



universidad
de león



GRADO EN INGENIERÍA INFORMÁTICA

Memoria del proyecto
Sistema de Información de Gestión
y Business Intelligence

Aplicación de sistema de recomendación



Autor: XIAO YANG

(ENERO-FEBRERO, 2023)

ÍNDICE

1.	Descripción del problema	3
2.	Herramientas utilizadas	5
1.	Herramientas utilizadas para construir base de datos	5
2.	Herramientas utilizadas para construir Backend.....	6
3.	Herramientas utilizadas para construir Frontend.....	7
3.	Aplicación	10
4.	Base de datos.....	11
1.	Obtener los datos	11
2.	Limpiar los datos.....	14
3.	Crear instancia en Neo4j aura.....	15
4.	Importar los datos	16
5.	Crear nuevos nodos y relaciones.....	18
6.	Visualización de base de datos.....	19
7.	Interfaz de usuario (Frontend)	26
8.	Servidor (Backend).....	36
9.	Algoritmos de recomendación	40
10.	Análisis de resultados	49
11.	Análisis D.A.F.O.....	57
1.	Debilidades	57
2.	Amenazas	57
3.	Fortalezas	58
4.	Oportunidades	58
12.	Líneas de futuro	59
13.	Lecciones aprendidas.....	60
14.	Referencias y Bibliografía.....	61

1. Descripción del problema

Hoy en día, la mayoría de las personas poseen un ordenador personal, sin embargo, la mayoría no está segura de cómo elegir el mejor ordenador para sus necesidades. Elegir el ordenador adecuado puede ser una tarea desalentadora, ya que hay una gran variedad de opciones en el mercado, incluyendo diferentes marcas, modelos y especificaciones. Esto puede provocar una sobrecarga de información y resultar abrumador para quienes no están familiarizados con la tecnología.

Aquí describo alguno de las dificultades que pueden encontrar:

1. Compatibilidad: Esto es un problema importante debido a la variedad de especificaciones y requisitos que existen en el mercado. Por ejemplo, una placa madre puede requerir un tipo específico de procesador o memoria RAM, y es importante asegurarse de que los componentes sean compatibles entre sí antes de comprarlos.
2. Presupuesto: Es fácil gastar más de lo que se quiere o se puede en componentes de alta calidad, pero es importante tener en cuenta que no siempre es necesario gastar mucho dinero para tener un ordenador funcional.
3. Especificaciones: Con tantos modelos y especificaciones diferentes en el mercado, es fácil sentirse abrumado y no saber cuál es el mejor componente para satisfacer sus necesidades. Por ejemplo, un usuario puede no saber cuánta memoria RAM o almacenamiento necesita para sus necesidades diarias.
4. Confianza en la marca: Con tantas marcas diferentes, es difícil saber cuál es la mejor opción y cuál es confiable. Además, los precios y la calidad varían entre las diferentes marcas, lo que puede hacer que sea difícil tomar una decisión informada.

Sin embargo, si busca una tienda física para consultar los detalles de los componentes del ordenador, la mayoría de las veces ellos dicen un precio más alto que el precio que puede encontrar en las páginas web.

Para resolver esos problemas es muy útil usar una aplicación de recomendación que recomienda a los usuarios cómo comprar los componentes.

La utilidad de una aplicación recomendadora es que puede proporcionar recomendaciones personalizadas basadas en sus necesidades y presupuesto.

La aplicación es capaz de buscar los mejores componentes que cumplan con los usos diarios del usuario, así como de asegurarse de que los componentes sean compatibles entre sí y con el ordenador que se está construyendo.

En resumen, una aplicación recomendadora puede ayudar a las personas a comprar componentes de manera más eficiente.



A continuación voy a explicar todos los detalles sobre “YX PC RECOMMENDER.”

2. Herramientas utilizadas

1. Herramientas utilizadas para construir base de datos

- a. **Web Scraper** es una extensión de Google Chrome que permite a los usuarios extraer información de páginas web y guardarla en una estructura de datos, como una hoja de cálculo, un archivo CSV o un archivo JSON. Esta extensión es especialmente útil para aquellos que necesitan recopilar información de muchas páginas web de manera automatizada y eficiente.



- b. **Excel**, es una herramienta valiosa para visualizar, limpiar y organizar datos. Utilizo Excel para limpiar los datos extraídos por Web Scraper, que contiene algunos caracteres vacíos, cadenas no estructuradas, contiene campos innecesarios, etc.



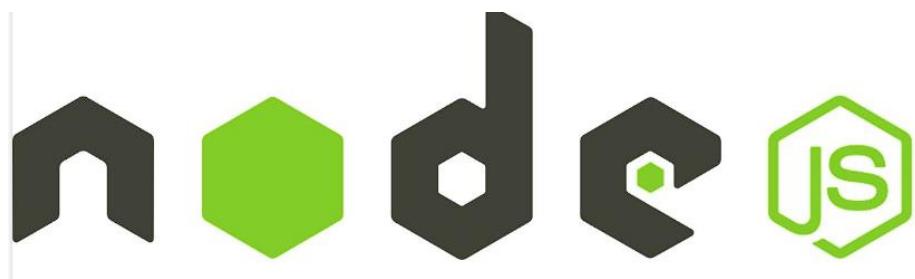
- c. **Neo4j** es un sistema de gestión de bases de datos NoSQL basado en grafos. Almacena la información en forma de nodos y enlaces para gestionar eficazmente consultas complejas entre elementos de datos. Es popular para aplicaciones de recomendación,

inteligencia artificial, análisis de redes sociales y seguimiento relacional.



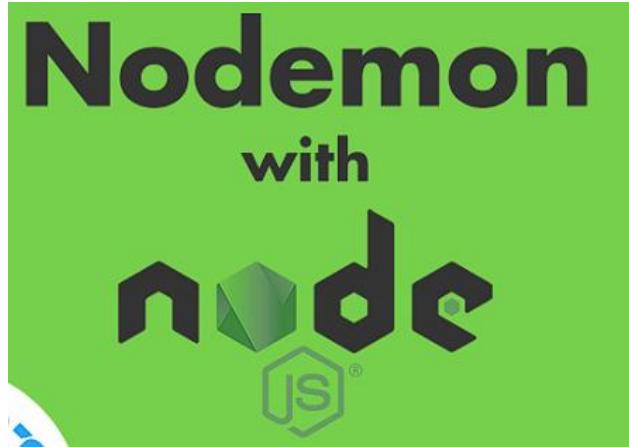
2. Herramientas utilizadas para construir Backend

- a. **Node.js** es un entorno de ejecución de JavaScript en el servidor que permite a los desarrolladores crear aplicaciones web de alta velocidad y escalabilidad. Utiliza un modelo de programación basado en eventos y es compatible con una amplia gama de módulos y bibliotecas. Es ideal para aplicaciones que requieren un alto rendimiento y capacidad de procesamiento.



- b. **Express** es un marco de aplicación web para Node.js. Es una biblioteca que simplifica la creación de aplicaciones web y API en Node.js. Proporciona una serie de herramientas y funcionalidades para facilitar la configuración de servidores HTTP, la gestión de solicitudes y respuestas, la gestión de middleware y la integración de otras tecnologías y bibliotecas.
- c. **Nodemon** es una herramienta de desarrollo para Node.js que facilita el proceso de desarrollo y depuración de aplicaciones al

monitorear automáticamente los cambios en el código y reiniciar el servidor de forma automática.



- d. **CORS (Cross-Origin Resource Sharing)** es un mecanismo de seguridad que limita la capacidad de una página web para acceder a recursos en otro dominio.
- e. **Librería config** es una herramienta popular para el manejo de configuraciones en aplicaciones de backend en Node.js. Permite a los desarrolladores definir valores de configuración en un archivo separado y acceder a ellos en el código de la aplicación.
- f. **Biblioteca neo4j-driver** es una biblioteca de software que proporciona una interfaz de programación de aplicaciones (API) para acceder a una base de datos Neo4j desde una aplicación de Node.js.

3. Herramientas utilizadas para construir Frontend

- a. **React** es una biblioteca de JavaScript de código abierto desarrollada por Facebook para el desarrollo de aplicaciones web, con un enfoque basado en componentes que permite crear aplicaciones dinámicas y de alta calidad. Es conocido por su rendimiento y escalabilidad, y cuenta con una amplia comunidad y documentación en línea.

- b. **Axios** es una biblioteca de JavaScript para hacer solicitudes HTTP cliente. Permite a los desarrolladores realizar solicitudes HTTP de manera sencilla y eficiente, lo que les permite recuperar o enviar datos desde un servidor o API. Axios es compatible con la mayoría de los navegadores y se puede utilizar en aplicaciones web y móviles. También proporciona funciones útiles como la interceptación de solicitudes y respuestas, y la gestión de errores, lo que lo hace una herramienta popular y versátil para el desarrollo de aplicaciones web.



- c. **Lodash** es una biblioteca de JavaScript que proporciona una amplia gama de funciones útiles y optimizadas para trabajar con datos y objetos. Mejora la productividad del desarrollador y simplifica tareas comunes como la manipulación de arrays y objetos, control de flujo y creación de funciones personalizadas. Es altamente optimizado, se integra fácilmente y es ampliamente utilizado en aplicaciones web y móviles.

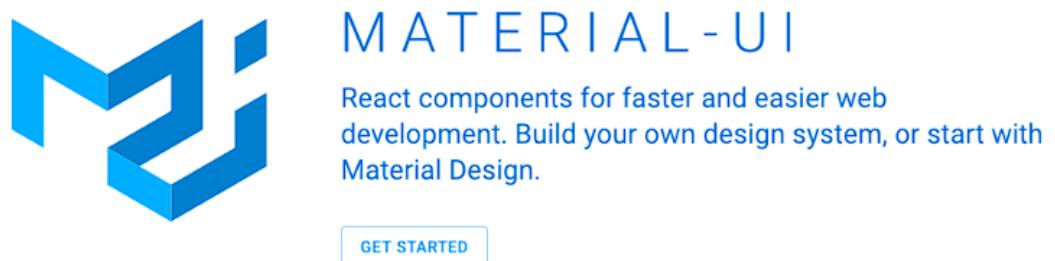


- d. **Flexbox Grid** es una librería de diseño web basada en flexbox que ofrece un sistema de cuadrícula fácil de usar y altamente personalizable. Permite crear diseños adaptativos y complejos con facilidad, definiendo posición de elementos, ancho de columnas, distribución de elementos, etc. Compatible con diferentes tamaños de pantalla y se integra bien con otras librerías y marcos. Es una

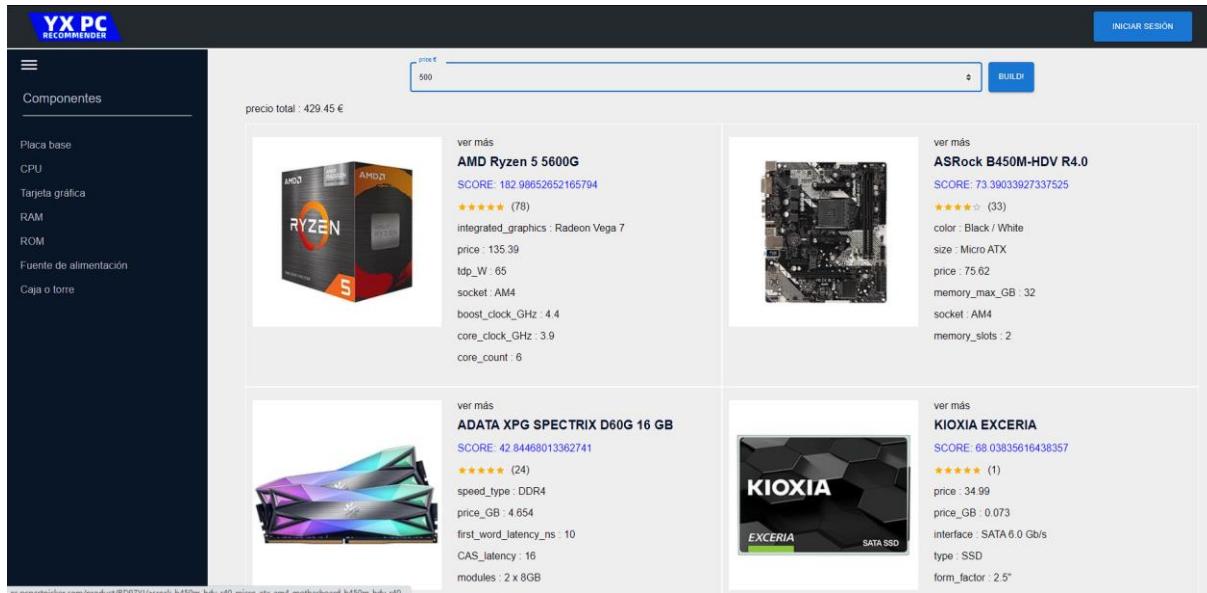
herramienta popular para el desarrollo de aplicaciones web con una estética y una funcionalidad atractivas.

Flexbox Grid

- e. **Librería Material-UI** es una biblioteca de componentes de React que implementa el diseño de Material Design de Google. Ofrece una amplia gama de componentes atractivos y personalizables, y es ampliamente utilizado por su facilidad de uso y rendimiento óptimo.



3. Aplicación



YX-PCRECOMMENDER es una Aplicación web que recomienda a los usuarios los mejores componentes para montar su propio ordenador personal.

Los usuarios pueden ver el ranking de todos los componentes ajustando a su presupuesto y su preferencia de uso del ordenador.

Puede ir a la página donde puede realizar la compra, detecta automáticamente las compatibilidades de los componentes y otros aspectos.

Las valoraciones de los productos se calcula cada vez que pulsa el botón 'build!',

Las valoraciones se calculan teniendo en cuenta sus rendimientos, precios, y también el número de valoraciones y su puntuación obtenida por los usuarios.

El usuario puede elegir su preferencia de uso del ordenador para obtener una recomendación que cubre mejor a su necesidad, los principales usos son los siguientes: Uso estándar, trabajar en la oficina, videojuegos y editar videos.

En los siguientes puntos veremos más detalles de la aplicación.

4. Base de datos

Para un sistema de recomendadora pueden funcionar correctamente uno de los factores más importantes son los datos.

Neo4j es una base de datos gráfica que se caracteriza por su capacidad para manejar relaciones complejas entre entidades y proporcionar resultados de consulta de forma eficiente. A continuación se enumeran algunas de las principales características de Neo4j.

1. **Modelo de grafos:** Neo4j utiliza un modelo de grafos en el que las entidades se representan como nodos y las relaciones entre ellas se representan como enlaces o relaciones. Este modelo permite representar de manera natural la complejidad de las relaciones en los datos.
2. **Consultas rápidas:** Neo4j utiliza un lenguaje de consulta llamado Cypher para realizar consultas y recuperar datos de la base de datos. Este lenguaje es altamente eficiente y permite realizar consultas complejas en tiempos de ejecución rápidos.
3. **Escalabilidad:** Neo4j es escalable y permite a los usuarios manejar grandes cantidades de datos y relaciones con eficiencia.
4. **Integración:** Neo4j se integra fácilmente con otras tecnologías, como aplicaciones web, bases de datos relacionales y sistemas de big data, lo que permite una mayor flexibilidad y eficiencia en el manejo de los datos.
5. **Comunidad activa:** Neo4j tiene una comunidad activa y una amplia gama de recursos y documentación disponibles, lo que facilita el aprendizaje y el uso de la tecnología.

