dump_item_variables` que procesa cada ID en `itemsidbarridos.txt` y consulta al API para obtener las variables específicas de cada ítem.
- Las variables se guardan en el archivo CSV llamado "items_variables.csv".

Paso 3: *Agregar Ítem por ID al CSV (Opcion 3 del menú)*

- El usuario puede agregar un nuevo ítem al archivo CSV "items_variables.csv" proporcionando solo el ID del ítem. Si el ítem ya existe en el CSV se le notificara al usuario.
- Se verifica que el archivo CSV "items_variables.csv" exista, de no ser así le notificara al usuario que debe ejecutar las opciones 1 y 2 del menú para crear el CSV.
- El programa consulta la API para extraer las variables de ese ítem y las agrega al CSV, verificando que no haya duplicados.

Paso 4: *Eliminar Ítem por ID del CSV (Opcion 4 del menú)*

- El usuario puede eliminar un ítem del archivo CSV "items_variables.csv" proporcionando solo el ID. Si el ítem no existe en el CSV se le notificara al usuario.
- Se verifica que el archivo CSV "items_variables.csv" exista, de no ser así le notificara al usuario que debe ejecutar las opciones 1 y 2 del menú para crear el CSV.
- El programa buscará el ID en el archivo items_variables.csv y eliminará la entrada correspondiente.

Paso 5: *Interfaz de Usuario Interactiva*

- El programa muestra un menú interactivo para que el usuario seleccione las acciones a realizar.
- El menú ofrece cinco opciones: barrer ítems, dumpear ítems, agregar ítem, eliminar ítem y salir.

4. Requisitos del Entorno

- **Python 3.7 o superior:** Para ejecutar el script principal y sus dependencias.
- **Librerías necesarias:** requests, csv, os, json.
- **Archivos necesarios:** Querysparabarrer.txt para ingresar queries, itemsidbarridos.txt para guardar los IDs barridos, items_variables.csv para almacenar las variables.

5. Consideraciones de Diseño

- **Gestión de Errores:** El código maneja excepciones para evitar errores en caso de que los archivos de entrada no existan o que las consultas a la API fallen.
- **Escalabilidad:** La solución es flexible para que el usuario pueda agregar o eliminar ítems al CSV sin necesidad de realizar un barrido completo.
- **Optimización:** El sistema evita duplicados en el CSV, mejorando la gestión de datos sin necesidad de reescribir ítems ya existentes.

6. Posibles Mejoras

- A futuro se podría agregar una nueva opción al menú para pedirle al usuario ingresar las queries que desee de forma interactiva (con sus respectivas validaciones y manejo de

errores) y no depender del archivo Querysparabarrer.txt que en esta versión se debe editar de forma manual.