

VIRTUAL BOOKS

Desenvolupament d'una biblioteca virtual individual global.

Mèmorria de treball de final de grau.

Grau Multimèdia, itinerari Enginyeria Web.

Autor: Aleix Tello Mainar
Consultor: Ignasi Lorente Puchadas
Professor: Carlos Casado Martínez
Data d'entrega: 22/06/2018



Als centenars de músics que m'han acompanyat durant tot el procés de naixement i creixement del projecte.

A tu, que l'estàs llegint.

ABSTRACT

Virtual Books neix de la necessitat de mantenir un ordre de la biblioteca que els lectors poseeixen. A més a més, dona la possibilitat de posar al mercat de segona mà els títols que es vulgui i realitzar una llista de llibres desitjats, convertint així l'aplicació en un centre de coordinació de tots els llibres de cadascú.

Consisteix en crear-se un usuari, crear la biblioteca personal, i a partir d'aquí poguer llistar els llibres poguent fer les recerques que es considerin necessàries, obtenint així els llibres que falten per X autor, o la resposta a si X llibre el vas agafar a la biblioteca municipal, el vas comprar, o te'l van deixar.

L'aplicació està desenvolupada mitjançant la metodologia MEAN (MongoDB - ExpressJS - Angular - NodeJS), que permet abordar tot el projecte mitjançant JavaScript.

ABSTRACT IN ENGLISH

Virtual Books was born by the idea of keep an order of the library that each one possess. No excel, no paperbooks (damn it!).

In addition, it gives the possibility of placing the books that you doesn't like in the second-hand market preventing them from picking up dust at home. Win-win scenario for everyone.

Virtual Books will be the books data center of each house.

It consists of creating a user, put it on the books that you own, and from there begin to make a lists. Lists are fun, Virtual Books is fun. You will know if you are a big fan of the indie editorial of the moment, if you're paying the bills of a specific author or if you're friends only gives you books every Christmas, birthday, wedding or whatever celebration that you do.

Virtual books is developed using MEAN methodology (MongoDB-ExpressJS-AngularJS-NodeJS), wich allows Javascript to entire project.

ÍNDIX

1. Introducció	
1.1 Context i justificació del treball.....	6
1.2 Objectius	7
1.3 Enfocament i mètode seguit.....	8
1.4 Planificació del treball.....	10
2. Continguts	
2.1 Estructurals.....	11
2.2 Funcionals.....	13
3. Arquitectura de l'aplicació	
3.1 Client.....	14
3.2 Servidor.....	14
3.2 Bases de dades.....	14
4. Plataforma de desenvolupament	
4.1 Software.....	15
4.2 Hardware.....	15
5. Planificació	
5.1 Temporal.....	16
5.2 Diagrama de Gantt.....	18
6. Procés de treball	
6.1 PAC1.....	19
6.2 PAC2.....	19
6.3 PAC3.....	19
6.4 Lliurament Final.....	20
7. APIS utilitzades	
7.1 Facebook i Twitter.....	21
8. Perfils d'usuari	
8.1 Diagrama UML.....	22
8.2 Històries d'usuari.....	23
8.3 Casos d'ús.....	24
9. Wireframes.....	26
10. Usabilitat UX.....	29
11. Seguretat	
11.1 Autenticació dels usuaris.....	31
11.2 'Use strict'	31
12. Fitxes	
12.1 Fitxes dels llibres.....	32
12.2 Fitxes dels usuaris.....	32
13. Instal·lació	
13.1 Requisits d'instal·lació.....	33
13.2 Instruccions d'instal·lació.....	33

ÍNDEX

14. Bugs.....	34
15. Projecció a futur.....	35
16. Estudi de viabilitat	
16.1 Pressupost.....	36
16.2 Anàlisi de mercat.....	37
16.3 Conclusions.....	38
17. Conclusions.....	39
18. Annexos	
18.1 Lliurables del projecte.....	40
18.2 Codi font.....	41
18.3 Llibreries.....	47
18.4 Captures de pantalla.....	48
18.5 Guia d'estil.....	52
18.6 Glossari.....	56
18.7 Bibliografia.....	57
18.8 Recursos.....	57

ÍNDIX

LLISTA DE TAULES I FIGURES

Taula 1. Dates Clau. Entrega de PAC.....	16
Taula 2. Tasques del projecte.....	17
Figura 1. Diagrama de Gantt. Planificació temporal del projecte.....	18
Figura 2. Codi font PassportJS Api Twitter i Facebook.....	21
Figura 3. Diagrama UML. Interacció Usuari/Aplicació.....	22
Figura 4. Wireframe formulari de registre.....	26
Figura 5. Wireframe de benvinguda.....	27
Figura 6. Wireframe organització de biblioteca.....	28
Figura 7. Terminal amb la connexió realitzada.....	33
Figura 8. Funció onKeyUp.....	34
Figures 9 i 10. Captura de pantalla Registre i Login.....	48
Figura 11. Captura de pantalla misstages error.....	48
Figura 12. Captura de pantalla Benvinguda usuari.....	49
Figura 13. Captura de pantalla Biblioteca usuari.....	49
Figura 14. Captura de pantalla Buscar un llibre.....	50
Figura 15. Captura de pantalla Crear un llibre.....	50
Figura 16. Captura de pantalla Administrar Biblioteca.....	51
Figures 17 i 18. Captures de pantalla editar i borrar llibres.....	51
Figura 19. Grid de Bootstrap.....	54
Figura 20. Elements d'interacció. Botons.....	54
Figura 21. Elements d'interacció. Formulari.....	55

1. INTRODUCCIÓ

1.1 CONTEXT I JUSTIFICACIÓ DEL TREBALL

Ordre i aventura és el títol d'un disc de Mishima amb el que ens van obsequiar l'any 2010. Els aficionats a la lectura, acostumen a tenir un especial cuidado i admiració per als llibres. Es compren llibreries i ocupen espais de la casa que no tenen. El seu ordre és una aventura.