En esta sección voy a construir un base de datos basado en grafos de conocimiento utilizando Neo4j Aura.

los pasos para construir son los siguientes:

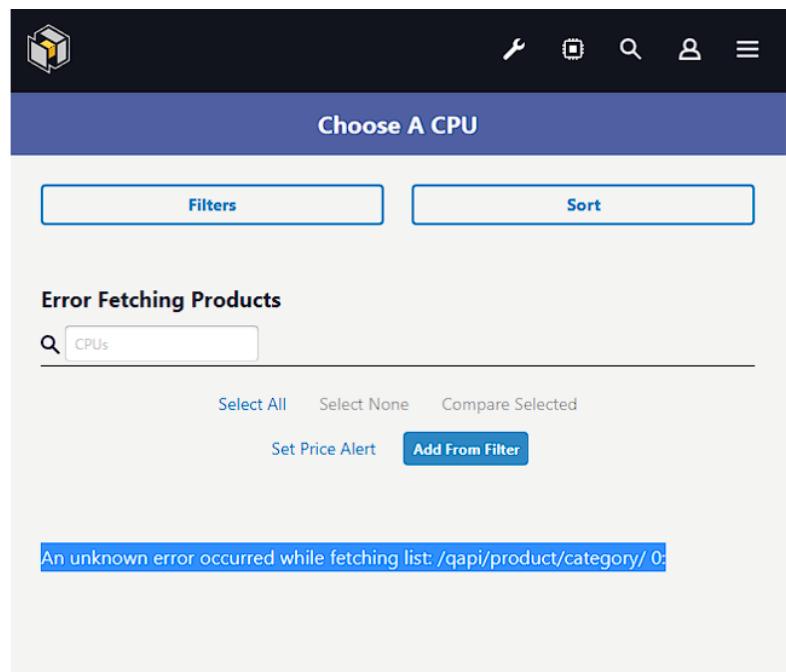
1. Obtener los datos

Para obtener los datos lo primero que hice fue buscar si existiera ya los datos que necesito en algunas páginas como Kaggle, PC componentes, GitHub, etc.

Pero, por desgracia, no encontré ninguna base de datos en la red que me gustara porque, o bien estaba incompleta y era antigua, o bien era de pago, así que tuve que aprender scraper para encontrar los datos que yo necesito.

Para ello, empecé probando con la librería de Python que se llama "Scrapy" , Scrapy es una herramienta de extracción de datos de código abierto que tiene como principal objetivo extraer la información de determinado sitio web, que funciona en mayoría de las páginas pero no en la página <https://es.pcpartpicker.com/products/cpu/>.

La que he decidí a rastrear, resulta que Scrapy no ha sido capaz de sacar los datos de los productos.



Lo pregunté ese problema en stackoverflow y el problema está en que esta página utiliza algún tipo de protección contra scrapers.

stack overflow
en español

Buscar...

- Comunidad Bot el 28 nov. 2022 a las 15:06

Pues en la página es.pcpaprticker.com/products/cpu tiene 1317 productos con datos, son datos que quiero extraer, pero no aparece esos datos en `response`. No sé qué puede faltar, probé : -Aumentar `DOWNLOAD_DELAY`. -Poner `USER_AGENT` como un navegador chrome. Pero sigue pasa lo mismo.

- Xiao el 28 nov. 2022 a las 16:25

Añade un comentario

1 respuesta Ordenado por Mayor puntuación (por defecto) ▾

Scrapy es una herramienta de extracción de datos que puede ayudarte a extraer información de páginas web. Sin embargo, si no puedes ver los datos que deseas extraer en la respuesta obtenida después de ejecutar el comando "view(response)", es posible que la página web utilice algún tipo de protección contra scrapers.

Una posible solución sería utilizar una cabecera de navegador falsa para hacer que Scrapy se presente como un navegador real y no como un scraper. Puedes hacer esto modificando el diccionario "DEFAULT_REQUEST_HEADERS" en tu archivo de configuración de Scrapy.

También puedes intentar utilizar una herramienta de desarrollador del navegador (como la consola de JavaScript en Google Chrome) para ver qué datos están siendo cargados en la página y cómo puedes acceder a ellos desde Scrapy.

También puedes intentar utilizar otras herramientas de extracción de datos como BeautifulSoup, Selenium o PyQuery. Cada una de estas herramientas funciona de manera diferente, así que tendrás que investigar un poco para ver cuál se adapta mejor a tus necesidades.

Otra opción es intentar contactar con el administrador de la página web y preguntarle si puedes acceder a los datos que deseas extraer de alguna manera. Si la página ofrece una API, por ejemplo, podría ser más fácil acceder a los datos que deseas.

Compartir Editar Seguir respondida el 2 dic. 2022 a las 22:37

 Cristian Zúñiga 435 • 1 • 4

Probé modificar `DEFAULT_REQUEST_HEADERS` pero no consigo obtener los datos. Las otras herramientas no creo que funcione tampoco, si están protegido los datos.. Necesito estos datos para hacer un trabajo universitario, entonces busco otras páginas para mi necesidad, pero muchísimas gracias. – Xiao el 5 dic. 2022 a las 17:56

Entonces probé otras herramientas y al final encuentro “Web Scraper”, es una extensión de Google que sí me funcionó para obtener los datos y exportarlos en los archivos CSV.

Para rastrear los datos tienen que crear un Sitemap que son especificaciones de urls y los campos de datos que se va a rastrear, entonces he creado un Sitemap para cada tipo de componentes del ordenador.

Durante el proceso la página bloqueó mi IP por hacer muchas peticiones en pocos tiempos, se resuelve cambiar la dirección IP, y ajustar el tiempo de intervalo.

The screenshot shows the 'Web Scraper' tab selected in the top navigation bar. Below it, there's a search bar for 'Sitemaps'. A table lists several sitemaps with their corresponding IDs and domains:

ID	Domain
cases-pcpartpicker	es.pcpartpicker.com
cpu-pcpartpicker	es.pcpartpicker.com
graphicCard-pcpartpicker	es.pcpartpicker.com
motherBoard-pcpartpicker	es.pcpartpicker.com
powerSupplies-pcpartpicker	es.pcpartpicker.com
ram-pcpartpicker	es.pcpartpicker.com
rom-pcpartpicker	es.pcpartpicker.com

Posteriormente Comprobé los datos son completos y los exporto en archivos CSV.

scraper_order	web-scraper.start.url	img_src	name	type	color	power_supply	side_panel	price	link	link_href
1672947222-893	https://es.pcpartpicker.com/products/case#sort=price&page=12	/cdna.pcpartpicker.com/static/forever/images/product/66138802582221c0532ee46d2752f400.256p.jpg	Fractal Design Meshify 2 XL	TypeATX Full Tower	Color/Black	Power Supply/None	Side Panel/Tinted	null	Fractal Design Meshify 2 XL (1)	https://es.pcpartpicker.com/product/2hmP8/fractal-design-meshify-2-is-atx-full-tower-case-fd-c-mesh2v-02
1672947224-894	https://es.pcpartpicker.com/products/case#sort=price&page=12	/cdna.pcpartpicker.com/static/forever/images/product/6811902d92132cb0c139e9b2e5c319030.256p.jpg	BEars b-Voguish	TypeATX Mid Tower	Color/Black	Power Supply/None	Side Panel/Tempered Glass	null	BEars b-Voguish (0)	https://es.pcpartpicker.com/product/knT2W/bearsb-voguish-alu-mid-tower-case-b-voguish
1672947226-895	https://es.pcpartpicker.com/products/case#sort=price&page=12	/l media-amazon.com/images/I/312+HK-MOMI.jpg	DIYPC J180	TypeATX Mid Tower	Color/White	Power Supply/None	Side Panel/Acrylic	null	DIYPC J180 (6)	https://es.pcpartpicker.com/product/9rD0k8/diypc-case-j180w

2. Limpiar los datos

Es muy importante limpiar los datos para evitar los problemas a posteriori como facilitar el query con cypher.

Los procesos para limpiar los datos son los siguientes:

1. Usar la opción Regex que aporta Web Scraper para eliminar caracteres raros.

The screenshot shows the configuration for the 'price' field. It includes fields for 'Id' (set to 'price'), 'Type' (set to 'Text'), 'Selector' (set to 'Select'), and 'Data' (set to 'Price'). There is also a checkbox for 'Multiple' which is unchecked. The 'Regex' field contains the pattern '[0-9]+[.]*[0-9]*'. The 'Parent Selectors' section lists '_root', 'product', and 'link'.

En este caso concreto, para obtener el campo precio, elimino todos los

caracteres que no fuesen números ni puntos.

- Usar Excel eliminar los campos que ponen 'null', eliminar columnas que no aporta datos útiles, separar algunos campos en dos, normalizar los datos por columnas (Siendo cada columna una propiedad) y eliminar los productos que no tienen ninguna valoraciones (Considero que no tiene apenas compradores y no es bueno recomendar a los usuarios.)

Datos antes de limpiar:

web-scr1	web-scr2	img-src	name	core count	perfoman	perfoman	tpc	integratec	price	link	link-href	rating	socket1	socket2
167294482 https://es.pcpartpicker.com/	https://m.pcpartpicker.com/	Intel Core	Intel Core	4	3.5	null	95	Integrated	null	Intel Core	https://es.pcpartpicker.com/	(7 Ratings, 4.7 Average)	Intel HD Graphics 3000	LGA1155
167294482 https://es.pcpartpicker.com/	https://m.pcpartpicker.com/	Intel Core	Intel Core	6	4	5	95	Integrated	null	Intel Core	https://es.pcpartpicker.com/	(17 Ratings, 4.9 Average)	LGA1151	Coffee Lake
167294482 https://es.pcpartpicker.com/	https://m.pcpartpicker.com/	Intel Core	Intel Core	4	3.5	4.1	65	Integrated	null	Intel Core	https://es.pcpartpicker.com/	(20 Ratings, 4.8 Average)	LGA1151	Kaby Lake-S

Datos después de limpiar:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
img_src	name	core count	core clock	boost clock	tdp_W	integratec	price	link	socket	rating_nuri	rating					
//cdn.pcpartpicker.com/	Intel Pentium Gold G6400	2	4		58	Intel UHD	63.81	https://es.pcpartpicker.com/	LGA1200	2	5					
//cdn.pcpartpicker.com/	Intel Celeron G5900	2	3.4		58	Intel UHD	66.98	https://es.pcpartpicker.com/	LGA1200	3	5					
//cdn.pcpartpicker.com/	Intel Core i3-10100F	4	3.6	4.3	65		71.4	https://es.pcpartpicker.com/	LGA1200	19	4.8					
//cdn.pcpartpicker.com/	Intel Core i3-10105F	4	3.7	4.4	65		76.97	https://es.pcpartpicker.com/	LGA1200	3	5					
https://m.pcpartpicker.com/	Intel Core i2 Quad Q6600	4	2.4		100		77.94	https://es.pcpartpicker.com/	LGA775	6	4.2					
//cdn.pcpartpicker.com/	AMD Ryzen 5 4500	6	3.6	4.1	65		84.48	https://es.pcpartpicker.com/	AM4	1	5					
//cdn.pcpartpicker.com/	AMD Ryzen 3 3200G	4	3.6	4	65	Radeon Vi	90.98	https://es.pcpartpicker.com/	AM4	64	4.8					
https://m.pcpartpicker.com/	Intel Core i3-8400	2	3		65		91.09	https://es.pcpartpicker.com/	LGA775	4	4.5					
//cdn.pcpartpicker.com/	AMD Athlon 200GE	2	3.2		35	Radeon Vi	94.8	https://es.pcpartpicker.com/	AM4	16	4.8					
https://m.pcpartpicker.com/	Intel Core i3-3250	2	3.5		55	Intel HD G	97.84	https://es.pcpartpicker.com/	LGA1155	1	5					
//cdn.pcpartpicker.com/	AMD FX-9370	8	4.4	4.7	220		99.97	https://es.pcpartpicker.com/	AM3+	9	4.4					

3. Crear instancia en Neo4j aura

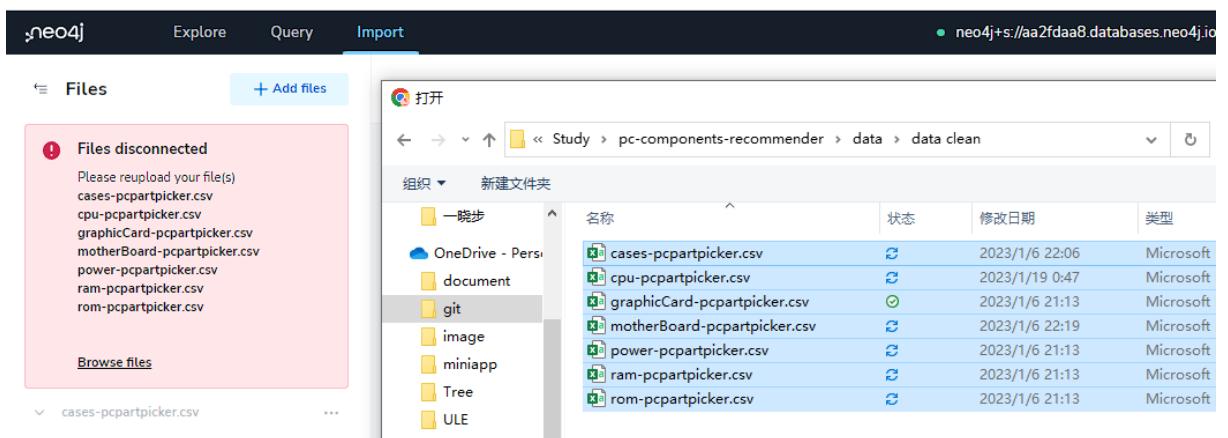
Crear una nueva instancia vacía para construir el propio base de datos, seguir todos los pasos que indica Neo4j aura.

The screenshot shows the Neo4j Aura web interface. On the left sidebar, there are links for 'Instances', 'Connect', 'AuraDS', 'Instances', and 'Getting Started'. The main content area is titled 'Instances' and shows a single instance named 'Instance01' which is currently 'Running'. Below the instance status, it provides details such as 'Neo4j version 5', 'Nodes 1910 / 200000 (1%)', 'Relationships 1257 / 400000 (0%)', and a 'Connection URI' of 'neo4j+s://aa2fdaa8.databases.neo4j.io'. There are also 'Open' and 'Delete' buttons for the instance.

