



universidad  
de león



## **GRADO EN INGENIERÍA INFORMÁTICA**

Sistemas de Información de Gestión y Business  
Intelligence

### **MEMORIA**

**Autor:** Maday Pablos Yugueros  
**Profesor:** Enrique López González

**Fecha:** 09-12-2020

# ÍNDICE

|                                   |           |
|-----------------------------------|-----------|
| <b>INTRODUCCIÓN</b>               | <b>3</b>  |
| <b>DESCRIPCIÓN DEL PROBLEMA</b>   | <b>4</b>  |
| <b>HERRAMIENTAS</b>               | <b>6</b>  |
| <b>APLICACIÓN</b>                 | <b>8</b>  |
| <b>ALGORITMO DE RECOMENDACIÓN</b> | <b>20</b> |
| <b>ANÁLISIS DE RESULTADOS</b>     | <b>22</b> |
| <b>DAFO</b>                       | <b>25</b> |
| <b>LÍNEAS DE FUTURO</b>           | <b>27</b> |
| <b>LECCIONES APRENDIDAS</b>       | <b>28</b> |
| <b>BIBLIOGRAFÍA Y ENLACES</b>     | <b>30</b> |

# INTRODUCCIÓN

A lo largo de este trabajo analizaremos un problema concreto de recomendación y lo resolveremos mediante un algoritmo de recomendación. Para ello he realizado un sitio Web donde se expone el problema ya resuelto.

El problema concreto es recomendar una serie de libros infantiles, especificando la edad del niño.

# DESCRIPCIÓN DEL PROBLEMA

El problema al que me he enfrentado ha sido realizar un programa de recomendación de libros infantiles. Cuando empecé a plantearme las diferentes opciones que tenía lo primero que pensé es de qué forma lo iba a hacer, si iba a recomendar en función del género, del autor, del número de dibujos....

Después de meditarlo un poco decidí que la mejor manera era recomendarlo en función de la edad, ya que al ser dedicado a niños, lo más fácil para el usuario es simplemente introducir la edad del niño.

A la hora de encontrar un libro para un niño es complicado dar en la clave, por ello he decidido mostrar cuatro libros diferentes, de los cuales mostraré el título, el autor, el precio, la puntuación que tiene, la portada del libro, y un pequeño resumen del libro.

También como no todos los niños crecen al mismo ritmo he decidido establecer unos rangos de edad, para que haya más diversidad. Pero por si acaso se quiere sacar un libro en concreto puedes filtrar.

También me pareció importante establecer que la página no va dirigida a los niños, sino que va dirigida a los padres, aún así también me pareció importante hacer una página muy intuitiva y fácil de usar, ya que aunque mis líneas de futuro vayan encaminadas a que el usuario de mi Web sean los padres, si un niño quiere buscar un libro también sea fácil.

Cuando te enfrentas a un sitio de recomendación de libros creo que es muy importante que cualquier persona pueda usarla, desde un niño que va a buscar un libro interesante para leer en vacaciones, como una persona mayor que busca un libro para leer en su tiempo libre.

Lo más difícil de todo esto yo creo que es que resulte atractiva, ya que aunque al final lo más importante es que la página cumple con su función, nunca olvidemos que la mayoría de personas sí que juzgan un libro por su portada.

Así que me esforzado en que además de cumplir con su función sea atractiva tanto para niños como para adultos.

# HERRAMIENTAS

La verdad es que al principio de este trabajo me encontré con varias dificultades y varias relacionadas con las herramientas que iba a usar. Una vez que tenía elegido el tema sobre el que iba a ir debía encontrar un conjunto de datos que se adaptaran a lo que yo quería conseguir. Después de consultar muchos datos mediante *kaggle* no encontré nada que me gustara, bueno directamente no encontré ninguna que tratara sobre literatura infantil.

Así que el primer problema fue diseñar mi propio conjunto de datos, yo lo organicé de la siguiente manera, en un documento.csv la primera línea contiene el nombre de cada uno de los datos, en este orden, Nombre, Autor, Precio, Genero, Puntuación, Edad, Imagen, Resumen. Mi conjunto de datos no es muy grande, no llega a doscientas líneas, siendo cada línea un dato distinto. Cada atributo de cada dato, en este caso, libros, se encuentran separados por “;”.

La principal herramienta que tenía que usar era NEO4J, totalmente desconocida para mí y he de decir que al principio fui muy reticente a usarla. Tardé bastante en adaptarme a ella, me hice un par de cursos de Neo4J y he de decir que la creación de la base de datos fue muy sencilla y muy visual con Neo.

Decidí tener dos tipos de relaciones para que el filtrado fuera más sencillo, una sería mediante la edad, y la otra mediante el género o el propósito, establecí tres tipos, “Lectura, escritura, u Otros” dependiendo de a lo que quieras que se enfoque el libro.

Para desarrollar el sitio Web utilicé Visual Studio, ya que ya lo había utilizado anteriormente y me sentía cómoda usándolo. Utilicé como lenguaje de desarrollo, JavaScript.

Una aplicación enfocada al usuario se divide en dos partes, el frontend, es la parte enfocada a desarrollar lo que el usuario va a ver, y el backend, enfocado a las conexiones entre la aplicación y la base de datos. Por tanto he utilizado diferentes cosas para cada una de las partes.