Hi ha qui porten el control amb llibretes que són autèntiques relíquies, amb dates de compra, impressions i dades que fan aquell llibre especial per alguna raó.

Altres han aconseguit sofisticar fulls de càlcul en les que han aconseguit filtratges i llistes segons preferències o estats d'ànim.

Hi ha lectors que són fidels a un autor, altres a una editorial, altres que compren lectures segons la portada, els que segueixen recomanacions... Hi ha lectors de tota mena, però a tots els uneix el mateix, un ordre.

Hi ha que l'aconsegueixen, hi ha que el fan servir com a decoració, hi ha que el desconeixen per complet.

Amb aquesta aplicació s'aconseguirà aquest ordre en la dimensió virtual. Llistar, classificar, seleccionar, ordenar, intercanviar, vendre.

Amb Virtual Books es podrà portar un control i un ordre dels llibres que es tenen i es volen de forma senzilla.

1. INTRODUCCIÓ

1.2 OBJECTIUS

L'objectiu general del projecte és crear Virtual Books, una aplicació web que pugui donar el servei dissenyat de manera estable.

A nivell personal, l'objectiu rau en aprendre sobre la metodologia MEAN, que actualment desconec. Abordar un projecte únicament en JS és quelcom que trobo interessant.

El projecte consta de dues fases, en les que podem agrupar els diferents objectius. La primera fase, contindrà els objectius mínims per al desenvolupament del treball i segons timings i corba d'aprenentatge, es poden implementar els objectius de la segona fase.

- Objectius de la primera fase:
 - Desenvolupar la biblioteca virtual.
 - Desenvolupar una aplicació web on l'usuari pugui pujar els llibres que té, i així poguer formar la seva pròpia biblioteca.
 - Desenvolupar una eina que permeti el filtratge segons l'usuari convingui.
 - Aconseguir un entorn agradable i atractiu
 - Mostrar la consolidació dels coneixements adquirits al llarg del grau.
 - Creació del logotip.
 - Selecció del 'packaging'. Colors i aspecte.
 - Decidir idioma.

- Objectius de la segona fase, si escau:
 - Home dinàmica, amb actualitat literaria o cites d'escriptors.
 - Implementar la venta i intercanvi de llibres entre els usuaris.
 - Implementar un xat.
 - Registre a partir de les xarxes socials més utilitzades, com Twitter o Facebook.
 - Desenvolupar l'aplicació per a dispositius mòbils.

1. INTRODUCCIÓ

1.3 ENFOCAMENT I MÈTODE SEGUIT

La idea parteix de dues matrius clares. En primer lloc, la necessitat de crear una plataforma en la que poder portar un control dels llibres que es tenen, i la segona, la facilitat de poder posar-los en venda en un sol clic. Aquesta segona part, podria xocar de front amb l'aplicació que actualment copa el mercat de segona mà, wallapop.

Hi ha diverses plataformes que han decidit independitzar-se de wallapop, per a crear el seu propi mercat, més especialitzat, com podria ser [BKIE](#), de bicicletes.

La distinció s'acostuma a fer per a detallar el producte. En el cas de Virtual Books, les decisions són diverses:

- El filtratge de llibres de text. Virtual Books no és una plataforma on hi tinguin cabuda els llibres de text.
- La correcta presentació de l'stock. Cada llibre tindrà la seva fitxa, perquè l'usuari així ho podrà. Les recerques seràn completes per editorial, per autor, per any d'edició.
- L'agrupació del producte. Al tractar-se nomès de llibres, a l'hora de realitzar les búsquedes, s'eviten dvds o cds.

La metodologia seguida per a dur a terme el projecte ha estat una metodologia clàssica.

El període de la presa de requisits ha estat el més llarg, ja que el plantejament del projecte ha necessitat d'un procés de maduració llarg, i a partir d'aquí, les diferents etapes que s'aniran esdevenint en cascada.

S'ha decidit fer-ho d'aquesta manera ja que previsiblement, les fases del projecte no canviaran, i desde un inici s'establiran dates i terminis fixos, uns marcats per les PACS i altres per els timmings necessaris per a assolir la data de finalització.

Les fites i els punts marcats desde un inici poden tenir alguna variació seguint les pautes i indicacions del consultor durant el desenvolupament del projecte, i en les dates claus, com poden ser les entregues de les PAC.

1. INTRODUCCIÓ

1.4.1 PLANIFICACIÓ DEL TREBALL - EINES DE DESENVOLUPAMENT

Per al desenvolupament del treball, es seguirà la metodologia MEAN. Aquesta Metodologia permet desenvolupar tot el projecte mitjançant Javascript per a cobrir tant el frontend com el backend i les bases de dades.

MEAN respon a MongoDB, ExpressJS, Angular i NodeJS.

MongoDB - Base de dades no relacional. NoSQL. Fa servir la notació JSON per a rebre, enviar i emmagatzemar dades.

ExpressJS - Framework de NodeJS.

Angular - Biblioteca/Framework per al FrontEnd.

NodeJS - Javascript en el costat servidor.