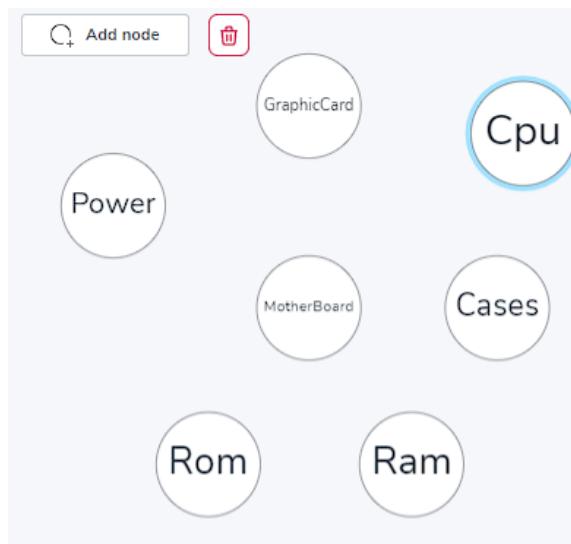
4. Importar los datos

Para importar los datos a Neo4j Aura, hay que seguir los siguientes pasos:

1. Acceder a la instancia con la contraseña ofrecida en la creación de la instancia.
2. En el apartado Import, pulsa el botón “+Add file” y seleccionar los archivos CSV que contiene los datos normalizados.



3. Crear los nodos Cases, Cpu, GraphicCard, MotherBoard, Power, Ram y Rom.



4. Conectar cada nodo a un archivo CSV, mapear todas las propiedades, elegir el tipo de la propiedad si es string, integer, etc. Posteriormente elige la propiedad name como ID del nodo y pulsa el botón “Run Import” para importar los datos al base de datos de Neo4j Aura.

Add files

- tel UH...
- i.81
- tps://es...
- iA1200
-
- :p... ...
- dna.pc...
- ius R...
- ideon ...
-
- IO
- ack / R...
- i9
- i.51
- tps://es...
-
- IC... ...
- dna.pc...

Preview
Run import
...

Add node

```

graph TD
    Cpu((Cpu)) --- Cases((Cases))
    Cases --- Ram((Ram))
    Cases --- Rom((Rom))
    Ram --- MotherBoard((MotherBoard))
    MotherBoard --- GraphicCard((GraphicCard))
    GraphicCard --- Power((Power))
  
```

Mapping details

File ?

cpu-pcpartpicker.csv
 ▼

Filter file

Definition
File mapping

+ Select from file
+ Add Property

name	string	<input type="button" value="edit"/>	<input type="button" value="remove"/>
img_src	string	<input type="button" value="edit"/>	<input type="button" value="remove"/>
core_count	integer	<input type="button" value="edit"/>	<input type="button" value="remove"/>
core_clock_GHz	float	<input type="button" value="edit"/>	<input type="button" value="remove"/>
boost_clock_GHz	float	<input type="button" value="edit"/>	<input type="button" value="remove"/>
tdp_W	integer	<input type="button" value="edit"/>	<input type="button" value="remove"/>
integrated_graphics	string	<input type="button" value="edit"/>	<input type="button" value="remove"/>
price	float	<input type="button" value="edit"/>	<input type="button" value="remove"/>
link	string	<input type="button" value="edit"/>	<input type="button" value="remove"/>
socket	string	<input type="button" value="edit"/>	<input type="button" value="remove"/>
rating_number	integer	<input type="button" value="edit"/>	<input type="button" value="remove"/>
rating	float	<input type="button" value="edit"/>	<input type="button" value="remove"/>

ID ?
File ?

name
 ▼

Import results							
Import completed without errors • Total time: 00:00:09							
cases		cases-pcpartpicker.csv					
Time Taken	File Size	File Rows	Nodes Created	Properties Set	Labels Added	Query Count	Query Time
00:00:01	274.3 KiB	1,130	800	8,918	800	1	00:00:01
cpu		cpu-pcpartpicker.csv					
Time Taken	File Size	File Rows	Nodes Created	Properties Set	Labels Added	Query Count	Query Time
00:00:00	78.5 KiB	346	311	3,759	311	1	00:00:00
graphicCard		graphicCard-pcpartpicker.csv					
Time	File Size	File	Nodes	Properties	Labels	Query	Query

5. Crear nuevos nodos y relaciones

Los nodos creados en el apartado anterior no existen ningún tipo de relaciones, pero para que la aplicación funcione correctamente la compatibilidad entre los productos son muy importantes, en este caso he creado dos nodos:

1. El nodo Socket que relaciona con el nodo Cpu y el nodo MotherBoard, que se utiliza para resolver los problemas de compatibilidad entre placa base y procesador. Se crea con siguientes query en apartado Query

```

neo4j Explore Query Import

$ 

1 MATCH (c:Cpu)
2 MERGE (s:Socket {type:c.socket})
3 MERGE (c)-[:USE_SOCKET]-(s)
,
```

```

1 MATCH (m:MotherBoard)
2 MERGE (s:Socket {type:m.socket})
3 MERGE (m)-[:USE_SOCKET]→(s)

```

2. El nodo Size que relaciona con el nodo MotherBoard y el nodo Cases, que se utiliza para resolver el problema de los tamaños entre placa base y la torre. Se crea con siguientes query:

```

MERGE (:Size {type:'EATX'})
MERGE (:Size {type:'ATX'})
MERGE (:Size {type:'Micro ATX'})
MERGE (:Size {type:'Mini ITX'})

MATCH (m:MotherBoard)
MATCH (s:Size)
WHERE m.size = s.type
MERGE (m)-[:IN_SIZE]->(s)

MATCH (c:Cases)
WHERE c.type IN ['MicroATX Mini ','ATX Mid ','MicroATX Mid ','Mini ITX Desktop','Mini ITX ','HTPC','MicroATX Desktop','ATX Full ','MicroATX Slim','ATX Desktop']
MATCH (s:Size)
WHERE s.type = 'ATX'
MERGE (c)-[:IN_SIZE]->(s)

MATCH (c:Cases)
WHERE c.type IN ['MicroATX Mini ','ATX Mid ','MicroATX Mid ','Mini ITX Desktop','Mini ITX ','MicroATX Desktop','MicroATX Slim']
MATCH (s:Size)
WHERE s.type = 'ATX'
MERGE (c)-[:IN_SIZE]->(s)

MATCH (c:Cases)
WHERE c.type IN ['MicroATX Mini ','MicroATX Mid ','Mini ITX Desktop','Mini ITX ','MicroATX Desktop','MicroATX Slim']
MATCH (s:Size)
WHERE s.type = 'Micro ATX'
MERGE (c)-[:IN_SIZE]->(s)

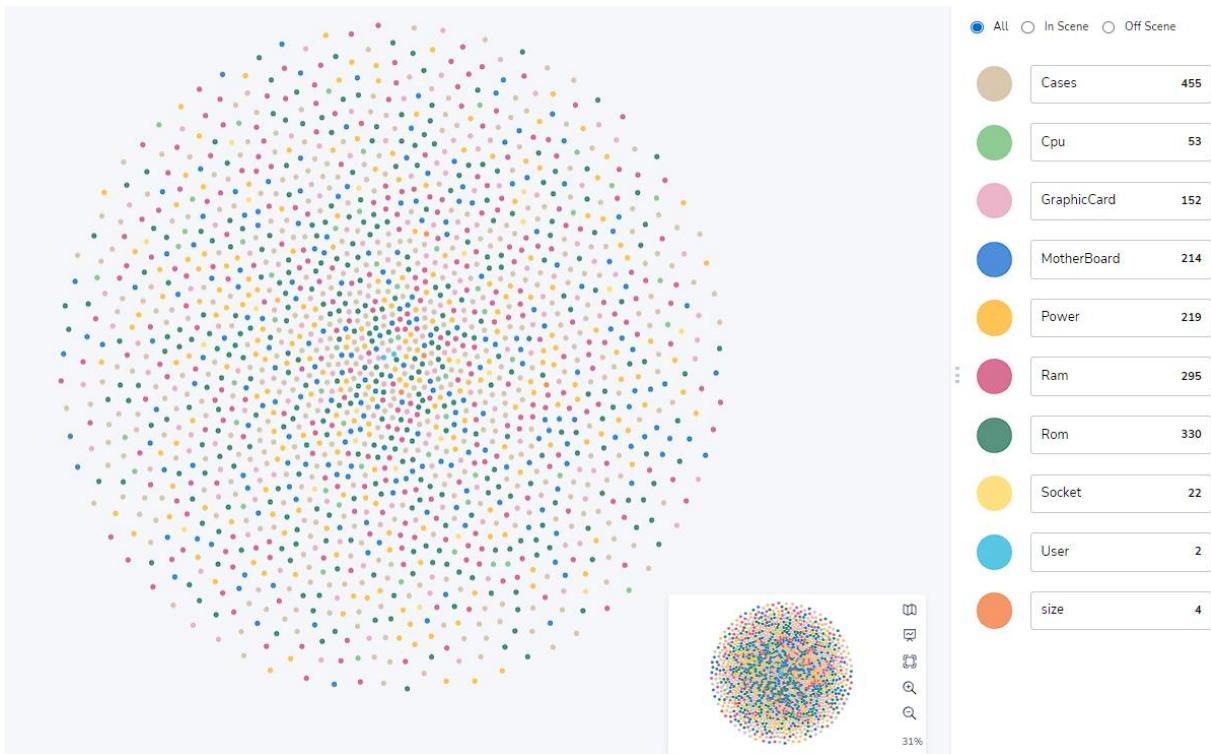
MATCH (c:Cases)
WHERE c.type IN ['Mini ITX Desktop','Mini ITX ']
MATCH (s:Size)
WHERE s.type = 'Mini ITX'
MERGE (c)-[:IN_SIZE]->(s)

```

Ejecutando cada sección de códigos por separado.

Y también está el nodo User que guarda el username, password y la preferencia de uso que se crea que se crea posteriormente cuando el usuario se registra.

6. Visualización de base de datos



Hay 10 tipos nodos, en total son 1910 nodos.



Hay 2 tipos de relaciones, en total son 1257 relaciones y 886 nodos están relacionados unos con otros.

Nodo User: guarda las informaciones del usuario. Tiene 3 propiedades que se puede visualizar en la siguiente imagen.

User 

Delete

Labels

User 

 Add labels

Property	Type	Exclude 
password	String	<input type="checkbox"/>
userType	String	<input type="checkbox"/>
username	String	<input type="checkbox"/>

Nodo Cases: guarda las características de la torre. Tiene 9 propiedades.

Cases 

Delete

Labels

Cases 

 Add labels

Property	Type	Exclude 
color	String	<input type="checkbox"/>
img_src	String	<input type="checkbox"/>
link	String	<input type="checkbox"/>
name	String	<input type="checkbox"/>
price	Float	<input type="checkbox"/>
rating	Float	<input type="checkbox"/>
rating_number	Integer	<input type="checkbox"/>
side_panel	String	<input type="checkbox"/>
type	String	<input type="checkbox"/>

Nodo Cpu: guarda las características del procesador. Tiene 12 propiedades.

Cpu			
Labels			
 Cpu 			
	 Add labels		
Property	Type	Exclude	
boost_clock_GHz	Float	<input type="checkbox"/>	
core_clock_GHz	Float	<input type="checkbox"/>	
core_count	Integer	<input type="checkbox"/>	
img_src	String	<input type="checkbox"/>	
integrated_graphics	String	<input type="checkbox"/>	
link	String	<input type="checkbox"/>	
name	String	<input type="checkbox"/>	
price	Float	<input type="checkbox"/>	
rating	Float	<input type="checkbox"/>	
rating_number	Integer	<input type="checkbox"/>	
socket	String	<input type="checkbox"/>	
tdp_W	Integer	<input type="checkbox"/>	

Nodo GraphicCard: guarda las características de la tarjeta gráfica. Tiene 12 propiedades.

Property	Type	Exclude	
boost_clock_MHz	Integer	<input type="checkbox"/>	
chipset	String	<input type="checkbox"/>	
color	String	<input type="checkbox"/>	
core_clock_MHz	Integer	<input type="checkbox"/>	
img_src	String	<input type="checkbox"/>	
length_mm	Integer	<input type="checkbox"/>	
link	String	<input type="checkbox"/>	
memory_GB	Float	<input type="checkbox"/>	
name	String	<input type="checkbox"/>	
price	Float	<input type="checkbox"/>	
rating	Float	<input type="checkbox"/>	
rating_number	Integer	<input type="checkbox"/>	

Nodo MotherBoard: guarda las características de la placa base. Tiene 11 propiedades.

Property	Type	Exclude ⓘ
color	String	<input type="checkbox"/>
img_src	String	<input type="checkbox"/>
link	String	<input type="checkbox"/>
memory_max_GB	Integer	<input type="checkbox"/>
memory_slots	Integer	<input type="checkbox"/>
name	String	<input type="checkbox"/>
price	Float	<input type="checkbox"/>
rating	Float	<input type="checkbox"/>
rating_number	Integer	<input type="checkbox"/>
size	String	<input type="checkbox"/>
socket	String	<input type="checkbox"/>

Nodo Power: guarda las características de la fuente de alimentación. Tiene 11 propiedades.

Property	Type	Exclude ⓘ
color	String	<input type="checkbox"/>
efficiency	String	<input type="checkbox"/>
img_src	String	<input type="checkbox"/>
link	String	<input type="checkbox"/>
modular	String	<input type="checkbox"/>
name	String	<input type="checkbox"/>
price	Float	<input type="checkbox"/>
rating	Float	<input type="checkbox"/>
rating_number	Integer	<input type="checkbox"/>
type	String	<input type="checkbox"/>
wattage_W	Integer	<input type="checkbox"/>

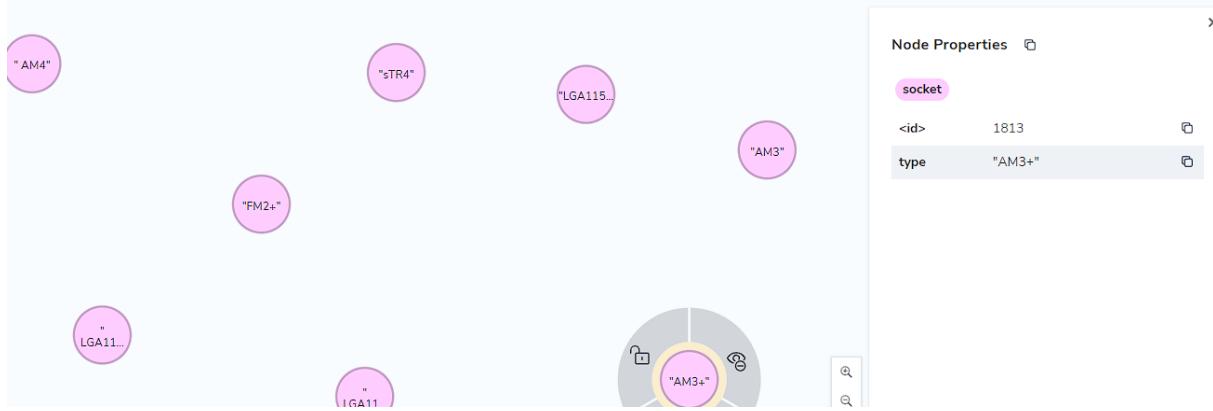
Nodo Ram: guarda las características de la memoria Ram. Tiene 13 propiedades.