Para el frontend, he usado vue.js, Vuetify, Vue-router y axios. Tanto Vue-router como axios lo use para la conexión con el backend y con la base de datos.

Para el backend también use axios y Vue- router para la conexión con la base de datos y con el frontend, y el driver de Neo4J para establecer la conexión con Neo4J.

Otra herramienta que me ha venido muy bien para hacer comprobaciones de peticiones del backend a la base datos y ver que todo era correcto es Postman. La verdad es que es cien por cien recomendable y está muy bien para ver que realmente estás haciendo bien las llamadas a la base de datos y también para ver que los datos están bien introducidos en la base de datos.

# APLICACIÓN

Mi aplicación comienza con la creación de un conjunto de datos que más tarde me valdrán para la creación de la base de datos. Empecé a buscar un conjunto de datos en kaggle que se adaptara a lo que yo tenía en mente, pero no encontré nada, ni siquiera un conjunto de datos que tratara sobre literatura infantil. Así que decidí que aunque me iba a llevar mucho más trabajo iba a crear mi propio conjunto de datos, ya que tenía muy claro la forma que quería que tuviera.

| Nombre                     | Autor           | Precio | Genero  | Puntuacion | Edad | Imagen  |
|----------------------------|-----------------|--------|---------|------------|------|---|
| ADIVINA CUANTO TE QUIERO   | Sam McBratney   | 5      | Lectura | 8          | 0    | <a href="https://i.ytimg.com/vi/URz5RvDkk1A/hqdefault.jpg">https://i.ytimg.com/vi/URz5RvDkk1A/hqdefault.jpg</a>                             |
| ADIVINA CUANTO TE QUIERO   | Sam McBratney   | 5      | Lectura | 8          | 1    | <a href="https://i.ytimg.com/vi/URz5RvDkk1A/hqdefault.jpg">https://i.ytimg.com/vi/URz5RvDkk1A/hqdefault.jpg</a>                             |
| ADIVINA CUANTO TE QUIERO   | Sam McBratney   | 5      | Lectura | 8          | 2    | <a href="https://i.ytimg.com/vi/URz5RvDkk1A/hqdefault.jpg">https://i.ytimg.com/vi/URz5RvDkk1A/hqdefault.jpg</a>                             |
| ADIVINA CUANTO TE QUIERO   | Sam McBratney   | 5      | Lectura | 8          | 3    | <a href="https://i.ytimg.com/vi/URz5RvDkk1A/hqdefault.jpg">https://i.ytimg.com/vi/URz5RvDkk1A/hqdefault.jpg</a>                             |
| SIEMPRE TE QUERRÉ PEQUEÑIN | Debi Gliori     | 10     | Lectura | 7          | 0    | <a href="https://estaticos.serpadres.es/media/cache/1140x">https://estaticos.serpadres.es/media/cache/1140x</a>                             |
| SIEMPRE TE QUERRÉ PEQUEÑIN | Debi Gliori     | 10     | Lectura | 7          | 1    | <a href="https://estaticos.serpadres.es/media/cache/1140x">https://estaticos.serpadres.es/media/cache/1140x</a>                             |
| SIEMPRE TE QUERRÉ PEQUEÑIN | Debi Gliori     | 10     | Lectura | 7          | 2    | <a href="https://estaticos.serpadres.es/media/cache/1140x">https://estaticos.serpadres.es/media/cache/1140x</a>                             |
| SIEMPRE TE QUERRÉ PEQUEÑIN | Debi Gliori     | 10     | Lectura | 7          | 3    | <a href="https://estaticos.serpadres.es/media/cache/1140x">https://estaticos.serpadres.es/media/cache/1140x</a>                             |
| EL GRÚFALO                 | Julia Donaldson | 22     | Otros   | 10         | 0    | <a href="https://imagessl6.casadellibro.com/a/l/t5/56/9788469605356.jpg">https://imagessl6.casadellibro.com/a/l/t5/56/9788469605356.jpg</a> |

(Pequeña muestra del conjunto de datos)

Como se puede observar en la pequeña muestra, he decidido que cada línea del documento va a ser luego en la base de datos un libro, y dentro de la línea tengo diferentes atributos, entre los que se encuentran el título del libro, el autor, el precio, el género que he decidido dividirlo en tres, Lectura, Escritura; y Otros, dependiendo de qué queremos que mejore lectura, escritura, u otro.



También añadí una puntuación del libro del uno al diez, que es puntuada por la gente que ha leído el libro. La edad va a ser algo importante ya que es en lo que se basa mi algoritmo de recomendación. Por último, la imagen que es la foto de la portada del libro.

Cuando empecé la aplicación mi conjunto de datos contaba con un total de sesenta libros y al final me animé a ampliarla un poquito y he conseguido ampliarla hasta tener un total de ciento noventa y ocho. De todas formas considero que mi aplicación tiene muchas posibilidades de crecer sobre todo en la base de datos.

He de decir que hacer el conjunto de datos me ha llevado mucho más tiempo del que creía ya que también quería añadir un resumen de cada libro, pero no entiendo porque en el momento que lo añadía mi aplicación dejaba de funcionar.