A més a més d'aquesta metodologia, per al desenvolupament del projecte, serà necessari el següent programari i maquinari.

iMac/MacBookPro - Màquina per a moure tot el programari.

Adobe Indesign CC 2018 - Programari per a la maquetació de tots els documents necessaris.

Adobe Illustrator CC 2018 - Programari per a la realització del logotip en format Vectorial.

Sublime Text 3 / Atom - Editor de text.

Durant el desenvolupament del projecte, poden apareixer nous programaris, donat el continu aprenentatge de MEAN i tot allò que l'envolta.

2. CONTINGUTS

2.1 ESTRUCTURALS

Pàgina d'inici.

- Fotografia de fons.
- Situació d'on es troba el visitant. Virtual Books
- Opció de registre o login.

Pàgina d'usuari

- Enllaç amb el nom de l'usuari per a sortir de la sessió.
- Llistat amb les opcions que pot realitzar amb l'aplicació.
- Informació sobre la seva biblioteca.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Registre

- Formulari per a introduir les dades necessàries per a formar part de Virtual Books.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Login

- Formulari per a introduir les dades necessàries per a ingressar a Virtual Books.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Biblioteca

- Taula amb tots els llibres que l'usuari ha donat d'alta.
- Llistat amb les diferents opcions per a ordenar la biblioteca.
- Enllaç amb el nom de l'usuari per a poder sortir de la sessió.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Buscador

- Formulari per a introduir les dades que l'usuari necessiti per a realitzar una recerca dins la seva llibreria.
- Enllaç amb el nom de l'usuari per a poder sortir de la sessió.
- Enllaç amb el nom de la pàgina per a poder tornar a la pàgina d'usuari.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Afegir Llibre

- Formulari per a introduir les dades necessàries per a donar d'alta un llibre a la biblioteca.
- Enllaç amb el nom de l'usuari per a poder sortir de la sessió.
- Enllaç amb el nom de la pàgina per a poder tornar a la pàgina d'usuari.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Gestionar la biblioteca

- Taula amb tots els llibres que l'usuari a introduït a la biblioteca amb botons per editar-los i borrar-los.
- Enllaç amb el nom de l'usuari per a poder sortir de la sessió.
- Enllaç amb el nom de la pàgina per a poder tornar a la pàgina d'usuari.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Borrar llibre

- Missatge de confirmació del borrar del llibre.
- Enllaç amb el nom de l'usuari per a poder sortir de la sessió.
- Enllaç amb el nom de la pàgina per a poder tornar a la pàgina d'usuari.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

2. CONTINGUTS

2.1 ESTRUCTURALS

Edició de llibre

- Missatge informatiu sobre el llibre que s'està editant.
- Formulari amb les dades necessàries a modificar.
- Enllaç amb el nom de l'usuari per a poguer sortir de la sessió.
- Enllaç amb el nom de la pàgina per a poguer tornar a la pàgina d'usuari.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Política de privacitat

- Informació d'interés sobre la política de privacitat a virtual books
- Enllaç amb el nom de l'usuari per a poguer sortir de la sessió.
- Enllaç amb el nom de la pàgina per a poguer tornar a la pàgina d'usuari.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Ajuda

- Què es pot fer a Virtual Books?
- Enllaç amb el nom de l'usuari per a poguer sortir de la sessió.
- Enllaç amb el nom de la pàgina per a poguer tornar a la pàgina d'usuari.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

Sobre Virtual Books

- Informació relativa a Virtual Books.
- Enllaç amb el nom de l'usuari per a poguer sortir de la sessió.
- Enllaç amb el nom de la pàgina per a poguer tornar a la pàgina d'usuari.
- Footer amb enllaços a la política de privacitat, sobre nosaltres i ajuda.

2. CONTINGUTS

2.2 FUNCIONALS

- Registrar i iniciar sessió.
 - Crear un llibre
 - Editar un llibre.
 - Borrar un llibre
 - Mostrar una biblioteca
 - Filtrar sobre els camps de la biblioteca
 - Ordenar els camps d'una biblioteca.
-
- Registrar i iniciar sessió mitjançant Facebook i Twitter.

3. ARQUITECTURA DE L'APLICACIÓ

3.1 CLIENT

Tot i que un principi s'havia evaluat seguir la metodologia MEAN, la part de client s'ha desenvolupat mitjançant HTML5/CSS i el framework Bootstrap 4.

Aquest framework agilitza el procés de treball, i s'ha usat, sobretot, per a organitzar la informació espacial de l'aplicació.

3.2 SERVIDOR

Per la banda de servidor, s'ha usat NodeJS, i la llibreria Express. NodeJS permet treballar amb JavaScript al costat del servidor, i Express s'encarrega de fer amigable tot el procés de l'encaminament i la recollida de dades entre client i servidor.

3.3 BASES DE DADES

Per a la recopilació de dades s'ha fet servir MongoDB, una base de dades no relacional que facilita molt la feina. La informació s'emmagatzema en col·leccions de documents JSON que no han de mantenir una estructura tan ferma com en les bases de dades relacionals (MySQL).

4. PLATAFORMA DE DESENVOLUPAMENT

4.1 SOFTWARE

NodeJS <https://nodejs.org/en>

NodeJS permet l'instal·lació de diferents aplicacions que seràn necessàries per al desenvolupament de Virtual Books mitjançant el seu gestor de paquets npm (Node Package Manager), a més a més de ser el servidor de l'aplicació.