Property	Type	Exclude ⓘ
CAS_latency	Integer	<input type="checkbox"/>
color	String	<input type="checkbox"/>
first_word_latency_ns	Float	<input type="checkbox"/>
img_src	String	<input type="checkbox"/>
link	String	<input type="checkbox"/>
modules	String	<input type="checkbox"/>
name	String	<input type="checkbox"/>
price	Float	<input type="checkbox"/>
price_GB	Float	<input type="checkbox"/>
rating	Float	<input type="checkbox"/>
rating_number	Integer	<input type="checkbox"/>
speed_frecuency	Integer	<input type="checkbox"/>
speed_type	String	<input type="checkbox"/>

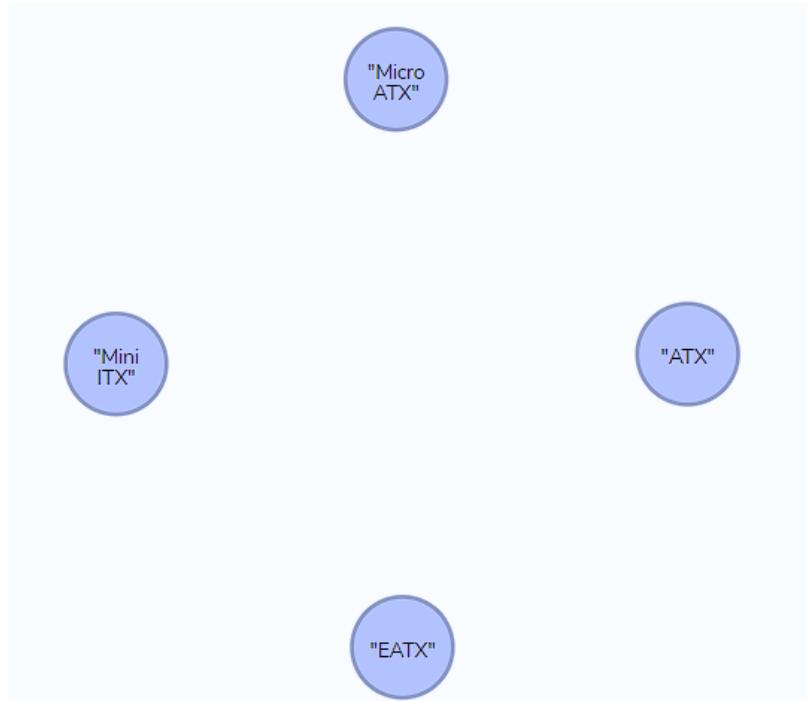
Nodo Rom: guarda las características de la memoria Rom. Tiene 12 propiedades.

Property	Type	Exclude ⓘ
cache	String	<input type="checkbox"/>
capacity	String	<input type="checkbox"/>
form_factor	String	<input type="checkbox"/>
img_src	String	<input type="checkbox"/>
interface	String	<input type="checkbox"/>
link	String	<input type="checkbox"/>
name	String	<input type="checkbox"/>
price	Float	<input type="checkbox"/>
price_GB	Float	<input type="checkbox"/>
rating	Float	<input type="checkbox"/>
rating_number	Integer	<input type="checkbox"/>
type	String	<input type="checkbox"/>

Nodo Socket: Contiene una propiedad que se llama type, y relaciona con el nodo Cpu y el nodo MotherBoard que tienen el mismo Socket. En total existe 22 tipo de socket almacenadas en base de datos.



Nodo Size: Contiene una propiedad type que describe el tamaño, y se relaciona con el nodo Cases y el nodo MotherBoard si ellos pertenecen o si son más pequeños que type. Especifiqué en 4 tamaños distintos.



7. Interfaz de usuario (Frontend)

Frontend es el término que se utiliza para describir la parte de una aplicación o sitio web que el usuario final puede ver e interactuar. Está compuesto por HTML, CSS y JavaScript, y se ejecuta en el navegador web del usuario. El Frontend trabaja en estrecha colaboración con el backend, que es la parte de la aplicación que se ejecuta en el servidor y realiza tareas como la gestión de la base de datos y el procesamiento de datos.

En la parte de Frontend utilice la librería React para crear interfaces de usuario interactivas.

Las hojas de estilos encascada (CSS) he utilizado la librería Flexbox Grid, eso significa que todos los componentes pueden ajustar a cualquier resolución de pantalla sin problema, entonces la aplicación funciona correctamente tanto en escritorio como en dispositivos móviles.

Se utiliza también la librería mui para crear los componentes de manera más rápida y eficaz.

El Frontend está construida por 2 páginas y 6 componentes que son los siguientes:

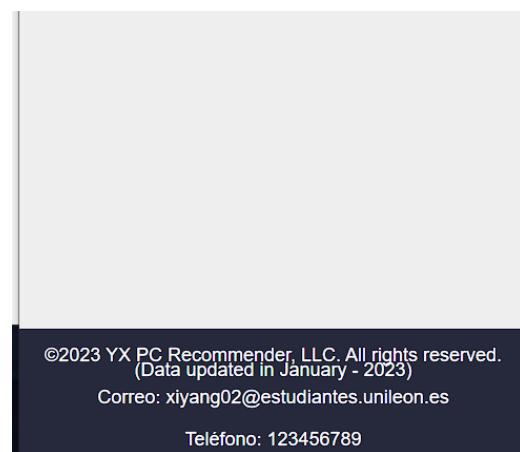
1. Componente Footer:

El footer es la parte inferior de la página, en la que se incluye unas informaciones de la página y datos personales.

a. Vista escritorio



b. Vista móvil



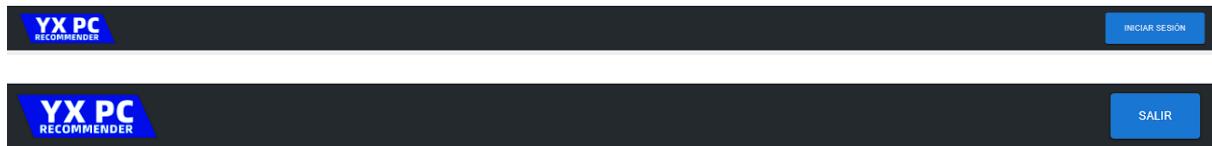
2. Componente Header:

El componente Header es la parte superior de la página, que contiene dos elementos:

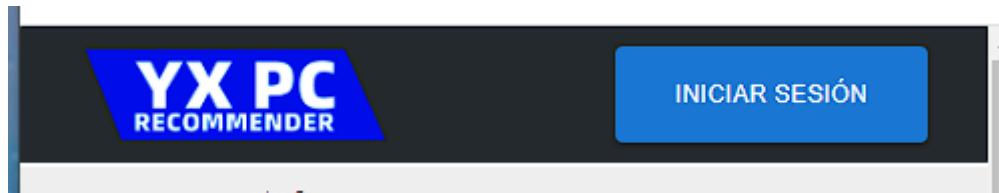
El logo: al pulsar, vuelve a la página home.

Botón de iniciar la sesión/salir la sesión: al pulsar, aparecerá el componente Form, que puede registrar e iniciar la sesión.

a. Vista escritorio



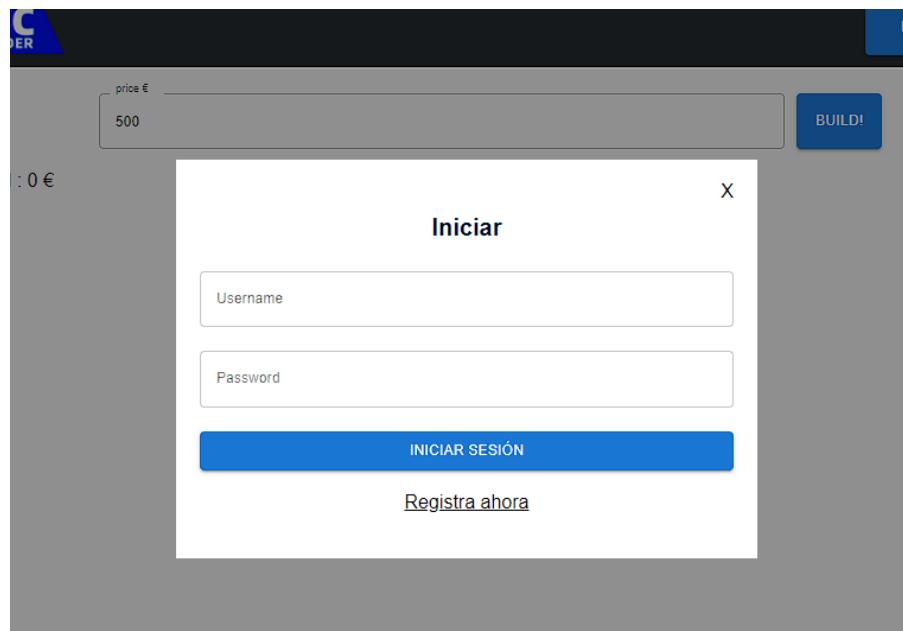
b. Vista móvil



3. Componente Form:

Es el componente que permite el usuario registrar e iniciar la sesión, manda las peticiones al backend e detectar la preferencia de uso del usuario.

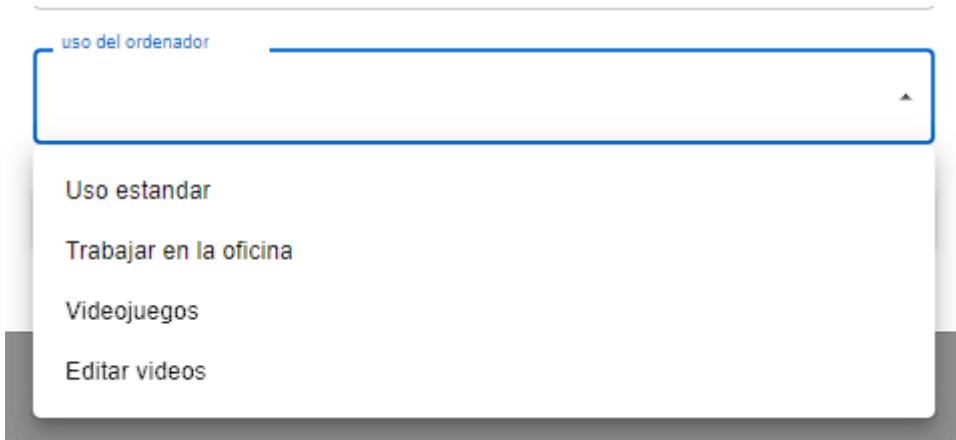
a. Iniciar sesión



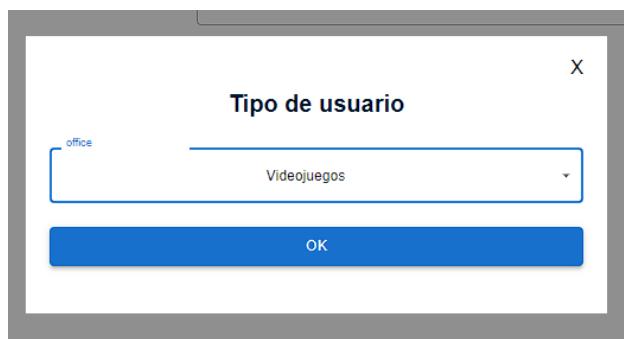
b. Registrar

A screenshot of a registration modal window. The title "Registrar" is at the top center. Below it are three input fields: "Username", "Password", and a dropdown menu with the placeholder "uso del ordenador". At the bottom is a large blue "REGISTRAR" button. In the top right corner of the modal is a small red "X". The background is a dark gray overlay.

Username
Password
uso del ordenador



c. Cambiar la preferencia de uso



4. Componente ProductBox:

Es el componente que se encarga de mostrar las informaciones de los productos.

a. ProductBox en la página ProductList

 <p>Asus TUF GAMING X570-PLUS (WI-FI) ★★★★★ (234) color : Black / Gold size : ATX price : 247.8 memory_max_GB : 128 socket : AM4 memory_slots : 4</p>	 <p>MSI MPG X570 GAMING EDGE W ★★★★★ (86) color : Black / Silver size : ATX price : 224.9 memory_max_GB : 128 socket : AM4 memory_slots : 4</p>
 <p>ASRock Fatal1ty B450 Gaming-ITX/ac ★★★★★ (79) color : Black / Silver size : Mini ITX price : 125.12 memory_max_GB : 64 socket : AM4 memory_slots : 2</p>	 <p>Gigabyte B550I AORUS PRO AX ★★★★★ (75) color : Black / Silver size : Mini ITX price : 213.84 memory_max_GB : 64 socket : AM4 memory_slots : 2</p>

b. ProductBox en Home (cuando recomienda el producto al usuario)

precio total : 429.45 €

ver más

AMD Ryzen 5 5600G

SCORE: 182.98652652165794

★★★★★ (78)

integrated_graphics : Radeon Vega 7

price : 135.39

tdp_W : 65

socket : AM4

boost_clock_GHz : 4.4

core_clock_GHz : 3.9

core_count : 6



ver más

ADATA XPG SPECTRIX D60G 16 GB

El botón ver más llama al método showMore para ver más los productos ordenada por SCORE, por si quiere ver productos similares y cambiar su selección.