Así que primero tuve que hacer el conjunto, una vez que lo probé y no me iba, llegué a la conclusión de que el problema podrían ser dos cosas o la imagen de la portada del libro o el resumen. Mi forma de averiguar qué era fue borrar todas las imágenes y todos los resúmenes del conjunto de datos, y poco a poco ir añadiendo y ver cuando funcionaban y cuando no.

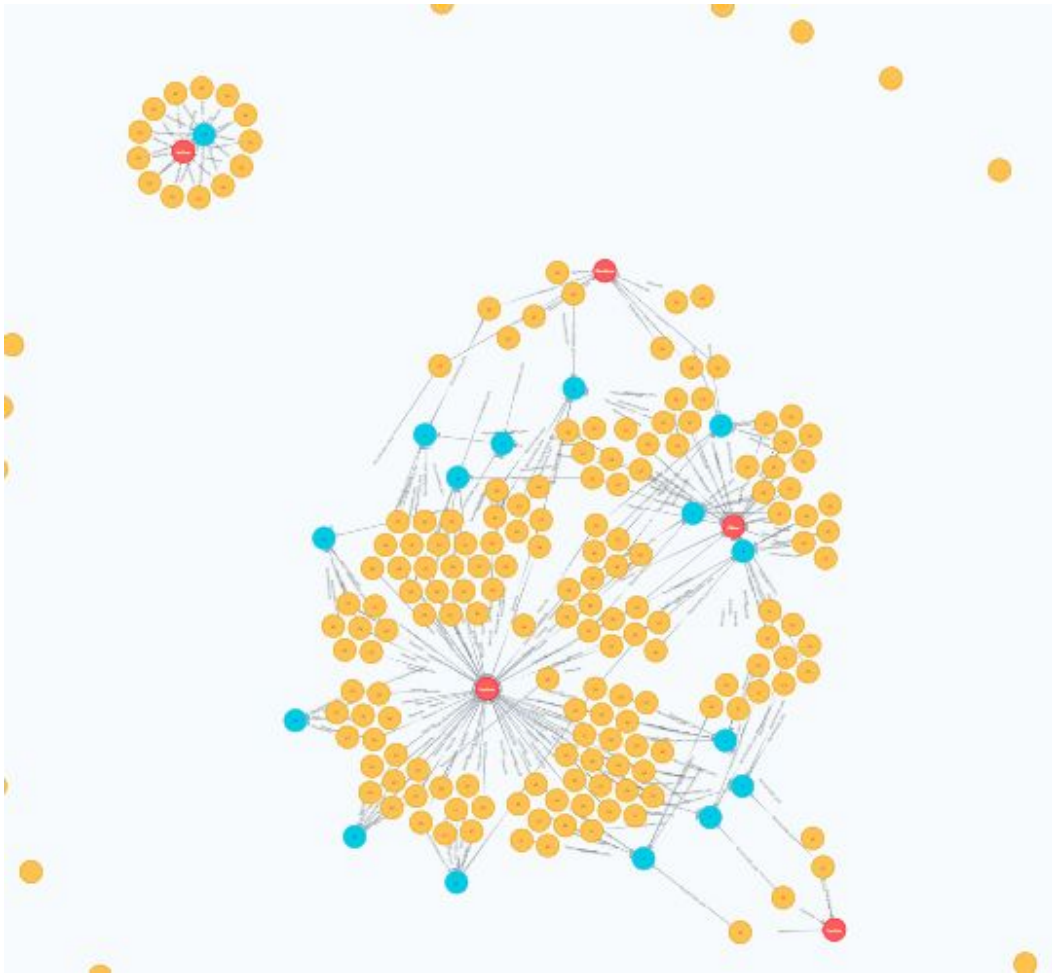
Cuando añadí las imágenes al primer conjunto no me daba fallo, así que continué añadiendo las imágenes por conjuntos y probando. Me di cuenta de que nada más añadir al primer conjunto los resúmenes mi sitio Web dejaba de funcionar.

Una vez que tenía el conjunto de datos me dispuse a realizar la base de datos, para ello use Neo4J. Me costó bastante empezar a usarla, realicé dos cursos de Neo y la verdad es que mi gran problema con esta plataforma, aunque pueda parecer un poco absurdo, fue el idioma, me costaba mucho entender bien las cosas en inglés, y el 98% de la información de Neo4j está en inglés. Intentaba usar las traducciones pero la verdad es que son bastante malas.

De todas formas, echando la vista atrás y viendo los resultados obtenidos he de decir que comparado con otras plataformas que ya había utilizado como MySQL, Neo4j es mucho más intuitiva y te ahorra mucho trabajo. Además es mucho más visual una vez que tienes ya la base de datos hecha.

El lenguaje que usa Neo4j es Cypher, que aunque pueda parecer muy complicado la verdad es que, no se si es debido a todas las horas que le dedique a aprenderlo y a que es un lenguaje sencillo, me pareció muy fácil tanto añadir la información de mi conjunto de datos como crear las relaciones entre los libros ya dentro de la base de datos.

He de decir que estoy muy contenta de haber descubierto Neo4j y haber podido trabajar con él, ya que me permitió hacer todos los cambios que quería y trastear mucho con el conjunto de datos sin que eso me llevara horas de trabajo para incluir los cambios en la base de datos.