MongoDB <https://www.mongodb.org>

És necessària la seva instal·lació i la posterior posada en marxa per a poder treballar al servidor local.

iTerm2 <https://www.iterm2.com/>

Terminal per a Mac, amb possibilitat de multipestanya, amb un codi de color configurable i autocompletat. Variès millores respecte al terminal per defecte, que fan el treball més àgil.

Atom i Sublime Text 3 <https://atom.io/> <https://www.sublimetext.com/3>

Editors de text. El projecte es va iniciar amb Sublime Text 3, però s'ha acabat amb Atom. Més enllà de cap altra raó, està la novetat d'aprendre nou programari. Atom és molt complet, amb un gestor de paquets, amb el que instal·lar JADE, el processador d'HTML que s'ha fet servir per a Virtual Books. A més a més, té una web molt agradable.

Draw.io <https://www.draw.io>

Eina online per al desenvolupament de Wireframes.

Google Drive <https://drive.google.com>

Emmagatzematge online, fet servir per a còpies de seguretat del projecte, i guardar informació com els vídeos necessàries per a la presentació.

4.2 HARDWARE

Per al desenvolupament de Virtual Books s'ha fet servir un MacBook Pro mid 2012 amb processador Intel Core i5 a 2,5Ghz i 8 GB de RAM i un iMac mid 2010 amb processador Intel Core i7 a 2,93Ghz i 20 GB de Ram.

Altrament, per a la recerca d'informació i la visualització de documents s'han fet servir diferents tauletes, la més usada una Samsung Tab1.

5. PLANIFICACIÓ

5.1 TEMPORAL

El projecte es durà a terme entre Febrer i Juny de 2018, amb previsió de dedicar 2 hores els dies laborables i 3 hores els cap de setmana, sempre que sigui possible. En un projecte llarg sempre hi poden sortir imprevistos.

El total d'hores estimades serà d'unes 15 setmanals, per una previsió total d'unes 260 hores.

Hi ha diverses dates inamovibles, marcades per la incoherència del projecte. Aquestes són les entregues parcials de les PAC, que es detallen a continuació:

ACTIVITAT	DATA
PAC1	06/03/2018
PAC2	04/04/2018
PAC3	13/05/2018
Lliurament Final	11/06/2018
Defensa Virtual	22/06/2018

Taula 1. Dates clau. Entrega de PAC

Tasques, atenent a la distribució d'hores i dies detallada anteriorment:

TASCA	DURACIÓ	INICI	FINAL
PAC1		21/02/2018	06/03/2018
Definició del projecte	12	21/02/2018	04/03/2018
Anàlisi i recerca d'informació	10	21/02/2018	02/02/2018
Pla de treball	1	01/02/2018	02/02/2018
Redacció de la memòria	2	04/02/2018	06/03/2018
PAC2		07/03/2018/	04/04/2018
Definició del Packaging (colors i estètica)	2	07/03/2018	09/03/2018
Definició fitxes dels llibres	2	10/03/2018	12/03/2018
Definició fitxes usuaris	2	13/03/2018	15/03/2018
Usabilitat/UX	2	16/03/2018	18/03/2018
Prototipatge (Wireframes)	11	19/03/2018	30/03/2018
Actualització de la memòria	3	01/04/2018	04/04/2018
PAC3		05/04/2018	13/05/2018
Maquetació web	5	05/04/2018	10/04/2015
Programació FrontEnd	10	11/04/2018	21/04/2018
Base de Dades	8	22/04/2018	30/04/2018
Programació Backend	12	30/04/2018	10/05/2018
Actualització de la memòria	3	10/05/2018	13/05/2018
Lliurament Final		14/05/2018	11/06/2018
Finalitzar projecte	15	14/05/2018	29/05/2018
Finalitzar memòria	5	30/05/2018	03/06/2018
Elaborar presentació (visual-escrita)	2	04/06/2018	06/06/2018
Video per la defensa del projecte	5	07/06/2018	11/06/2018
Defensa Virtual	4	18/06/2018	22/06/2018

Taula 2. Tasques del projecte.

5. PLANIFICACIÓ

5.2 DIAGRAMA DE GANTT



Figura 1. Diagrama de Gantt Virtual Books.

6. PROCÉS DE TREBALL

El procés de desenvolupament de Virtual Books ha estat marcat per dos factors importants. El primer, el nivell d'aprenentatge que s'ha anat aconseguint de la metodologia amb la que s'ha desenvolupat tota l'aplicació i el segon, les dates de les entregues marcades per les diferents PAC.

6.1 PAC 1

Aquesta primera entrega, és l'entrega mental. Pensar i plasmar sobre el paper la definició del projecte, el calendari a seguir, els seus objectius i establir uns mínims als que arribar.

L'objectiu principal d'aquest Treball de fi de grau, a més a més de treure un producte al mercat, amb uns certs mínims viables, era aprendre a desenvolpar-se amb MEAN. Per aquesta raó, tant el nom com la imatge gràfica, s'han deixat una mica de banda.

La planificació temporal establerta des d'un principi, era una planificació fictícia, ja que desconeixia completament el temps que em portaria poguer aprendre cada nou concepte, i molts d'ells eren necessaris per a poguer donar el següent pas. Tampoc es contempla, en un principi cap dia de descans, que al cap i a la fi, resulten claus per a borrar i tornar a posar el cervell a lloc en processos greus d'ofusció i veure la pantalla negra.

6.2 PAC 2

Però, i tot això, ho podré fer?

Si la PAC1 és l'entrega mental, la PAC2 és l'entrega impossible.