The screenshot displays a list of four processor products arranged in a grid:

- AMD Ryzen 5 5600G**
SCORE: 182.98652652165794
★★★★★ (78)
integrated_graphics : Radeon Vega 7
price : 135.39
tdp_W : 65
socket : AM4
boost_clock_GHz : 4.4
core_clock_GHz : 3.9
core_count : 6
- AMD Ryzen 3 3200G**
SCORE: 162.72383189185035
★★★★★ (64)
integrated_graphics : Radeon Vega 8
price : 90.98
tdp_W : 65
socket : AM4
boost_clock_GHz : 4
core_clock_GHz : 3.6
core_count : 4
- Intel Core i3-10100**
SCORE: 157.74993342590635
★★★★★ (38)
integrated_graphics : Intel UHD Graphics 630
price : 110.65
tdp_W : 65
socket : LGA1200
boost_clock_GHz : 4.3
core_clock_GHz : 3.6
core_count : 4
- Intel Pentium G3258**
SCORE: 154.9707700757489
★★★★★ (159)
integrated_graphics : Intel HD Graphics
price : 149.53
tdp_W : 53
socket : LGA1150
core_clock_GHz : 3.2
core_count : 2

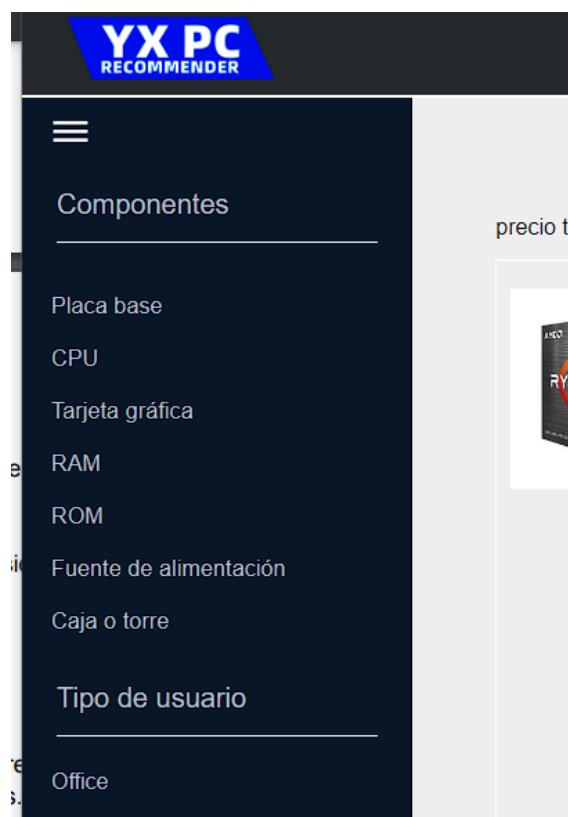
5. Componente Sidebar:

Es la barra lateral que permite al usuario accede a la página ProductList de cada tipo de productos, y también permite cambiar el tipo de usuario si ya está log in. Puede aparecer o no pulsando el icono

a. Desaparece Sidebar



b. Aparece Sidebar



6. Componente withRouter:

Es el único componente funcional y que no muestra por la pantalla, su utilidad es usarlo en otros componentes de clase, para permitir a los componentes de clase pueden utilizar el hook de React, en este caso el useNavigate para cambiar de una página a otra.

```
function withRouter(Component) {
  function ComponentWithRouterProp(props) {
    let location = useLocation();
    let navigate = useNavigate();
    let params = useParams();
    return (
      <Component
        {...props}
        router={{ location, navigate, params }}
      />
    );
  }

  return ComponentWithRouterProp;
}

export default withRouter;
```

7. Página Home:

Está formado por componentes Header, Footer, SideBar, el input de precio y un “botón Build!”, al pulsar el botón, la aplicación manda las peticiones al Backend, encontrar los productos y comparar para obtener los mejores productos para el usuario.

- a. Antes de ejecutar los algoritmos de recomendación



- b. Después de ejecutar los algoritmos de recomendación muestra el componente ProductBox que contiene las informaciones de los productos.

YX PC
RECOMMENDER

SALIR

price €
500

BUILD!

precio total : 429.45 €

ver más

AMD Ryzen 5 5600G

SCORE: 182.98652652165794

★★★★★ (78)

integrated_graphics : Radeon Vega 7

price : 135.39

tdp_W : 65

socket : AM4

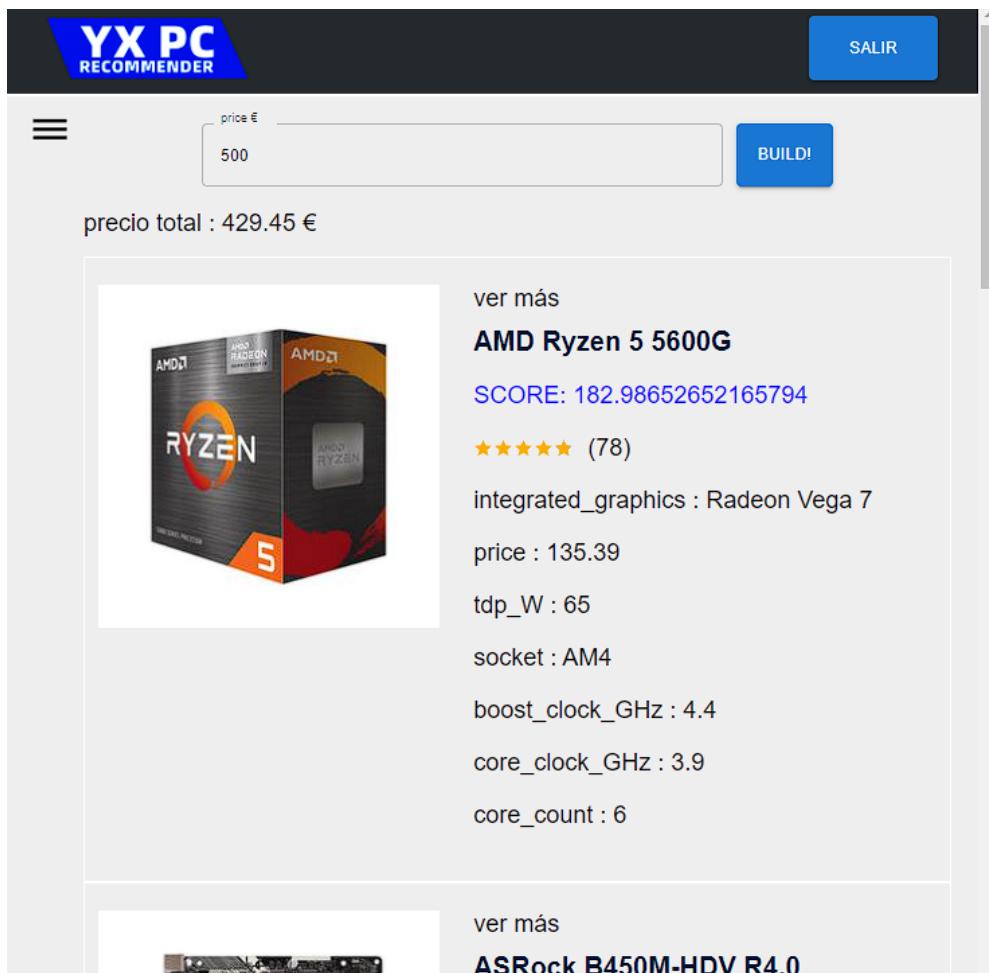
boost_clock_GHz : 4.4

core_clock_GHz : 3.9

core_count : 6

ver más

ASRock B450M-HDV R4.0



8. Página ProductList:

Se accede esta página desde SideBar.

Está formado por componentes Header, Footer, SideBar, ProductBox y la paginación. Cada página contiene como máximo 10 productos.



Tarjeta gráfica

**Gigabyte OC**

★★★★★ (50)

length_mm : 226

memory_GB : 6

boost_clock_MHz : 1830

color : Black / Silver

price : 404.49

chipset : GeForce GTX 1660 SUPER

core_clock_MHz : 1530

**EVGA KO ULTRA GAMING**

★★★★★ (45)

length_mm : 202

memory_GB : 6

boost_clock_MHz : 1755

**Gigabyte AORUS MASTER**

★★★★★ (15)

length_mm : 290

memory_GB : 8

boost_clock_MHz : 1845

color : Black

price : 1071.48

chipset : GeForce RTX 3070

core_clock_MHz : 1500

8. Servidor (Backend)

Para Backend utilice Node.js, que es la que mejor manejo y tiene las siguientes ventajas:

1. Rapidez de desarrollo: Node.js es conocido por su velocidad de desarrollo debido a su naturaleza basada en JavaScript y a la disponibilidad de una gran cantidad de módulos y paquetes de código abierto.
2. Alta escalabilidad: Node.js permite una alta escalabilidad de las aplicaciones gracias a su arquitectura basada en eventos y su capacidad para manejar una gran cantidad de conexiones simultáneas.
3. Comunidad grande y activa: Node.js cuenta con una gran comunidad de desarrolladores que comparte soluciones y recursos, lo que facilita la resolución de problemas y mejora continua de la plataforma.
4. Integración con otras tecnologías: Node.js se integra fácilmente con otras tecnologías, lo que permite un desarrollo más rápido y eficiente.

Para recibir las peticiones generadas por la librería axios de Frontend utilice la librería express, que se escucha las peticiones http en el puerto 5000 o puerto asignado por el variable de entorno.

```
//asignar el puerto que escucha las peticiones
const PORT = process.env.PORT || 5000;
```

Para hacer Query y recibir las respuestas de BBDD de neo4j, se utiliza la librería neo4j driver.

```
const driver = neo4j.driver(uri, neo4j.auth.basic(user, password), { disableLosslessIntegers: true })
```

El disableLossLessIntegers porque en Neo4j, los números en formato float que utiliza 64bits, sin embargo JavaScript utiliza 32 bits por defecto.

En generar mi servidor responde a 4 peticiones que son los siguientes:

1. Recibe un GET, sirve solo para comprobar si el servidor está arrancando

correctamente o no.

```
app.get("/home"), function (req, res) {
  res.send("Servidor activada");
}
```

2. Recibe un HTTP POST, en request pasando en body el parámetro query que es la consulta cypher de Neo4j y al final responde el dato de los productos que devuelve por Neo4j al Frontend.

```
app.post("/api/getProduct", function (request, response) {
  console.log("ejecutando " + request.body.query);
  const query = request.body.query;
  const session = driver.session();
  const resultPromise = session.run(query);
  resultPromise
    .then((result) => {
      const datos = result.records;
      var clean = [];
      datos.forEach((element) => {
        element.forEach((element1) => {
          console.log(element1.properties);
          clean.push(element1.properties);
        });
      });
      response.send(clean);
      session.close();
    })
    .catch((error) => {
      console.log(error);
      session.close();
    });
});
```

Ejemplo de uso en Frontend:

```
findTopSpeedCpu = () => {
  const a = axios.post('/api/getProduct',{
    query: `MATCH (a:Cpu) WHERE a.core_clock_GHz IS NOT NULL return a order by a.core_clock_GHz DESC limit 1`
  })
  .then(function(res){
    return res.data[0];
  })
  .catch(function (error) {
    console.log('query error in frontend');
  });
  return (a);
}
```

3. Recibe un HTTP POST, en request pasando en body el parámetro query que es la consulta cypher de Neo4j y al final devuelve al Frontend.

En mi caso se utiliza para consultar el número total de un tipo de producto, como por ejemplo número total de Cpu, para construir la paginación en la Página de ProductList.

```
app.post("/api/getProductAmount", function (request, response) {
  console.log("ejecutando " + request.body.query);
  const query = request.body.query;
  const session = driver.session();
  const resultPromise = session.run(query);
  resultPromise
    .then((result) => {
      const datos = result.records;
      response.send(datos[0]._fields);
      session.close();
    })
    .catch((error) => {
      console.log(error);
      session.close();
    });
});
```

Ejemplo de uso en Frontend:

```
getProductAmount = () => {
  var that = this;
  axios.post('/api/getProductAmount',{
    query: `MATCH (a:${this.state.productType}) return COUNT(a)`
  })
  .then(function(res){
    if(that.state.productNumber !== res.data[0]){
      that.setState({productNumber: res.data[0]});
    }
  })
  .catch(function (error) {
    console.log("error in frontend: " + error);
  });
}
```

4. Recibe un HTTP POST, en request pasando en body el parámetro query que es la consulta cypher de Neo4j y al final devuelve al Frontend.

En mi caso lo utilizo para devolver los datos del usuario para iniciar la sesión, registrar y consultar el preferencia de uso del usuario.

```

app.post("/api/query", function (request, response) {
  console.log("ejecutando " + request.body.query);
  const query = request.body.query;
  const session = driver.session();
  const resultPromise = session.run(query);
  resultPromise
    .then((result) => {
      const datos = result.records;
      response.send(datos);
      session.close();
    })
    .catch((error) => {
      console.log(error);
      session.close();
    });
});

```

Ejemplo de uso en Frontend:

```

login = () => {
  const that = this;
  this.props.setShowLogin(false);
  this.setState({registerForm: false});
  axios.post('/api/query',{
    query: `MATCH (u:User {username:'${this.props.HeaderState.username}', password:'${this.props.HeaderState.password}'}) return u`
  })
  .then(function(res){
    const user = _get(res,"data[0]()._fields[0].properties",undefined);
    if(user == undefined){
      alert('No estás registrada');
    }else{
      that.props.setLogged(true);
      const userType = _get(user,"userType",'standard')
      that.props.setUserType(userType);
    }
  })
  .catch(function (error) {
    console.log('query error in frontend');
  });
}

```

9. Algoritmos de recomendación

La Aplicación usa el **algoritmo híbrido** que combina diferentes técnicas de recomendación como los siguientes:

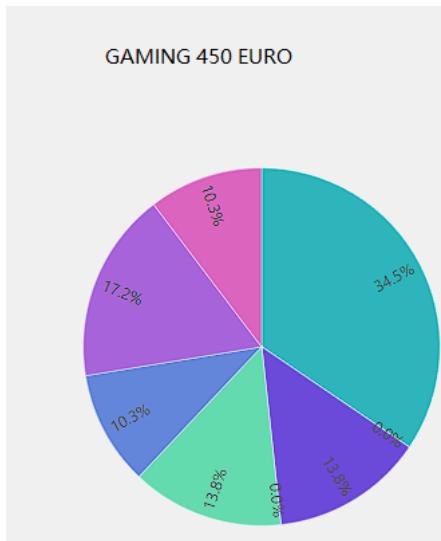
1. **Recomendación basada en contenido:** La aplicación compara las características del producto con los gustos y preferencias del usuario para recomendar el mejor producto, que calcula la puntuación de los productos y elige el mejor.
2. **Filtrado colaborativo:** utiliza las valoraciones históricas del producto para identificar patrones en las preferencias de los usuarios y recomendar productos en función de preferencia de uso del ordenador del usuario.

A continuación explicaré de forma más detallada del algoritmo:

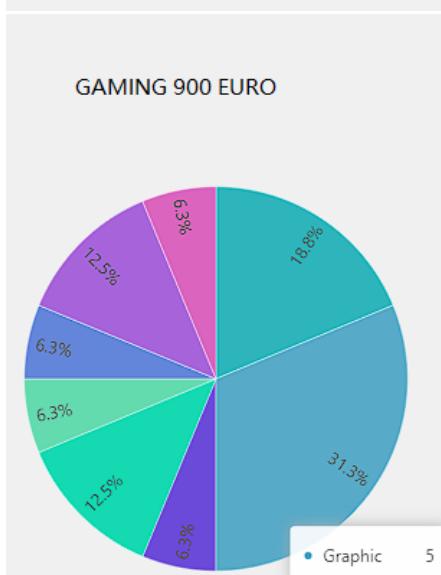
A la hora de recomendar los componentes de ordenador, al principio pensaba que lo bueno o lo malo de una misma categoría de productos como por ejemplo CPU viene determinado por el tipo de usuario, pero posteriormente me di cuenta de que no era así, independientemente del tipo de usuario, si un componente es bueno, es bueno para cualquier tipo de usuario, la mayor dificultad está en cómo debería repartir el presupuesto y cómo elegir el mejor producto con su presupuesto para este componente.