(Base de datos)

Aquí muestro una parte de mi base de datos, sobretodo la parte de las relaciones. los círculos que aparecen en color amarillo son los libros, que son los atributos qu forman la base de datos. Los círculos rojos y azules establecen los dos tipos de relaciones que he creado. Los azules relacionan libros por la edad, y los rojos por el género.

Como consejo diría que hay que tener bastante cuidado a la hora de introducir los datos en la base de datos ya que para Neo4j no es lo

mismo “*Lectura*” que “*lectura*”, ya que distingue entre letras mayúsculas y minúsculas, y te creará relaciones distintas.

Para introducir los datos en la base de datos use el siguiente código Cypher:

```
LOAD CSV WITH HEADERS FROM 'file:///lecturasInfantil.csv' AS line
FIELDTERMINATOR ';'
CREATE ( v:Libro)
set v= {
nombre: line.Nombre,
autor: line.Autor,
precio: line.Precio,
género: line.Género,
puntuación: line.Puntuacion,
edad: line.Edad,
imagen: line.Imagen
}
```

(Código Cypher para cargar los datos a la base de datos)

A la hora de indicar el lugar donde tenía el documento “lecturasInfantil.csv”, que era donde tenía el conjunto de datos, tuve bastantes problemas para establecer la ruta, así que investigando un poco descubrí que si en la carpeta *import* dentro de nuestro proyecto Neo4j colocas el documento Neo4j tiene acceso directo a él y por tanto solo hace falta especificar el nombre de la ruta.

Yo establezco que en mi base datos cada círculo, que es un nodo para Neo4j, para mi va a ser un libro, y lo denoto como "*v:libro*". Una vez que le he dicho a Neo4j lo que es cada nodo, tengo que especificar qué atributos va a tener cada uno, si se fija en el conjunto de datos, en la primera línea aparecen el nombre de cada uno de los atributos.

Entonces mi siguiente paso sería indicar que los atributos aparecen todos en una línea, el orden en el que van a aparecer, y después especifique que iban a aparecer separados por ";". Probé a separarlos por comas pero Neo4j no lo leía bien. Una vez que lo separé po punto y coma funcionaba a la perfección.

Para establecer las diferentes relaciones entre los libros use el siguiente código, tuve problemas ya que me daba errores por valores nulos aunque la verdad es que no tengo ningún valor nulo.

```
FOREACH ( n IN (CASE WHEN line.Edad IS NULL THEN [] ELSE [1] END) |  
MERGE (e:Recomendado{edad:line.Edad})  
MERGE (v)-[:Recomendado_para]->(e)  
)
```

```
FOREACH ( n IN (CASE WHEN line.Genero IS NULL THEN [] ELSE [1] END) |  
MERGE (e:Habilidad{habilidad:line.Genero})  
MERGE (v)-[:Desarrolla]->(e)  
)
```

(Código Cypher para establecer relaciones)

Decidí crear dos tipos de relaciones porque veo mucho futuro a filtrar por género, de hecho en mi primera opción estaba filtrar por género y por edad. Pero al final no lo veía claro por como lo había montado.

El único problema que he tenido con la base de datos fue a la hora de intentar añadir un atributo resumen a cada libro. Al ser un texto tan largo no me salía bien y me daba problemas. Así que después de intentarlo varias veces decidí no incluir los resúmenes de los libros, aunque no descarto añadirlos en un futuro.

También es recomendable cuando uses Neo4j usar el comando : *“Match(n) return n”*. Para que muestre todo lo que contiene la base de datos. Es importante también es importante que cada vez que se añadan libros se borre la base de datos porque si no vamos a tener datos duplicados, para ello lo más fácil es usar *“Match(n) delete n”*, pero como hay relaciones entre los nodos tenemos que usar el siguiente comando:

```
MATCH (n)
OPTIONAL MATCH (n)-[r]-()
DELETE n,r
```

*(Comando para eliminar nodos y relaciones)*

Una vez que ya lo tenía todo montado en Neo4j tocaba ponerse a crear lo que sería mi Sitio Web. Para ello lo tenía que dividir en dos partes, el frontend, y el backend. El backend ocupa la parte del backend.js, está en JavaScript y consiste en la conexión con la base de datos y mandar los datos necesarios al frontend.

La parte de la conexión de datos he de decir que me costó un poco y que la verdad no encontré la suficiente información que necesitaba por internet, y me las tuve que apañar yo sola o con la ayuda de mis compañeros. Al principio me daba un error raro que no entendía que podía ser y debido a esa falta de información tuve que borrar la base de datos y volver a crearla, aunque como ya comenté anteriormente el hecho de borrar y crear una base de datos en Neo4j no es muy complicado ya que una vez que lo haces por primera vez y dispones de los comandos Cypher adecuados es simplemente copiar y pegar.