Disseny de l'aplicació per a deixar tot llest per a començar a picar el codi.

El prototipatge és la part a la que se li havia pressupost més temps, però gran part d'aquest temps es va dedicar a començar a estudiar i aprendre si el que s'havia plasmat en paper, es podria duir a terme.

És en aquesta fase, on es decideix prescindir d'AngularJS, degut a les dificultats que es tenen amb NodeJS i ExpressJS.

El frontEnd es farà tot amb Bootstrap 4 i HTML5, que ja es coneixen i permeten dedicar tot el temps disponible (i el que no, també), a adquirir els coneixements necessaris per a que els mínims marcats es puguin complir.

6.3 PAC 3

A poc a poc.

Aquesta tercera PAC, si hagués de respondre a un gràfic, la seva representació seria calcada a la que es dibuixa quan hi ha una notícia que fa explotar el Dow Jones.

El paral·lelisme entre anar adquirint els coneixements i anar realitzant el desenvolupament ha evocat a una quantitat d'hores molt superior a la planificada.

Finalment, no s'ha pogut arribar a entregar la versió de l'aplicació amb els mínims desitjats, però si que s'ha avançat el suficient per a tenir la sensació que la fita proposada a la primera PAC era possible.

Amb l'entrega de la PAC, s'adjunta un video on es demostra aquest fet.

6. PROCÉS DE TREBALL

6.4 LLIURAMENT FINAL

En aquesta etapa s'havia d'aconseguir que l'aplicació tingués tots els mecanismes per ser viable, realitzar una revisió a fons de la memòria, realitzar un video de presentació general de Virtual Books, i un altre sobre el Treball de fi de grau. En un mes.

Virtual Books ha assolit totes les funcionalitats desitjades en una primera fase, però queda lluny de ser una aplicació amb un posicionament estable en el mercat.

S'ha aconseguit el que es desitjava, una biblioteca personal, en la que poguer relitzar totes les funcionalitats que aporta un API REST, i que tot funcioni i sigui estable.

Ara bé, en una segona fase, s'hauria d'implementar:

- Possibilitat de gestionar més d'una llibreria per usuari.
- Criteris de recerca a gust del consumidor.
- Sistema per a recordar la contrasenya de l'usuari.
- Evitar la repetició a la bbdd, si dos usuaris tenen el mateix llibre.
- Realitzar un logotip corporatiu.

7. APIS UTILITZADES

7.1 FACEBOOK I TWITTER

Virtual Books està preparada per a realitzar el registre i el login mitjançant Facebook i Twitter. Per a fer-ho s'ha fet servir la funcionalitat de la llibreria PassportJS, que utilitza les APIS de Facebook i Twitter per a gestionar les dades dels usuaris.

En un primer moment, al ser una aplicació per a ser gestionada en local, les funcionalitats no estan habilitades, ja que els serveis de FB i Twitter demanen enllaços vàlids.

```
/**
//register Facebook
passport.use(new FacebookStrategy({
  clientID: '2140061612842019',
  clientSecret: 'bd15280e137af06499e1499d1e1108e7',
  callbackURL: "http://localhost:8088/auth/facebook/callback"
}),
function(accessToken, refreshToken, profile, done){
  User.findOrCreate({facebookId: profile.id},function(err,user){
    if(err)throw(err);
    done(null, user);
  });
});
//register twitter
passport.use(new TwitterStrategy({
  consumerKey: 'twitter api',
  consumerSecret: 'twitter api',
  callbackURL: "http://localhost:8088/auth/twitter/callback"
}),
function(accessToken, refreshToken, profile, done){
  User.findOrCreate({twitterId: profile.id},function(err,user){
    if(err)throw(err);
    done(null, user);
  });
});
});
```

Figura 2. Codi per al registre amb Facebook o Twitter.

8. PERFILS D'USUARI

Hi poden haver dos perfils d'usuaris diferents.

- Usuari No Registrat: Nomès podrà veure les llibreries de les quals tingui un enllaç directe, o que siguin obertes. Que l'usuari propietari hagi decidit que poden ser visibles per a tothom.

- Usuari Registrat: L'usuari principal. Usuari que crea la seva biblioteca, i consulta la dels altres mitjançant els vincles que es puguin crear.

Més endavant, amb la possibilitat de desenvolupar la segona part del projecte, podríem trobar més tipus d'usuaris diferents, que es caracteritzarien per la freqüència d'ús, i el tipus d'ús, principalment de les recerques que puguin fer.

Col·leccionistes, usuaris lectors esporàdics que no vulguin gastar diners en producte nou, o reventes.

8.1 DIAGRAMA UML

S'ha utilitzat un únic actor per a la seva representació:

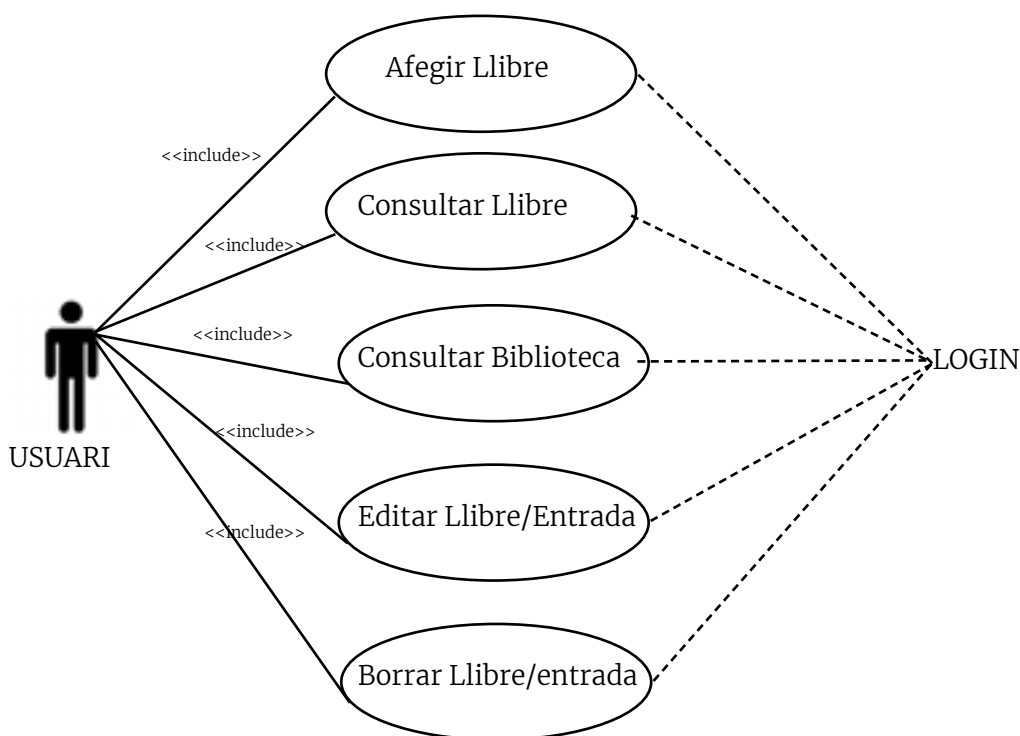


Figura 3. Diagrama UML. Interacció usuari/aplicació.

8. PERFILS D'USUARI

8.2 HISTÒRIES D'USUARI

No hi ha moltes opcions de camins que pugui seguir un usuari. A continuació es detallen algunes que poden esdevenir:

COM A	Usuari potencial.
VULL	Saber de qué va l'aplicació.
PER A	Saber si pot ser del meu interès, i convertir-me en usuari habitual.
CRITERIS D'ACCEPTACIÓ	Verificar que les pàgines d'informació en les que no és necessari el registre (footer), funcionen correctament i mostren la informació necessària per a que serveixi de reclam.
COM A	Usuari.
VULL	Accedir a l'aplicació mitjançant Twitter o Facebook
PER A	Estalviar-se haver de recordar les credencials d'acreditació.
CRITERIS D'ACCEPTACIÓ	Verificar que la llibreria PassportJS funcioni de manera correcte.
COM A	Usuari registrat i loguejat
VULL	Obtenir un llistat de tots els llibres que tinc donats d'alta.
PER A	Tenir un control de la meva biblioteca.
CRITERIS D'ACCEPTACIÓ	Verificar que les pàgines d'informació en les que no és necessari el registre (footer), funcionen correctament i mostren la informació necessària per a que serveixi de reclam.
COM A	Usuari registrat i loguejat.
VULL	Ampliar la meva biblioteca
PER A	Afegir els llibres que vaig obtenint al llarg del temps
CRITERIS D'ACCEPTACIÓ	Verificar que quan s'ompli el formulari de Nou llibre i es cliqui al botó Guardar, es registre el nou llibre a la base de dades.
COM A	Usuari registrat i loguejat.
VULL	Marcar que se m'ha retornat un llibre que tenia deixat
PER A	Tenir la biblioteca actualitzada
CRITERIS D'ACCEPTACIÓ	Verificar que quan s'ompli el formulari d'edició del llibre i es cliqui al botó Actualitzar, el llibre quedi degudament editat a la base de dades.
COM A	Usuari registrat i loguejat.
VULL	Trobar el llibre per a aquest moment, que tinc poc temps per a llegir
PER A	Adaptar les meves circumstàncies actuals a una lectura adient
CRITERIS D'ACCEPTACIÓ	Verificar que quan es cliqui sobre la pestanya de nombre de pàgines dins la biblioteca, es mostri un ordre adequat al criteri de recerca (en aquest cas, per nombre de pàgines).

8. PERFILS D'USUARI

8.3 CASOS D'ÚS

CAS D'ÚS	Publicar un llibre
RESUM DE LA FUNCIONALITAT	Permet a l'usuari publicar un llibre a la seva biblioteca.
PRECONDICIONS	L'usuari ha d'estar registrat i loguejat.
ESCENARI PRINCIPAL D'ÈXIT	<ol style="list-style-type: none"> 1. L'usuari prem el botó de Afegir Llibre. 2. L'usuari omple el formulari per afegir el llibre i prem el botó de confirmació. 3. Les dades del llibre es desen a la base de dades. 4. L'usuari veu un missatge de confirmació.
ESCENARIS ALTERNATIUS	<ol style="list-style-type: none"> 1. Les dades introduïdes per l'usuari no són vàlides per: <ul style="list-style-type: none"> - El llibre ja ha estat introduït. - El format d'alguna de les dades no és correcte.
CAS D'ÚS	Donar d'alta un usuari
RESUM DE LA FUNCIONALITAT	Permet a un usuari poguer fer servir l'aplicació
PRECONDICIONS	L'usuari ha de tenir interès per l'aplicació
ESCENARI PRINCIPAL D'ÈXIT	<ol style="list-style-type: none"> 1. L'usuari prem el botó de registre. 2. L'usuari omple el formulari per donar-se d'alta i prem el botó de confirmació. 3. Les dades de l'usuari es desen a la base de dades. 4. L'usuari veu un missatge de confirmació.
ESCENARIS ALTERNATIUS	<ol style="list-style-type: none"> 1. Les dades introduïdes per l'usuari no són vàlides per: <ul style="list-style-type: none"> - L'usuari ja està registrat - El format d'alguna de les dades no és correcte.
CAS D'ÚS	Login
RESUM DE LA FUNCIONALITAT	Permet a un usuari poguer fer servir l'aplicació
PRECONDICIONS	L'usuari ha d'estar registrat.
ESCENARI PRINCIPAL D'ÈXIT	<ol style="list-style-type: none"> 1. L'usuari prem el botó de Login. 2. L'usuari omple el formulari ingressar a l'aplicació i prem el botó de confirmació. 3. Les dades d'inici es desen a la base de dades i es cotejen. 4. L'usuari veu un missatge de confirmació.
ESCENARIS ALTERNATIUS	<ol style="list-style-type: none"> 1. Les dades introduïdes per l'usuari no són vàlides per: <ul style="list-style-type: none"> - L'usuari no ha estat donat d'alta previament. - El format d'alguna de les dades no és correcte.