El algoritmo se concentra en dos enfoques:

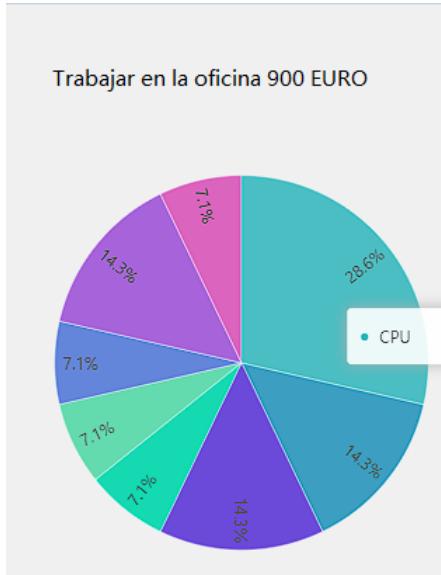
1. **Algoritmo de repartición de presupuestos,** la idea principal fue buscar las compras históricas de los usuarios, sabiendo su preferencia de uso diario, el presupuesto total y el porcentaje medio de su inversión para cada categoría del producto, y entonces puede crear un algoritmo que calcula el porcentaje medio sabiendo su presupuesto para esta compra, pero no tengo los datos históricos de otros usuarios entonces ajusté manualmente esa repartición con mis conocimientos. El algoritmo no está implementado por falta de datos.



	B	C	D	E	F	G	H	I	Cases
1	CPU	Graphic	Ram	HDD	SSD	Power	MotherBoard		
2	5	0	2	0	2	1.5	2.5		1.5
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									



	B	C	D	E	F	G	H	I	Cases
1	CPU	Graphic	Ram	HDD	SSD	Power	MotherBoard		
2	3	5	1	2	1	1	1	2	1
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									



	B	C	D	E	F	G	H	I	MotherBoard
1	CPU	Graphic	Ram	HDD	SSD	Power			
2	4	2	2	1	1	1	1	2	2
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									

Varían tanto por el presupuesto total como la preferencia de uso del

ordenador.

Como no tengo datos suficientes para implementar ese algoritmo, hice una partición manualmente por mi experiencia personal que se puede ver en los diagramas de arriba. Y aquí dejo la partición de presupuesto completa que está usando la aplicación actualmente.

```
if(this.state.price <= 450){
    this.setState({pricePart: {cpu: 5, graphic: 0, ram: 2, ssd: 2, hdd: 0, power: 1.5, motherBoard: 2.5, cases: 1.5, total: 14.5}});
} else {
    switch(this.props.AppState.userType){ //standard || office || gaming || editVideo
        case 'standard':
            if(this.state.price <= 800){
                this.setState({pricePart: {cpu: 4, graphic: 0, ram: 2, ssd: 1, hdd: 1, power: 1, motherBoard: 2, cases: 1, total: 12}});
            } else {
                this.setState({pricePart: {cpu: 4, graphic: 3, ram: 2, ssd: 1.5, hdd: 0, power: 1, motherBoard: 2.5, cases: 1, total: 15}});
            }
            break;
        case 'office':
            if(this.state.price <= 800){
                this.setState({pricePart: {cpu: 4, graphic: 0, ram: 2, ssd: 1, hdd: 1, power: 1, motherBoard: 2, cases: 1, total: 12}});
            } else {
                this.setState({pricePart: {cpu: 4, graphic: 2, ram: 2, ssd: 1, hdd: 1, power: 1, motherBoard: 2, cases: 1, total: 14}});
            }
            break;
        case 'gaming':
            if(this.state.price <= 800){
                this.setState({pricePart: {cpu: 4, graphic: 4, ram: 1.5, ssd: 1.5, hdd: 0, power: 1, motherBoard: 2, cases: 1, total: 15}});
            } else {
                this.setState({pricePart: {cpu: 3, graphic: 5, ram: 1, ssd: 2, hdd: 1, power: 1, motherBoard: 2, cases: 1, total: 16}});
            }
            break;
        case 'editVideo':
            if(this.state.price <= 800){
                this.setState({pricePart: {cpu: 4, graphic: 4, ram: 1, ssd: 1, hdd: 1, power: 1, motherBoard: 2, cases: 1, total: 15}});
            } else {
                this.setState({pricePart: {cpu: 3, graphic: 5, ram: 1, ssd: 1.5, hdd: 1.5, power: 1, motherBoard: 2, cases: 1, total: 16}});
            }
            break;
    }
}
```

Porcentaje se calcula el número asignado a la categoría dividiendo entre total.

Por ejemplo Presupuesto: 450 euro, CPU:4, TOTAL: 12.

Precio que gasta en CPU es: $450 * 4 / 12 = 150$ euros

```
let lista = null;
await axios.post('/api/getProduct', {
    query: `MATCH (a:GraphicCard) WHERE a.price < ${this.state.price*this.state.pricePart.graphic>this.state.pricePart.total} return a order by a.core_clock_GHz DESC limit 50`
})
.then(function(res){
```

En algunos casos podéis ver que Graphic: 0.

Este caso quiere decir que el sistema recomienda comprar un CPU que tiene la gráfica integrada, porque un GPU < 80 euros tienen rendimiento peor que un CPU con gráfica integrada.

```
let query = '';
if(this.state.pricePart.graphic === 0){
    query = `MATCH (a:Cpu) WHERE a.price < ${this.state.price*this.state.pricePart.cpu>this.state.pricePart.total}
    AND a.integrated_graphics > '' return a order by a.core_clock_GHz DESC limit 50`;
} else{
    query = `MATCH (a:Cpu) WHERE a.price < ${this.state.price*this.state.pricePart.cpu>this.state.pricePart.total} return a order by a.core_clock_GHz DESC limit 50`;
}
await axios.post('/api/getProduct', {
```

2. **Algoritmo de Scoring**, es el algoritmo que calcula la puntuación del componente con sus características, número de valoraciones y su valoración. Son diferentes para cada tipo porque sus propiedades son diferentes y tiene diferente comportamiento, pero a su vez muy parecidas. El algoritmo tiene dos pasos, primer paso es calcular SCORE_BASE, que se obtiene por sus características y después multiplicando por INCREMENTO, que se obtiene por sus valoraciones:

A continuación voy calculando SCORE_BASE de cada categoría:

- a. Cpu, para calcular SCORE_BASE del CPU elegí 3 propiedades que son: Frecuencia base, Frecuencia overClocking y número de núcleos.

$$\text{SCORE_BASE} = \left(\frac{\text{frecuencia}}{\text{frecuencia del procesador más alta}} + \frac{\text{frecuencia overClocking}}{\text{frecuencia del procesador más alta}} * 0.5 + \frac{\text{numero nucleo}}{\text{máx num nucleo}} * 0.5 \right) * 100$$

Las propiedades pueden tener diferentes importancias, por existe ese multiplica por 0.5, en este caso la valoración SCORE_BASE está acotado entre 0 y 200

```
setCpuListByScore = async() => {
  try{
    const that = this;
    const topSpeedCpu = await this.findTopSpeedCpu();
    const topBoostSpeedCpu = await this.findTopBoostSpeedCpu();
    const mostCoreCpu = await this.findMostCoreCpu();

    const topSpeed = topSpeedCpu.core_clock_GHz;
    const mostCore = mostCoreCpu.core_count;
    const topBoostSpeed = topBoostSpeedCpu.boost_clock_GHz;
    let speed = 0;
    let boostSpeed = 0;
    let core_count = 0;
    let scoreBase = 0;
    let incremento = 0;
    let lista = null;
    let query = '';

    if(this.state.pricePart.graphic === 0){
      query = `MATCH (a:Cpu) WHERE a.price < ${this.state.price*this.state.pricePart.cpu>this.state.pricePart.total}
      AND a.integrated_graphics > '' return a order by a.core_clock_GHz DESC limit 50`;
    }else{
      query = `MATCH (a:Cpu) WHERE a.price < ${this.state.price*this.state.pricePart.cpu>this.state.pricePart.total} return a order by a.core_clock_GHz DESC limit 50`;
    }
    await axios.post('/api/getProduct',{
      query: query
    })
    .then(function(res){
      for(let i = 0; i< res.data.length; i++){
        speed = res.data[i].core_clock_GHz;
        core_count = res.data[i].core_count;
        boostSpeed = _get(res.data[i], 'boost_clock_GHz', speed);
        //calcular score
        scoreBase = ((speed/topSpeed + boostSpeed/topBoostSpeed*0.5 + core_count/mostCore*0.5) *100);
        if(res.data[i].rating_number > 200){
          incremento = (Math.pow(200, 1/Math.E)/10) * (res.data[i].rating*2/5-1);
        }else {
          incremento = (Math.pow(res.data[i].rating_number, 1/Math.E)/10) * (res.data[i].rating*2/5-1);
        }
        res.data[i].score = scoreBase + scoreBase*incremento;
      }
      lista = _orderBy(res.data,['score'], ['desc']);
      that.setState({socket: lista[0].socket});
      that.setState({cpulist: lista});
    })
  }
}
```

- b. Gráfica, para calcular SCORE_BASE del CPU elegí 3 propiedades que son:
 Frecuencia base, Frecuencia overClocking y tamaño de memoria.
 Que la frecuencia base y el tamaño de memoria tiene mayor peso

$$\text{SCORE_BASE} = \left(\frac{\text{frecuencia}}{\text{frecuencia del procesador max}} + \frac{\text{frecuencia overClocking}}{\text{frecuencia del procesador max}} * 0.5 + \frac{\text{memoria}}{\text{memoria max}} * 0.8 \right) * 100$$

En este caso la valoración SCORE_BASE está acotado entre 0 y 230

```
.then(function(res){
  for(let i = 0; i < res.data.length; i++){
    speed = _get(res.data[i], 'core_clock_GHz', 0);
    boostSpeed = _get(res.data[i], 'boost_clock_MHz', speed);
    memory = _get(res.data[i], 'memory_GB', 4);
    //calcular score
    scoreBase = ((speed/topSpeed + boostSpeed/topBoostSpeed*0.5 + memory/mostMemory*0.8) *100);
```

- c. RAM, SCORE_BASE de la memoria RAM está afectados por las propiedades: memoria, precio/memoria y frecuencia

$$\text{SCORE_BASE} = \left(\frac{\text{memoria}}{\text{memoria max}} + \frac{\text{mejor precio por memoria}}{\text{precio por memoria}} * 0.1 + \frac{\text{frecuencia}}{\text{frecuencia}} * 0.1 \right) * 100$$

Siendo precio/memoria el precio que necesita para obtener 1 GB de Ram.

$$\frac{\text{mejor precio por memoria}}{\text{precio por memoria}}$$

Por eso en este caso mejor precio por memoria a diferencia con los otros, está invertido el orden de la división, el que tiene valor más baja es el mejor.

Las dos propiedades que multiplican por 0.1, sea un valor relativamente baja porque memoria/meroria_max normalmente es muy baja. 16GB/256GB no llega ni a 0.1 el valor. Y es la capacidad más utilizada hoy en día.



ADATA XPG SPECTRIX D60G 16 GB
 SCORE: 42.84468013362741
 ★★★★★ (24)
 speed_type : DDR4
 price_GB : 4.654
 first_word_latency_ns : 10
 CAS_latency : 16
 modules : 2 x 8GB
 speed_frequency : 3200
 price : 74.47



TEAMGROUP T-Force Vulcan TUF Gaming Alliance 16 GB
 SCORE: 37.15796444751744
 ★★★★★ (12)
 speed_type : DDR4
 color : Gold / Black
 price_GB : 4.91
 first_word_latency_ns : 10
 CAS_latency : 16
 modules : 2 x 8GB
 speed_frequency : 3200
 price : 78.56

Ram tiene SCORE relativamente baja, la mayoría no llega a 50, aunque el valor máximo de SCORE_BASE puede llegar hasta 120, (SCORE puede tener valor mucho más alto de SCORE_BASE por el INCREMENTO)

d. ROM, La comparación se divide en dos bloques SSD y HDD porque sus tecnologías son distintas, que SSD cuesta mucho más que el HDD, en misma capacidad, puede tener el precio triple. Entonces se compara SSD con SSD, y HDD con HDD.

Las propiedades que afectan al cálculo de SCORE son los siguientes:
Capacidad y precio/capacidad (precio/GB)

$$\text{SCORE_BASE} = \left(\frac{\text{capacidad}}{\text{numero TB}} + \frac{\text{mejor precio por memoria}}{\text{precio por memoria}} * 0.1 \right) * 100$$

Siendo número TB = 1000GB, quiere decir que consigue cada 100 punto por TB que tiene, entonces no puede fijar una cota máxima del SCORE.

La mayoría de los ordenadores no tienen un ROM de 1TB.

```
await axios.post('/api/getProduct',{
  query: `MATCH (a:Rom) WHERE a.price < ${this.state.price*this.state.pricePart.ssd>this.state.pricePa
})
.then(function(res){
  for(let i = 0; i< res.data.length; i++){
    priceMemory = _get(res.data[i], 'price_GB', 0);
    price = _get(res.data[i], 'price');
    capacity = price/priceMemory;
    //calcular score
    scoreBase = ((capacity/numberTB + bestPriceMemory/priceMemory*0.1) *100);
```

e. Fuente de alimentación, Las propiedades que afectan al cálculo de SCORE son los siguientes:

Voltaje, eficiencia y modularidad. SCORE_BASE es igual a

$$\frac{\text{VOLTAJE}}{\text{VOLTAJE MAX}} * 100 * \text{EFICIENCIA} * \text{MODULARIDAD}$$

El voltaje puede ser de 3 tipos (GOLD, SILVER Y BRONCE), si es GOLD, multiplica por 1.5, si es SILVER, multiplica por 1.25.

El modularidad puede ser de 3 tipos también (FULL, SEMI, NO), si es FULL, multiplica por 1.5, si es SEMI, multiplica por 1.25.

SCORE_BASE teóricamente está entre 0 y 225.

- f. Torre y la placa base son dos casos más especial, porque por un lado es difícil recomendar la torre para montar el ordenador, depende sobre todo por el gusto personal, pero lógicamente el más caro tiene mejor material e iluminación, y por otro lado para seleccionar la placa base lo más importante es la compatibilidad con el CPU, y eso ya se lo comprueba automáticamente por el sistema, y con la misma razón que la torre, la más cara normalmente aporta mejor compatibilidad con todos otros componentes.
entonces el SCORE_BASE de la torre y la placa base = Price*0.5
Afecta sobre todo por el INCREMENTO que viene dada por las valoraciones de los usuarios.
Y no se puede acotar SCORE_BASE máximo.

Por la diversidad que tienen la forma de calcular SCORE_BASE, su valor solo tiene sentido comparar con los componentes de la misma categoría.

Ahora vemos cómo se calcula el SCORE utilizando SCORE_BASE.

Para elegir el mejor producto no solo es importante sus características escritas, sino también depende mucho de su marca, material, belleza, etc.

Pero esos son aspectos incomparables entonces me resulta muy importante las valoraciones de los usuarios.

SCORE = SCORE_BASE + SCORE_BASE*INCREMENTO.