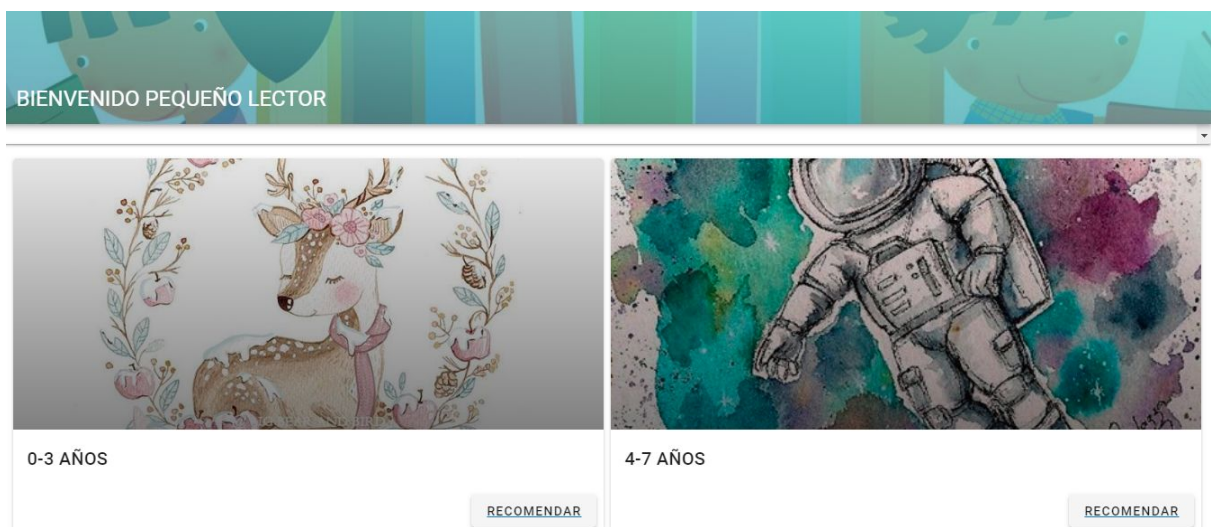
Tras varios días intentando que se conectaran lo conseguí, recomendando una herramienta, que es un complemento de google, llamado "Postman", que a mi me fué de gran utilidad ya que te permite ver si realmente te está sacando bien los datos y si la conexión es correcta. Por ejemplo, tuve en error que gracias a Postman pude solucionar sin romperme mucho la cabeza, cuando introducía un valor numérico lo hacía como : *"Integer(line.Valor)"* , y no me mostraba bien el valor, tuve que quitarle el *Integer*.

El funcionamiento del backend no es muy complicado, también he de decir que tengo un algoritmo de recomendación sencillo. Al backend desde el frontend le paso una edad y mediante una query saco todos los libros en un rango de edad determinado. Pero eso lo explicaré mejor en el apartado del Algoritmo de recomendación.

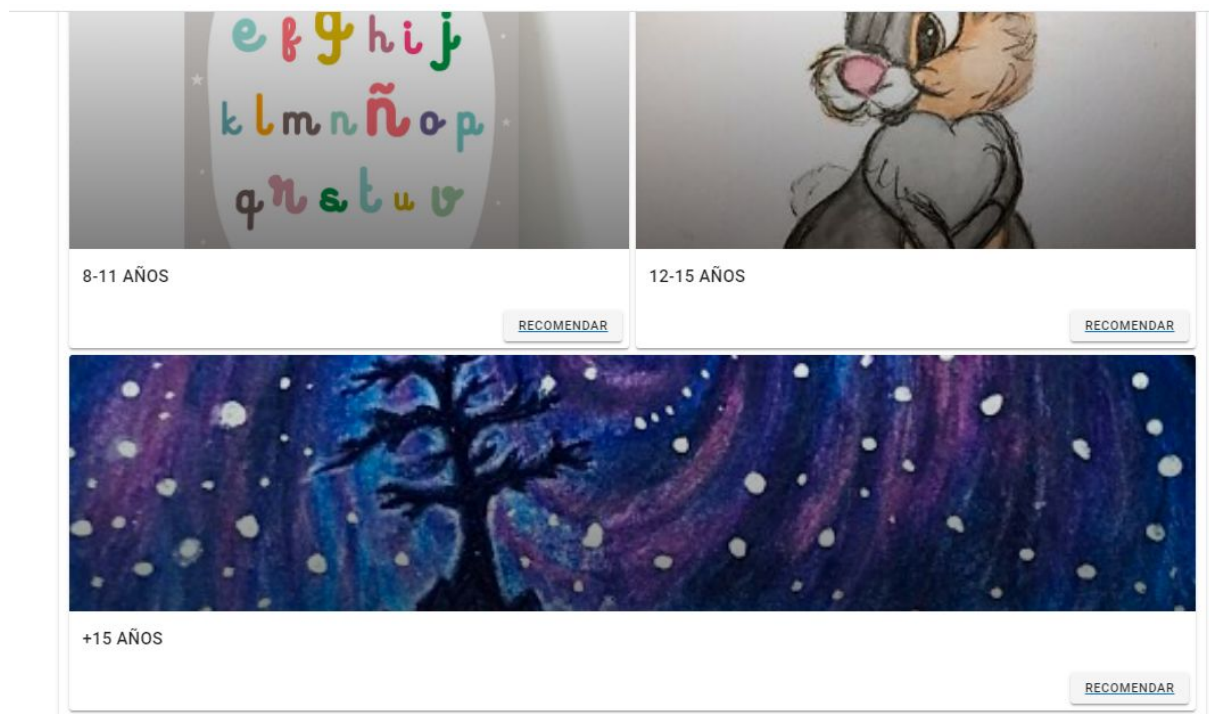
Una vez que he obtenido todos los libros los guardo en un array y se lo mando al frontend donde allí operará con ellos.

Pasamos al frontend, aquí tuve bastantes dudas de la forma que quería que tuviera, al principio quería que todo se mostrara en la misma view, que se filtrara por la edad concreta.... Tuve un problema al principio que me borró todo lo que tenía hecho y creo que me vino bien para establecer lo que quería.

Al final mi sitio Web tiene dos views distintas, quería que fuera navegable entonces incluí botones que te permiten saltar de una a otra sin problema. La view “Home.vue” es la ventana principal.





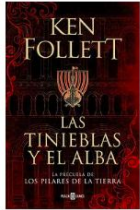




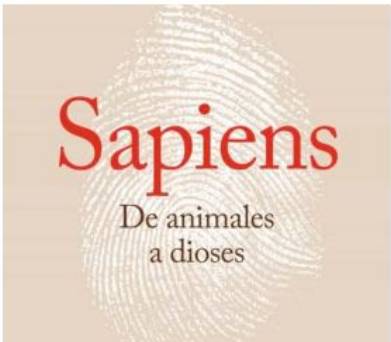
(Ventana principal)

En la venta principal tengo una serie de cartas, concretamente cinco con diferentes rangos de edad. En cada carta se encuentra un botón, “*RECOMENDAR*”, que al pulsarlo te redirige a la segunda ventana.

Si analizamos el código, cada carta está numerada del uno al cinco, diferenciando cada grupo de edad. Al pulsar en el botón “Recomendar”, se le pasa a un método el número de la carta en la cual se ha pulsado el botón. Posteriormente nada más pulsar el botón se indica que se tiene que cambiar de ventana, y pasaremos a “*recomendar.vue*” .

[VOLVER A INICIO](#)

|  |   |
|--|---|
| <p><b>LAS TINIEBLAS Y EL ALBA</b></p> <p>El autor de este libro es, KEN FOLLETT, el precio es 25 euros, la puntuación de este libro es, 10</p> |  |
| <p><b>AQUITANIA</b></p>  |   |

|  |  |
|--|--|
| <p><b>¿QUE PODEMOS PERDER?</b></p> <p>El autor de este libro es, SANDRA MIRO, el precio es 14 euros, la puntuación de este libro es, 9</p>               |   |
| <p><b>SAPIENS (DE ANIMALES A DIOS)</b></p> <p>El autor de este libro es, YUVAL NOAH HARARI, el precio es 25 euros, la puntuación de este libro es, 9</p> |  |

**(Ventana Recomendar)**

En la ventana de “*recomendar*” aparecen 4 libros recomendados para esa edad. Junto con cada libro sale el título del libro en grande, mayúscula, y en negrita, el autor del libro, el precio del libro, y una puntuación del uno al diez.

También a la derecha y en grande aparece la foto de la portada del libro, ya que no hay que juzgar un libro por su portada pero muchas veces es lo que nos anima a elegir entre un libro u otro. Además como la Web no ofrece la opción de compra también es más fácil a la hora de buscar el libro que te ha interesado en la librería reconocerlo por la portada.

En esta ventana disponemos de un botón que se llama “*Volver a inicio*”, a través del cual podemos volver a la pantalla de inicio. Si volvemos a darle al mismo botón de recomendar nos saldrían libros diferentes, ya que aunque la base de datos es pequeña hay aproximadamente unos quince libros por edad lo cual nos permite ver cómo van variando los resultados.

# ALGORITMO DE RECOMENDACIÓN

Mi algoritmo de recomendación no es muy complicado, consiste básicamente en un filtrado un poco diferente. Me pareció que definir un libro concreto para una edad concreta es muy difícil, ya que no todos los niños crecen al mismo ritmo ni tienen las mismas necesidades, así que decidí que en vez de recomendar para una edad concreta, establecer unos rangos de edad determinados.

Para establecer el rango de edad me basé en como están establecidos en una biblioteca o en una tienda de libros, habiendo diferentes grupos, yo establecí cinco grupos. El primer grupo son los libros dirigidos a niños que todavía no saben leer que son de cero a tres años, por tanto son libros no enfocados tanto a lectura sino más visuales.

El segundo grupo son los que van dirigidos a niños de cuatro a siete años, estos están dirigidos a los niños que están aprendiendo a leer, o que todavía no tienen mucha soltura leyendo. La verdad es que luego no sabía si hacer más distinciones o no, pero había muchas edades de por medio así que decidí hacer dos distinciones hasta los quince años, de ocho a once años, y de doce a quince años.

Por último me pareció interesante poner la recomendación de más de quince años, ya que recomiendan libros interesantes ya de lectura un poco más densa que puede ser una buena línea de futuro para ampliarlo a todas las edades.

El frontend sería el que le pasara la edad al backend y este haría la petición a la base de datos. La comunicación entre el frontend y el backend, se realiza entre *“recomendar.vue”* y *“backend.js”*.

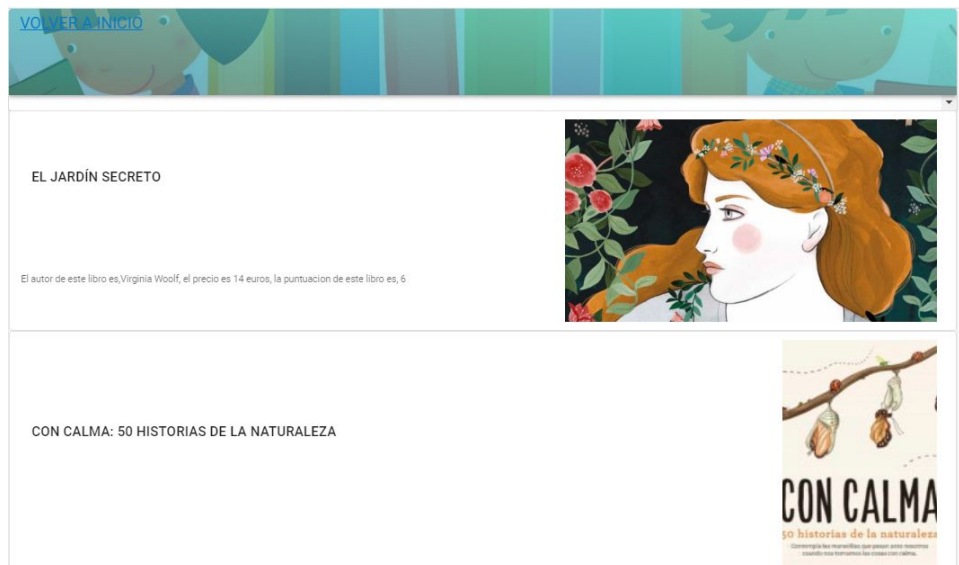
El backend.js le manda el array a recomendar.vue y en esta ventana colocamos cada título, edad, precio, puntuación e imagen en un array y en un orden determinado y sería el propio recomendar.vue el que elige qué libros mostrar por medio de números aleatorios.

Al principio pensé en hacerlo recomendando por mayor puntuación pero de esta manera me dí cuenta que siempre se iban a mostrar los mismos libros y me interesaba tener variedad en los libros y que se pusieran recomendar varios libros.

# ANÁLISIS DE RESULTADOS

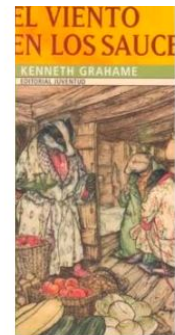
Realmente mi proyecto es muy sencillo pero exacto, no tiene fallos. Siempre va a tener un resultado, me he encargado de que pase lo que pase siempre hay mínimo un resultado.

Se puede entender mejor el resultado de mi sitio Web en el video que acompaña a este proyecto pero aun asi voy a poner diferentes capturas de los resultados obtenidos.



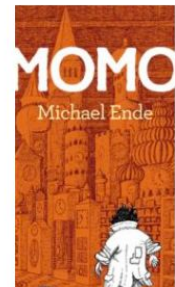
EL VIENTO DE LOS SAUCES

El autor de este libro es,Kenneth Grahame, el precio es 21 euros, la puntuacion de este libro es, 7



MOMO

El autor de este libro es,MICHAEL ENDE, el precio es 17 euros, la puntuacion de este libro es, 8

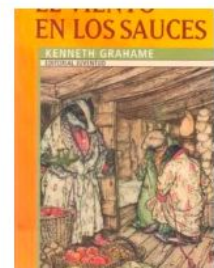


(resultado de recomendar libros para 8-11 años)

[VOLVER A INICIO](#)

EL VIENTO DE LOS SAUCES

El autor de este libro es,Kenneth Grahame, el precio es 21 euros, la puntuacion de este libro es, 7



DONDE VIVEN LOS MOSNTRUOS

El autor de este libro es,Maurice Sendak, el precio es 17 euros, la puntuacion de este libro es, 8



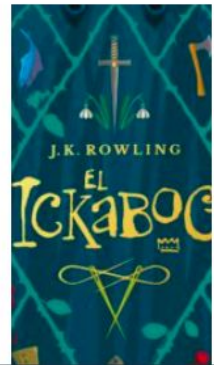
DISNEY BUENAS NOCHES

El autor de este libro es,Disney, el precio es 17 euros, la puntuacion de este libro es, 8



EL ICKABOG

El autor de este libro es,J.K. ROWLING, el precio es 18 euros, la puntuacion de este libro es, 9



(Resultado de recomendar por segunda vez un libro para 8-11 años)

Como podemos observar en el ejemplo se obtienen resultados diferentes para una misma búsqueda aunque puede que coincida algún libro en concreto o no, por eso mismo me gustaría ampliar más la base de datos y poder mejorar el filtrado, para poder concretar aún más la búsqueda de los libros.



# DAFO

En este apartado voy a estudiar las limitaciones y las ventajas que tiene mi proyecto. La verdad es que es algo que tengo muy claro, sé muy bien que mi Web es eficaz porque cumple su trabajo, pero también sé que podría haber hecho mucho más.

La ventaja de mi sitio web es que no hay casos malos, siempre va a tener un resultado sea cuál sea la pregunta, ya que aunque mi conjunto de datos no es muy grande ya que cuenta solo con doscientos libros, es lo suficientemente completa como para sacar un resultado siempre.