8. PERFILS D'USUARI

CAS D'ÚS	Sortir de l'aplicació
RESUM DE LA FUNCIONALITAT	Permet a l'usuari tancar la seva sessió.
PRECONDICIONS	L'usuari ha d'estar registrat i loguejat.
ESCENARI PRINCIPAL D'ÈXIT	1. L'usuari prem el botó de Logout. 2. L'usuari veu un missatge de confirmació.
ESCENARIS ALTERNATIUS	1. Les dades introduïdes per l'usuari no són vàlides per: - El llibre ja ha estat introduït. - El format d'alguna de les dades no és correcte.
CAS D'ÚS	Borrar un llibre
RESUM DE LA FUNCIONALITAT	L'usuari ha d'estar registrat, loguejat i tenir registres a la base de dades.
PRECONDICIONS	L'usuari ha de tenir algun registre a la base de dades.
ESCENARI PRINCIPAL D'ÈXIT	1. L'usuari prem el botó d'eliminar un llibre. 2. L'usuari confirma l'acció mitjançant una contrassenya. 3. Les dades del llibre s'esborren de la base de dades. 4. L'usuari veu un missatge de confirmació.
ESCENARIS ALTERNATIUS	1. Les dades introduïdes per l'usuari no són vàlides per: - La contrassenya és incorrecta.
CAS D'ÚS	Buscar un llibre
RESUM DE LA FUNCIONALITAT	Permet a l'usuari fer una filtratge de la seva base de dades.
PRECONDICIONS	L'usuari ha d'estar registrat, loguejat i tenir registres a la base de dades.
ESCENARI PRINCIPAL D'ÈXIT	1. L'usuari introdueix els registres que vulgui al formulari de filtratge. 2. L'usuari prem el botó de confirmació. 3. Es mostren les entrades relacionades amb el filtratge de l'usuari.
ESCENARIS ALTERNATIUS	1. Les dades introduïdes per l'usuari no són vàlides per: - No hi ha registres que coincideixin amb el que l'usuari busca - El format d'alguna de les dades no és correcte.
CAS D'ÚS	Editar un llibre
RESUM DE LA FUNCIONALITAT	Permet a l'usuari editar les dades d'una entrada.
PRECONDICIONS	L'usuari ha d'estar registrat, loguejat i tenir registres a la base de dades.
ESCENARI PRINCIPAL D'ÈXIT	1. L'usuari prem el botó d'Editar llibre. 2. L'usuari omple el formulari per a rectificar les dades de l'entrada. 3. Les noves dades del llibre prescriuen les anteriors i es desen a la base de dades. 4. L'usuari veu un missatge de confirmació.
ESCENARIS ALTERNATIUS	1. Les dades introduïdes per l'usuari no són vàlides per: - El format d'alguna de les dades no és correcte.

9. WIREFRAMES

Relació dels wireframes de les pantalles principals.
Formulari de recerca d'un llibre. Tots els formularis estan tallats sota el mateix patró. Es pot observar també, l'estructura base de disseny de Virtual Books.

The wireframe shows a search interface for a virtual library. At the top left is the text 'VIRTUAL BOOKS' and at the top right is 'SORTIR'. The main title 'BIBLIOTECA' is centered. Below the title are four menu items: 'BIBLIOTECA', 'BUSCAR', 'AFEGIR', and 'GESTIONAR'. A central search form is titled 'BUSCAR' and contains five input fields: 'TITOL', 'AUTOR', 'EDITORIAL', 'Nº PÀGINES', and two smaller fields 'L'HE DEIXAT?' and 'A QUI?'. A 'BUSCAR' button is located at the bottom of the form. At the bottom of the page are three links: 'P.PRIVACITAT', 'SOBRE VB', and 'AJUDA'.

Figura 4. Wireframe Formulari de registre/recerca

9. WIREFRAMES

Pàgina de llibreria. Selecció i adició de llibreries.

Aquesta pàgina també servirà com a base per a les fitxes de llibres i l'adició de llibres.

