

Web Scraper con Puppeteer y MongoDB

Este proyecto realiza **web scraping** en **eBay** utilizando **Puppeteer** para extraer productos relacionados con "Fernando Alonso".

Los datos recolectados incluyen: **nombre**, **precio** e **imagen del producto**, y se almacenan en una **base de datos MongoDB**.

Tecnologías utilizadas

- Node.js
- Puppeteer
- MongoDB con Mongoose
- JSON para almacenamiento local

Instalación y configuración

Clona este repositorio

```
https://github.com/MSS1410/WEB-SCRAP---EBAY.git
```

Instala las dependencias

```
npm install
```

Configura MongoDB

Crea un archivo `.env` en la raíz del proyecto y agrega tu cadena de conexión a MongoDB:

```
DB_URI=mongodb+srv://admin:scrapbooking@cluster0booking.sqmut.mongodb.net/?  
retryWrites=true&w=majority&appName=Cluster0Booking
```

Ejecuta el scraper

```
npm run start
```

El script **buscará productos en eBay**, **recorrerá todas las páginas** y **guardará los datos en MongoDB**.

Estructura del proyecto

```
web-scraper-ebay  
├── models  
│   └── Product.js # Modelo de MongoDB para productos  
├── pageScraper.js # Lógica de scraping con Puppeteer  
├── index.js       # Punto de entrada del script  
├── browser.js     # Configuración del navegador Puppeteer  
├── db.js          # Conexión a MongoDB  
└── .env           # Variables de entorno (MongoDB URI)
```

```
└─ package.json    # Dependencias y configuración de NPM
└─ README.md       # Documentación del proyecto
```

Consultar datos en MongoDB

Abre **MongoDB Compass** o usa la terminal: