Universidad Gerardo Barrios

Programación II – Proyecto Parcial

Integrantes

- José Alberto Velázquez Paz (SMSS051924)
- Luis Ángel Zúniga Menjívar (SMSS081824)
- Víctor Arnoldo Iglesias Sandoval (SMSS070523)
- Ricardo Alberto Mendiola Hernández (SMSS061924)

Docente

Willian Villegas Montes

Fecha de Entrega

20/10/2025

Librerías Utilizadas

1. Selenium

¿Para qué sirve?

Selenium es una herramienta de automatización de navegadores web. Permite controlar navegadores como Chrome, Edge o Firefox mediante código Python, simulando las acciones de un usuario real.

¿Cuándo usarla?

- Sitios web con contenido dinámico cargado con JavaScript (React, Vue, Angular).
- Cuando se necesita interactuar con elementos (clicks, formularios, scroll).
- Páginas que requieren inicio de sesión.
- Sitios con contenido que se carga al hacer scroll (scroll infinito).
- Cuando se necesita esperar que elementos específicos aparezcan en la página.

Casos de uso en nuestro proyecto:

- Navegar a URLs automáticamente para acceder a sitios de comercio electrónico.
- Esperar la carga de productos antes de extraer datos.
- Simular scroll para activar la carga de más productos.
- Evadir detección de bots modificando propiedades del navegador.

2. BeautifulSoup

¿Para qué sirve?

BeautifulSoup es una librería para analizar y extraer datos de documentos HTML y XML. Convierte el código HTML en una estructura de datos navegable con la que podemos buscar elementos fácilmente.

¿Cuándo usarla?

- Extraer datos específicos de páginas web (precios, títulos, descripciones).
- Buscar elementos por etiquetas, clases CSS o IDs.
- Navegar por la estructura jerárquica del HTML.
- Limpiar y procesar texto extraído.
- Usarla cuando el contenido HTML ya está cargado (después de Selenium).

Casos de uso en nuestro proyecto:

- Parsear HTML obtenido por Selenium para convertirlo en un objeto navegable.
- Buscar productos mediante clases CSS específicas.
- Extraer atributos como URLs de imágenes o enlaces de productos.
- Filtrar contenido no deseado o publicitario.

Casos de Uso del Web Scraper

E-commerce (Caso implementado)

- Sitios: eBay, Amazon, Mercado Libre, AliExpress.
- Datos extraídos: Productos, precios, enlaces, imágenes.
- Aplicaciones: Comparadores de precios, análisis de mercado, monitoreo de competencia.

Noticias y Medios

- Sitios: Periódicos digitales, blogs, portales de noticias.
- Datos extraídos: Títulos, fechas, contenido, autores.
- Aplicaciones: Agregadores de noticias, análisis de tendencias, detección de noticias falsas.

Bienes Raíces

- Sitios: Portales inmobiliarios.
- Datos extraídos: Precios, ubicaciones, características, fotos.

- Aplicaciones: Análisis de mercado inmobiliario, alertas de nuevas propiedades.

Empleo

- Sitios: LinkedIn, Indeed, Computrabajo.
- Datos extraídos: Ofertas laborales, salarios, requisitos.
- Aplicaciones: Análisis de mercado laboral, alertas de empleo.

Redes Sociales

- Sitios: Twitter, Instagram, Facebook (con limitaciones).
- Datos extraídos: Publicaciones, comentarios, estadísticas.
- Aplicaciones: Análisis de sentimiento, monitoreo de marca.

Datos Financieros

- Sitios: Yahoo Finance, Google Finance.
- Datos extraídos: Cotizaciones, históricos, noticias financieras.
- Aplicaciones: Trading algorítmico, análisis de inversiones.

Solución de Problemas Comunes

Problema: No se encontraron elementos

- Aumentar los tiempos de espera.
- Verificar los selectores CSS en el navegador (F12).
- Revisar si hay CAPTCHA.
- Guardar la página como page_full.html para depurar.

Problema: WebDriver no encontrado

- Instalar EdgeDriver manualmente.
- Usar webdriver-manager para instalación automática.
- Verificar la versión del navegador.

Problema: Página bloqueada o CAPTCHA

- Usar proxies rotativos.
- Cambiar el User-Agent.
- Resolver CAPTCHA manualmente.
- Usar APIs oficiales si están disponibles.

Problema: Datos incompletos extraídos

- Aumentar el scroll y los tiempos de espera.
- Verificar que JavaScript haya terminado de cargar.
- Inspeccionar la estructura HTML manualmente.