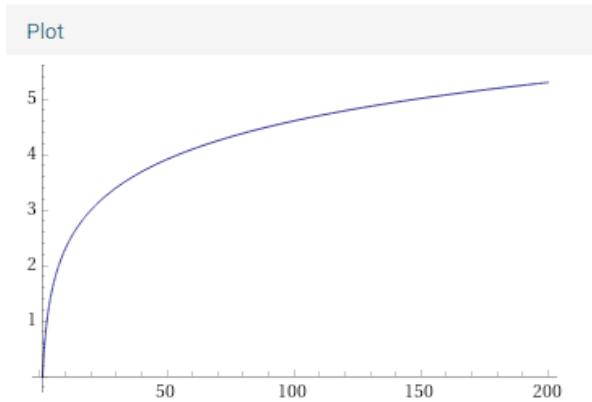
El INCREMENTO puede ser positive o negativo dependiendo de las valoraciones que se recibe el producto.

```
res.data[i].score = scoreBase + scoreBase*incremento;
```

Para el cálculo de INCREMENTO busqué algunas funciones crecientes cóncavas y al final me resulta útil esos dos:

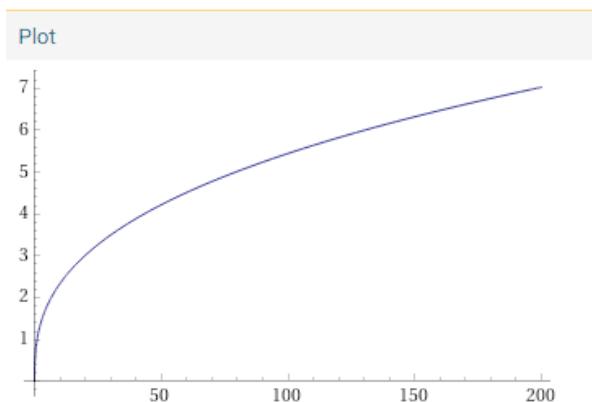
1. Función ln(x)

```
plot log(x) x = 1 to 200
```



2. Función raíz e de X

```
plot sqrt(e^x) x = 0 to 200
```



Busco este tipo de funciones porque me parece que ajustan a la importancia de las valoraciones, siendo X número de las valoraciones, mayor número, mayor importancia tiene su valoración medio, que las valoraciones son más fiables.

Y además si dos productos uno tiene 100 número de valoraciones, otro 50 número de valoraciones y ambos tienen la valoración media 5 (el valor máximo), no quiere decir que tiene una diferencia doble en los gustos de los usuarios, pero el que tenga 100 sí es cierto que han sido gustado por más personas, por eso la función tiene que ser creciente y cóncava.

Al final elegí la **función raíz e de X** para el cálculo de incremento porque su gráfica decrece de forma más suave y a mí personalmente me gusta más.

Para el cálculo del INCREMENTO utiliza 3 variables:

$$\text{INCREMENTO} = \sqrt[n]{\text{numero de valoraciones}} * \frac{1}{\alpha} * \text{valoracion_media_normalizada}$$

La primera ya mencionada anteriormente.

1/alfa, siendo alfa un constante para controlar la importancia de las valoraciones, por ejemplo para comprar una torre/caja, las valoraciones tiene más peso que a la hora de comprar un CPU, a la compra CPU, la valoración afecta, pero no tanto como la torre, que sus características tienen más importancias.

El valor alfa elegida para CPU: 10, GPU: 10, RAM: 2, ROM: 5, Fuente de alimentación: 5, Torre/Caja: 2 y Placa base: 2.

Mayor sea la alfa, menor incremento, menor importancia de las valoraciones. Y viceversa.

Valoración media normalizada, la valoración media antes de normalizar tiene un valor entre 0 y 5, la idea es cambiar el rango entre -1 a 1, de esta manera si la valoración es muy baja, INCREMENTO es negativo y baja el SCORE, y si es positiva aumenta el SCORE

$$\text{valoración normalizada} = \text{valoracion} * \frac{2}{5} - 1$$

SCORE final se calcula con la siguiente manera:

$$\text{SCORE} = \text{SCORE_BASE} + \text{SCORE_BASE} * \text{INCREMENTO}$$

10. Análisis de resultados

La aplicación existen 4 tipo de usuario, distinguida por preferencia de uso del ordenador, que son: estándar, oficina, video juegos y editar videos.

El usuario no logueado se considera como usuario con preferencia de uso estándar.

Si el presupuesto total es menor que 450, se comporta todo de misma manera, porque hasta ese momento es mejor conseguir un ordenador con rendimiento equilibrio, que no tiene cantidad suficiente para comprar una tarjeta gráfica ni para tener dos ROM que combina SSD y HDD.

En esta sección voy a analizar el resultado de todos los tipos de usuario.

1. Presupuesto 300: mismo para todos los tipos de usuario

 ver más AMD Ryzen 3 3200G SCORE: 162.72383189185035 ★★★★★ (64) integrated_graphics : Radeon Vega 8 price : 90.98 tdp_W : 65 socket : AM4 boost_clock_GHz : 4 core_clock_GHz : 3.6 core_count : 4	 ver más Gigabyte GA-A320M-S2H SCORE: 61.355750452503955 ★★★★☆ (24) color : Black size : Micro ATX price : 66.8 memory_max_GB : 32 socket : AM4 memory_slots : 2
 ver más Corsair CMV8GX4M1A2133C15 8 GB SCORE: 28.366190682688625 ★★★★★ (8) speed_type : DDR4 color : Black price_GB : 3.375 first_word_latency_ns : 14.065 CAS_latency : 15 modules : 1 x 8GB speed_frequency : 2133	 ver más KIOXIA EXCERIA SCORE: 68.03835616438357 ★★★★★ (1) price : 34.99 price_GB : 0.073 interface : SATA 6.0 Gb/s type : SSD form_factor : 2.5" capacity : 480 GB
 ver más Xilence XP400R6 SCORE: 21.333333333333332 ★★☆☆☆ (1) wattage_W : 400 modular : No color : Black / Red price : 35.04 type : ATX	 ver más Deepcool MATREXX 30 SCORE: 25.03047898866016 ★★★★☆ (21) side_panel : Tempered Glass color : Black price : 29.9 type : MicroATX Mini

El sistema ha sido capaz de gastar 284 euros de 300, que cubre todos los componentes necesarios para montar un ordenador, como no tiene

suficiente presupuesto para comprar una tarjeta gráfica, ha elegido un CPU que tiene la gráfica integrada. Ha sido capaz de resolver los problemas del tamaño y la compatibilidad entre los componentes.

2. Presupuesto 250: mismo para todos los tipos de usuario

precio total : 159.75 €	
	ver más Intel Pentium Gold G6400 SCORE: 136.78571151406862 integrated_graphics : Intel UHD Graphics 610 price : 63.81 tdp_W : 58 socket : LGA1200 core_clock_GHz : 4 core_count : 2
	ver más Corsair CMV8GX4M1A2133C15 8 GB SCORE: 28.366190682688625 speed_type : DDR4 color : Black price_GB : 3.375 first_word_latency_ns : 14.065 CAS_latency : 15 modules : 1 x 8GB speed_frequency : 2133 price : 27
	ver más Crucial MX500 SCORE: 52.79370579512451 cache : 256 MB price : 33.9 price_GB : 0.136 interface : SATA 6.0 Gb/s type : SSD form_factor : 2.5" capacity : 250 GB
	ver más Xilence XP400R6 SCORE: 21.333333333333332 wattage_W : 400 modular : No color : Black / Red price : 35.04 type : ATX

El sistema solo es capaz de encontrar 4 componentes, que falta la placa base y la torre/caja para montar un ordenador, en este caso no tiene el presupuesto suficiente para montar un ordenador para uso.

3. Usuario standard presupuesto 500:

precio total : 429.45 €	
	ver más AMD Ryzen 5 5600G SCORE: 182.98652652165794 integrated_graphics : Radeon Vega 7 price : 135.39 tdp_W : 65 socket : AM4 boost_clock_GHz : 4.4 core_clock_GHz : 3.9 core_count : 6
	ver más ASRock B450M-HDV R4.0 SCORE: 73.39033927337525 color : Black / White size : Micro ATX price : 75.62 memory_max_GB : 32 socket : AM4 memory_slots : 2
	ver más ADATA XPG SPECTRIX D60G 16 GB SCORE: 42.84468013362741 speed_type : DDR4 price_GB : 4.654 first_word_latency_ns : 10 CAS_latency : 16 modules : 2 x 8GB speed_frequency : 3200 price : 74.47
	ver más KIOXIA EXCERIA SCORE: 68.03835616438357 price : 34.99 price_GB : 0.073 interface : SATA 6.0 Gb/s type : SSD form_factor : 2.5" capacity : 480 GB

	<p>ver más</p> <p>Seagate BarraCuda</p> <p>SCORE: 95 70534688445295</p> <p>★★★★★ (111)</p> <p>cache : 16 MB price : 39.9 price_GB : 0.08 interface : SATA 6.0 Gb/s type : 7200 RPM form_factor : 3.5" capacity : 500 GB</p>		<p>ver más</p> <p>Xilence XP400R6</p> <p>SCORE: 21.333333333333332</p> <p>★★★★★ (1)</p> <p>wattage_W : 400 modular : No color : Black / Red price : 35.04 type : ATX</p>
	<p>ver más</p> <p>Lian Li LanCool PC-K57</p> <p>SCORE: 25.53</p> <p>★★★★★ (1)</p> <p>color : Black price : 34.04 type : ATX Mid</p>		

El sistema recomienda no comprar tarjeta gráfica, sino comprar CPU que contiene la tarjeta integrada, comprar dos memorias ROM, una SSD que puede instalar el sistema operativo y juegos con velocidad rápida y comprar otra HDD que puede almacenar otras cosas.

Si miramos la lista de los CPUs que pueden comprar con su presupuesto, es verdad que AMD Ryzen 5 5600G es mejor que Ryzen3 3200G.

	<p>AMD Ryzen 5 5600G</p> <p>SCORE: 182.98652652165794</p> <p>★★★★★ (78)</p> <p>integrated_graphics : Radeon Vega 7 price : 135.39 tdp_W : 65 socket : AM4 boost_clock_GHz : 4.4 core_clock_GHz : 3.9 core_count : 6</p>		<p>AMD Ryzen 3 3200G</p> <p>SCORE: 162.72383189185035</p> <p>★★★★★ (64)</p> <p>integrated_graphics : Radeon Vega 8 price : 90.98 tdp_W : 65 socket : AM4 boost_clock_GHz : 4 core_clock_GHz : 3.6 core_count : 4</p>
	<p>Intel Core i3-10100</p> <p>SCORE: 157.74993342590835</p> <p>★★★★★ (38)</p> <p>integrated_graphics : Intel UHD Graphics 630 price : 110.65 tdp_W : 65 socket : LGA1200 boost_clock_GHz : 4.2</p>		<p>Intel Pentium G3258</p> <p>SCORE: 154.9707700757489</p> <p>★★★★★ (159)</p> <p>integrated_graphics : Intel HD Graphics price : 149.53 tdp_W : 53 socket : LGA1150 core_clock_GHz : 3.2</p>

4. Usuario gaming presupuesto 500:

precio total : 438.69 €

	<p>ver más</p> <p>AMD Ryzen 3 3200G</p> <p>SCORE: 162.72383189185035</p> <p>★★★★★ (64)</p> <p>integrated_graphics : Radeon Vega 8</p> <p>price : 90.98</p> <p>tdp_W : 65</p> <p>socket : AM4</p> <p>boost_clock_GHz : 4</p> <p>core_clock_GHz : 3.6</p> <p>core_count : 4</p>		<p>ver más</p> <p>Gigabyte GV-N1030OC-2GI</p> <p>SCORE: 43.423784631156586</p> <p>★★★★★ (8)</p> <p>length_mm : 168</p> <p>memory_GB : 2</p> <p>boost_clock_MHz : 1544</p> <p>color : Black / Orange</p> <p>price : 120</p> <p>chipset : GeForce GT 1030</p> <p>core_clock_MHz : 1290</p>
	<p>ver más</p> <p>Gigabyte GA-A320M-S2H</p> <p>SCORE: 61.355750452503955</p> <p>★★★★☆ (24)</p> <p>color : Black</p> <p>size : Micro ATX</p> <p>price : 66.8</p> <p>memory_max_GB : 32</p> <p>socket : AM4</p> <p>memory_slots : 2</p>		<p>ver más</p> <p>Patriot Viper 3 16 GB</p> <p>SCORE: 35.67259661996955</p> <p>★★★★★ (9)</p> <p>speed_type : DDR3</p> <p>color : Black / Silver</p> <p>price_GB : 2.874</p> <p>first_word_latency_ns : 10.718</p> <p>CAS_latency : 10</p> <p>modules : 2 x 8GB</p> <p>speed_frequency : 1866</p> <p>price : 45.99</p>
	<p>ver más</p> <p>SanDisk SSD PLUS</p> <p>SCORE: 84.44974822275783</p> <p>★★★★★ (23)</p> <p>price : 49.98</p> <p>price_GB : 0.104</p> <p>interface : SATA 6.0 Gb/s</p> <p>type : SSD</p> <p>form_factor : 2.5"</p> <p>capacity : 480 GB</p>		<p>ver más</p> <p>Xilence XP400R6</p> <p>SCORE: 21.333333333333332</p> <p>★★☆☆☆ (1)</p> <p>wattage_W : 400</p> <p>modular : No</p> <p>color : Black / Red</p> <p>price : 35.04</p> <p>type : ATX</p>
	<p>ver más</p> <p>Deepcool MATREXX 30</p> <p>SCORE: 25.03047898866016</p> <p>★★★★☆ (21)</p> <p>side_panel : Tempered Glass</p> <p>color : Black</p> <p>price : 29.9</p> <p>type : MicroATX Mini</p>		

Gasta 438 sobre 500.

Comparando con el usuario estándar, para el usuario gaming, el sistema recomienda comprar una gráfica y gastar menos en otros componentes, porque considera que GPU tiene mayor preferencia.

	Gigabyte GV-N1030OC-2GI SCORE: 43.423784631156586 ★★★★★ (8) length_mm : 168 memory_GB : 2 boost_clock_MHz : 1544 color : Black / Orange price : 120 chipset : GeForce GT 1030 core_clock_MHz : 1290		Gigabyte GV-N1030SL-2GL SCORE: 43.28112531916923 ★★★★★ (5) length_mm : 175 memory_GB : 2 boost_clock_MHz : 1506 color : White price : 107.02 chipset : GeForce GT 1030 core_clock_MHz : 1252
	Asus Phoenix Fan OC SCORE: 42.780848884302 ★★★★★ (4) length_mm : 184 memory_GB : 2 boost_clock_MHz : 1531		Gigabyte GV-N1030D5-2GL SCORE: 42.248407472765365 ★★★★★ (9) length_mm : 150 memory_GB : 2 boost_clock_MHz : 1506

Rangking de GPU, el sistema recomienda los GPUs que tienen valoraciones, la primera tiene 3 valoraciones más que la segunda, sin embargo la segunda tiene mayor la valoración media, pero la frecuencia de la primera es mayor que la segunda, por eso tienen los dos SCORE muy cercanas, pero considero que es una recomendación bastante buena.