También he de decir que me esforcé mucho en que fuera eficiente, que aunque el trabajo que realiza puede llegar a resultar simple, sea un buen trabajo y no de ningún error.

Limitaciones la verdad es que a mi parecer tiene muchas, ya que si quieres buscar un libro en concreto no puedes. Es algo que me gustaría haber podido incluir. Poder filtrar por género o por puntuación.

Otra cosa que me gustaría haber podido incluir es que si te gusta un libro poder darle a me gusta y que eso se guarde en la base de datos, para mostrar primero esos, o incluso que se te guardaran o se te quedaran almacenados.

Una ventaja sería que tiene muy buena capacidad de crecimiento, funcionaria igual de bien con doscientos libros que con diez mil, lo cual es muy beneficioso, ya que nos permite que a medida que crece la base de datos el rendimiento de la web no baje.

Una limitación también es que los libros son mostrados de una manera un poco aleatoria si te gusta un libro y recargas la página se puede cambiar el libro que te gusta, y bueno ahora teniendo una base de datos pequeña no importa porque seguramente no pasaría mucho hasta que volviera aparecer pero con una base grande seguramente no volviera aparecer.

Me gustaría en un futuro poder hacer otra ventana con favoritos, y al poner un usuario y una contraseña, que te salgan los libros que te gustan o que te han parecido interesantes.

De todas formas no es algo que me preocupe en exceso ya que añadiendo un link a la compra del libro quedaría registrado en la pagina de venta del libro y ya quedaría marcado que has visto ese libro y que te gusta.

# LÍNEAS DE FUTURO

La verdad es que cuando empecé con este trabajo no sabía a donde llegaría ni que conseguiría. Una vez que he realizado el trabajo la verdad es que he visto muchas opciones de crecimiento.

La primera que veo es añadir otra ventana para diferenciar los tipos de recomendaciones, evidentemente este sitio va dirigido a los padres, por tanto estaría bien recomendar diferentes libros a la hora de abordar temas como problemas con los niños.

También me parece que puede dar el salto y convertirse en un sitio Web de recomendaciones en general filtrar por edades y elegir libros que puedan ir encaminados a los diferentes problemas que puedan aparecer en las distintas etapas de la vida.

La verdad es que aunque estoy bastante contenta con el trabajo realizado, cuanto más hacía más veía que podía hacer. Por ejemplo me hubiera gustado ahora al final, poder haber hecho un filtrado por género además de por edad, ya que si observa mi base datos ya está hecha la relación porque tenía esa idea desde el principio.

# LECCIONES APRENDIDAS

En primer lugar, quiero decir que estoy contenta con el trabajo que he realizado, ya que es cien por cien mío, desde el conjunto de datos hasta la interfaz. Podría haber hecho muchas cosas más, pero he de decir que me llevó más tiempo del que esperaba obtener un conjunto de datos amplio y completo con el que poder trabajar.

La verdad es que he aprendido mucho realizando este trabajo, ya que he tenido que buscarme mucho la vida en el sentido de que no sabía cómo funcionaba Neo4j, y tuve que ponerme mucho las pilas ya que sentía que me costaba un montón aprender cómo funcionaba.

El gran problema que tuve con Neo4j fue el idioma, el 99% de toda la información que hay sobre Neo4j está en inglés. Sentía que por muchos cursos que hiciera, o muchos vídeos que viera no estaba aprendiendo cómo funcionaba Neo4j.

Decidí que la mejor manera que había de aprender era instalarme la aplicación de Neo4j y empezar a cacharrear, en ese momento me dí cuenta que todo lo que había hecho y todos los videos que había visto sí que habían merecido la pena.

Es verdad que la mayoría de información que necesitas está en internet, pero también hay que buscarse mucho la vida, y cacharrear mucho, por que aunque parezca que la informática es A o B no siempre es así, en

esta práctica me han salido errores, en concreto recuerdo uno nada más empezar Neo4j.

Me salio un error nada más empezar con Neo4j, era un error que aparecía en casi todos los foros que hay de Neo4j, pero en ninguno explican cómo solucionarlo. Lo que yo hice fue borrar la base de datos y volver a crearla y simplemente con eso se me soluciono.

Puede parecer una solución fácil pero llegar a esa conclusión me llevó dos días de romperme la cabeza pensando que era lo que estaba haciendo mal.

Tenía mucho miedo a este proyecto, me costó mucho que mi programa funcionara a la perfección y cuando empecé con ello pensé que era imposible que yo sola sacara adelante un proyecto como este pero nada es imposible echándole horas y poniéndole empeño.

# BIBLIOGRAFÍA Y ENLACES

- API de Vuetify:  
<https://v2.vuetifyjs.com/>
- Información borrar base de datos:  
<https://stackoverflow.com/es/q/1068977>
- Videos Vueify:  
[https://www.youtube.com/channel/UCW5YeuERMmInqo4oq8vwUp\\_g](https://www.youtube.com/channel/UCW5YeuERMmInqo4oq8vwUp_g)
- Página de Neo4j:  
<https://neo4j.com/graphacademy/training-intro-40/02-neo4j-graph-platform/>
- Postman:  
<https://www.postman.com/>
- La casa del libro:  
<https://www.casadellibro.com/>