5. Usuario office presupuesto 500:

Tiene el mismo resultado que el usuario con preferencia de uso estándar. Porque es verdad que hasta llegar al 800, su partición de presupuesto total es la misma.

6. Usuario editVideo presupuesto 500:

	ver más AMD Ryzen 3 3200G SCORE: 162.72383189185035 ★★★★★ (64) integrated_graphics : Radeon Vega 8 price : 90.98 tdp_W : 65 socket : AM4 boost_clock_GHz : 4 core_clock_GHz : 3.6 core_count : 4		ver más Gigabyte GV-N1030OC-2GI SCORE: 43.423784631156586 ★★★★★ (8) length_mm : 168 memory_GB : 2 boost_clock_MHz : 1544 color : Black / Orange price : 120 chipset : GeForce GT 1030 core_clock_MHz : 1290
	ver más Gigabyte GA-A320M-S2H SCORE: 61.355750452503955 ★★★★★ (24) color : Black size : Micro ATX price : 66.8 memory_max_GB : 32 socket : AM4 memory_slots : 2		ver más Corsair CMV8GX4M1A2133C15 8 GB SCORE: 28.366190682688625 ★★★★★ (8) speed_type : DDR4 color : Black price_GB : 3.375 first_word_latency_ns : 14.065 CAS_latency : 15 modules : 1 x 8GB speed_frecuency : 2133 price : 27

 <p>ver más</p> <p>Crucial BX500</p> <p>SCORE: 44.87225170443254</p> <p>★★★★☆ (26)</p> <p>price : 26.42 price_GB : 0.11 interface : SATA 6.0 Gb/s type : SSD form_factor : 2.5" capacity : 240 GB</p>	 <p>ver más</p> <p>Seagate Pipeline HD</p> <p>SCORE: 70.14902086263197</p> <p>★★★★☆ (7)</p> <p>cache : 8 MB price : 26.96 price_GB : 0.054 interface : SATA 3.0 Gb/s type : 5900 RPM form_factor : 3.5" capacity : 500 GB</p>
 <p>ver más</p> <p>Xilence XP400R6</p> <p>SCORE: 21.333333333333332</p> <p>★★★★☆ (1)</p> <p>wattage_W : 400 modular : No color : Black / Red price : 35.04</p>	 <p>ver más</p> <p>Deepcool MATREXX 30</p> <p>SCORE: 25.03047898866016</p> <p>★★★★☆ (21)</p> <p>side_panel : Tempered Glass color : Black price : 29.9 type : MicroATX Mini</p>

Gasta 423 sobre 500 euros.

El resultado es parecida que el usuario gaming, con una gráfica que tiene SCORE muy parecida, un RAM peor, pero añadió una Memoria ROM HDD, que los editores de videos necesita una capacidad más grande que los usuarios gaming.

7. Analizar los CPU

 <p>Intel Core i7-7700K</p> <p>SCORE: 212.40994567432432</p> <p>★★★★★ (307)</p> <p>integrated_graphics : Intel HD Graphics 630 price : 430 tdp_W : 91 socket : LGA1151 boost_clock_GHz : 4.5 core_clock_GHz : 4.2 core_count : 4</p>	 <p>AMD Ryzen 7 5800X</p> <p>SCORE: 210.0705395840033</p> <p>★★★★★ (288)</p> <p>price : 245 tdp_W : 105 socket : AM4 boost_clock_GHz : 4.7 core_clock_GHz : 3.8 core_count : 8</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tienen SCORE muy cercanas, ambas tienen su ventaja, la de Izquierda tiene mayor frecuencia base, es la propiedad más importantes de CPU, sin embargo la segunda tiene 8 núcleos, entonces realmente puede dudar entre los dos componentes. Depende un poco del uso de ordenador, si quiere tener más núcleos o más frecuencia base. Que en este caso no está especificada en los algoritmos.

8. Analizar los GPU

**MSI GAMING X**

SCORE: 95 91065069519492

★★★★★ (44)

length_mm : 248

memory_GB : 8

boost_clock_MHz : 1695

color : Black / Silver

price : 940

chipset : GeForce RTX 2060 SUPER

core_clock_MHz : 1470

**PowerColor Fighter**

SCORE: 94.68306491104525

★★★★★ (2)

length_mm : 200

memory_GB : 8

boost_clock_MHz : 2589

color : Black

price : 533.58

chipset : Radeon RX 6600 XT

core_clock_MHz : 1968

Son dos Gpus que tiene SCORE muy cercanas, porque la de derecha tiene 1968 MHz de frecuencia mientras que la izquierda solo tiene 1695MHz, pero sin embargo la GPU izquierda tiene 42 números de valoraciones más que la derecha y mantiene una valoración media casi de un 5, el máximo.

Las valoraciones son datos útil porque una persona que conoce los GPUs sabe que además la marca MSI es mucho mejor que la marca PowerColor.

9. Analizar RAM



Corsair CORSAIR Vengeance LPX 16GB
288-Pin DDR4 SDRAM DDR4 3000 (PC4
24000) Desktop Memory Model
CMK16GX4M1D3000C16 16 GB

SCORE: 36.69954585012753

★★★★★ (6)

speed_type : DDR4

color : Black

price_GB : 3.393

first_word_latency_ns : 10.667

CAS_latency : 16

modules : 1 x 16GB

speed_frequency : 3000

price : 54.29



Patriot Viper 3 16 GB

SCORE: 35.67259661996955

★★★★★ (9)

speed_type : DDR3

color : Black / Silver

price_GB : 2.874

first_word_latency_ns : 10.718

CAS_latency : 10

modules : 2 x 8GB

speed_frequency : 1866

price : 45.99

Son dos productos que tienen valoraciones muy cercanas. Ambas tienen 16GB en total.

La de izquierda tiene ventaja de frecuencia, que tiene 3000 mientras la de derecha solo tiene 1866.

La de derecha tiene ventaja de precio/GB que es más barato.

10. Analizar ROM-SSD



Intel 660p

SCORE: 232.04324156021107

★★★★★ (130)

price : 81.88

price_GB : 0.08

interface : M.2 PCIe 3.0 X4

type : SSD

form_factor : M.2

capacity : 1.02 TB



Silicon Power A60

SCORE: 184.82290855087732

★★★★★ (35)

price : 78.98

price_GB : 0.079

interface : M.2 PCIe 3.0 X4

type : SSD

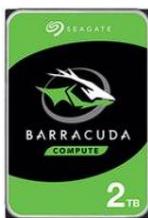
form_factor : M.2

capacity : 1 TB

Ambas tienen 1TB, la de derecha tiene 50 puntuaciones más porque tiene 50 números de valoraciones más, y tiene una valoración media cercanía de 5.

Para comprar SSD, es bastante importante las valoraciones, porque hay mucha diferencia entre RAM de marca buena y RAM de marca mala.

11. Analizar ROM-HDD



Seagate Barracuda Compute

SCORE: 469.46028849956076

★★★★★ (670)

cache : 256 MB

price : 45.99

price_GB : 0.023

interface : SATA 6.0 Gb/s

type : 7200 RPM

form_factor : 3.5"

capacity : 2 TB



Toshiba DT01ACA200

SCORE: 345.49500192771643

★★★★★ (51)

cache : 64 MB

price : 66.98

price_GB : 0.033

interface : SATA 6.0 Gb/s

type : 7200 RPM

form_factor : 3.5"

capacity : 2 TB

Ambas tienen 2TB, pero la de izquierda gana por las valoraciones y su precio/GB, que es más barato.

12. Analizar fuente de alimentación:



EVGA 500 BQ

SCORE: 72.63672408341772

★★★★★ (101)

wattage_W : 500

efficiency : 80+ Bronze

modular : Semi

price : 61.86

type : ATX



EVGA 600 W1

SCORE: 58.786640127277906

★★★★★ (45)

wattage_W : 600

efficiency : 80+

modular : No

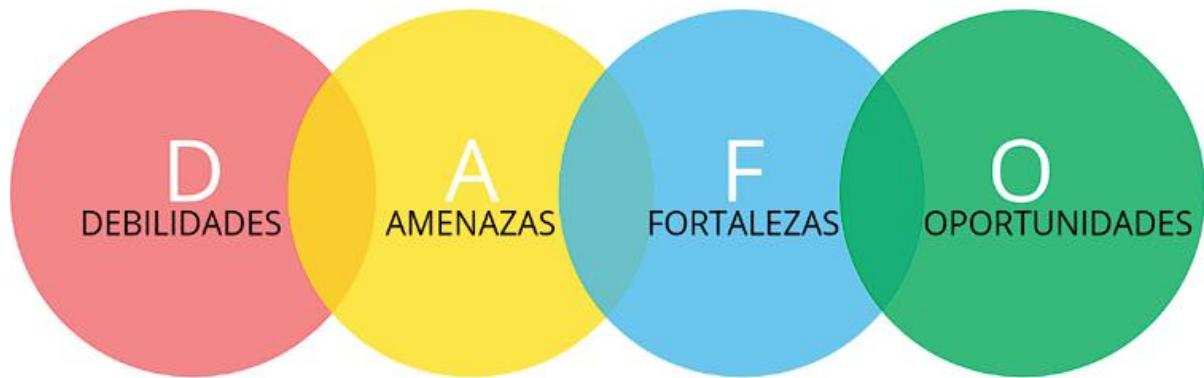
price : 57

type : ATX

La de izquierda tiene mayor voltaje pero pierde, porque la de Izquierda tiene la modularidad: Semi, que aumenta su SCORE_BASE un 25%, y además 50 valoraciones más y una media más alta.

11. Análisis D.A.F.O.

La aplicación viene ya totalmente explicada, pero es importante realizar un análisis DAFO para ver las debilidades, amenazas, fortalezas y oportunidades.



1. Debilidades

a. Códigos

Durante el desarrollo estuve creando muchas funciones, pero al terminar encontré muchos códigos duplicados que pude factorizar las funciones más pequeños para no duplicar tantos códigos, esto puede disminuir mucho más el tamaño de la aplicación y mejorar su rendimiento.

b. Datos limitados

La base de datos de los productos ahora mismo está bastante bien construida, pero el sistema no detecta los cambios, quizá dentro de pocos meses los precios de los productos pueden variar mucho y la aplicación no está con los datos actualizados, es conveniente hacer unas programas que actualiza temporalmente la base de datos.

c. feedback

Como es una aplicación que no tiene usuario no puedo saber cómo va la aplicación para otros usuarios, si les resulta útil o no

2. Amenazas

En el desarrollo de la aplicación no apliqué ninguna tecnología contra las amenazas, las contraseñas de los usuarios están sin encriptar, eso puede provocar problemas muy graves, tampoco implementé técnicas para

prevenir ciberataques, si algún día tengo que publicar esta aplicación, toda esta parte tengo que tener en cuenta.

Pero ahora mismo no me dio tiempo para esta parte.

3. Fortalezas

- a. **Robustez:** El sistema y algoritmos son bastante completos que ofrece buenas recomendaciones teniendo en cuenta las características de los productos, las valoraciones históricas y la preferencia de uso del usuario.
- b. **Utilización de tecnologías de auge:** La aplicación está utilizada las tecnologías de última generación, proporciona una eficiencia relativamente alta.
- c. **Escalabilidad:** una de las ventajas de usar base de datos basados en grafos de conocimiento, el Neo4j.
- d. **Facilidad de uso:** La interfaz de la aplicación es bastante general, la página está formado por la barra lateral, Header, etc.
- e. **Multiplataforma:** es compatible para cualquier resolución de pantalla, la ventaja de una aplicación es que casi todos los sistemas operativos son capaces de usar navegadores, desde cualquier navegador puede acceder a la aplicación.

4. Oportunidades

- a. **Escasa competencia:** Existe muy pocas páginas que recomienda los componentes de ordenadores de manera eficiencia, ni Amazon ni Pc componente, que no tiene en cuenta la preferencia de los usuarios y además en muchos casos venden productos que ya hacen años.

12. Líneas de futuro

Durante el desarrollo de la aplicación he surgido muchas ideas que puede mejorar la aplicación y expandir su funcionalidad.

Podía buscar mejores algoritmos de DeepLearning para repartir mejor el presupuesto para cada componente y asignar algunos constantes como alfa que he utilizado en el cálculo de INCREMENTO, se puede conseguir entrenando los datos históricos para obtener el valor óptimo.

Puede actualizar la base de datos temporalmente para tener siempre los datos actualizados, de hecho el propio Web Scraper ofrece la posibilidad de Scaper periódica, usando API.

Otra parte es mejorar la parte sobre la interacción con el usuario, puede conocer más sus usos de manera más profunda, comprobar de manera automáticamente si su presupuesto es suficiente o no para su interés, puede hacer un sistema de pago en la propia página de recomendación para facilitar la compra, incluso ofrecer servicio de montar el ordenador.

Otra opción es filtrar los productos que quieren comprar, por ejemplo filtrar los productos por las marcas, socket, el tamaño de la Caja/Torre, actualmente se recomienda todos los productos para montar un ordenador nuevo, pero puede ofrecer otra opción de recomendar un componente solo para que actualice su ordenador.

También puede hacer un sistema que valora la puntuación de un ordenador ya montado, e ofrecer ordenadores ya montados.

13. Lecciones aprendidas

Durante el desarrollo de esta aplicación de recomendación, he aprendido a utilizar Neo4j, una base de datos en grafos, lo que me ha brindado una mayor comprensión sobre cómo funcionan estas bases de datos y cómo se pueden utilizar para resolver problemas específicos. Además, he mejorado mis habilidades en frontend y backend, desarrollando la interfaz de usuario y la lógica detrás de la aplicación.

También he tenido la oportunidad de aprender y trabajar con React y Node, lo que ha mejorado mis habilidades en el desarrollo de aplicaciones web. Y por último, he aprendido sobre los diferentes algoritmos de recomendación y cómo se pueden utilizar para personalizar las recomendaciones para los usuarios.

14. Referencias y Bibliografía

<https://neo4j.com/>

<https://scrapy.org>

https://www.webscraper.io/documentation?utm_source=extension&utm_medium=popup

<https://nodejs.org/es/>

<https://expressjs.com/es/>

<https://nodemon.io/>

<https://es.reactjs.org/>

<https://axios-http.com/es/docs/intro>

<https://lodash.com/>

<http://flexboxgrid.com/>

<https://mui.com/>

<https://es.stackoverflow.com/>

<https://chat.openai.com/chat>

<https://www.wolframalpha.com/>

<https://www.xataka.com/especiales/comprar-y-montar-tu-pc-por-piezas-guia-para-elegir-procesador-ssd-ram-y-tarjeta-grafica>

<https://es.pcpartpicker.com/>

<https://www.infoautonomos.com/plan-de-negocio/analisis-dafo/>