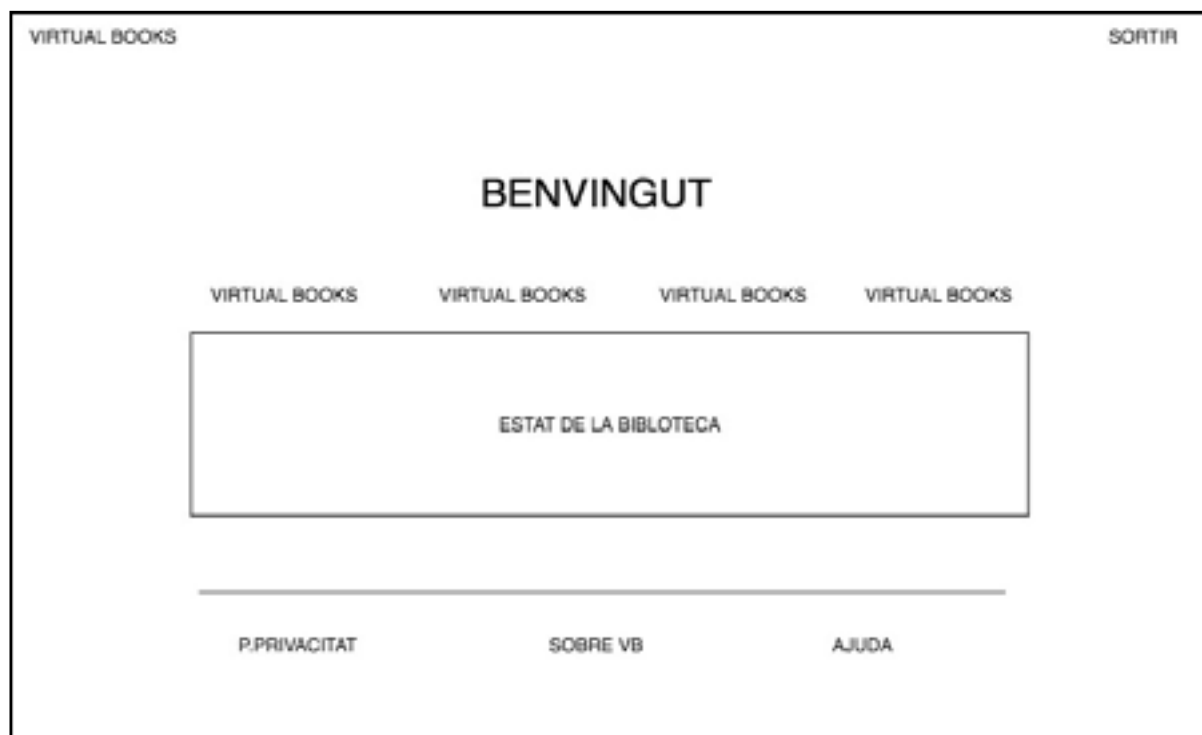


Figura 5. Wireframe de Benvinguda

9. WIREFRAMES

Recerca i organització dins de la llibreria. Mostra de com es visualitzaran els diferents llistats quan siguin necessaris.

The wireframe shows a web page for a virtual library. At the top left is the text 'VIRTUAL BOOKS' and at the top right is 'SORTIR'. The main heading is 'BIBILOTECA'. Below it is a search bar with the label 'BUSCAR' and a table with columns: TITOL, AUTOR, EDITORIAL, LLEGITS, and PÀGINES. The main content is a table with the following data:

Títol	Autor	Editorial	Llegit	Nº Pàgines	Temàtica
Títol 1	John	Blackie Books	Si	325	Esports
Títol 2	Mary	Contra	No	325	Novela
Títol 3	James	Malpaso	Si	325	Esports

At the bottom, there are three links: P.PRIVACITAT, SOBRE VB, and AJUDA.

Figura 6. Organització de la llibreria

10. USABILITAT / UX

La premisa principal per al desenvolupament de l'aplicació ha estat, per sobre de tot, la seva facilitat d'ús. El Disseny Centrat en l'Usuari (DCU) ^{B1} és una metodologia de disseny que pren com a centre els usuaris del producte. La internacional Organization for Standardization (ISO) especifica en la seva norma ISO 13407 quatre fases a seguir en el seu procés:

- **Analitzar i entendre el context d'ús:** Identificar les persones a les quals es dirigeix el producte, i identificar el seu ús i les condicions en el que el faran servir.
- **Especificar els requisits:** Identificar els objectius dels usuaris i dels proveïdor del producte que s'hauràn de satisfer.
- **Aportar solucions de disseny.**
- **Avaluació:** Validar les solucions de disseny i detectar els problemes d'usabilitat mitjançant test d'usuari.

Per avaluar la usabilitat de l'aplicació, he seguit els principis heurístics proposats per Jakob Nielsen.

- **Visibilitat de l'estat del sistema:** *El lloc web o aplicació ha de mantenir sempre informat l'usuari del que està ocorrent i proporcionar-li una resposta en un temps raonable.*

No hi ha masses processos que puguin donar error o puguin demorar molt en la seva execució, però en cas que es produeixi un error, sempre apareixerà un missatge per pantalla, perquè l'usuari sàpiga en tot moment que si no s'ha processat correctament la seva petició, la raó per la que ha estat.

- **Adequació entre el sistema i el món real:** *El lloc web o aplicació ha d'utilitzar el llenguatge de l'usuari, amb expressions i paraules que li resultin familiars.*

No s'utilitzen tecnicismes que a l'usuari li puguin resultar desconeguts i es fa servir la segona persona del singular en totes les interaccions entre l'aplicació i l'usuari. Totes les interaccions de l'aplicació amb l'usuari són ordres clares amb frases curtes o fins i tot paraules.

- **Llibertat i control per part de l'usuari:** *En cas d'elegir alguna opció del lloc web o aplicació per error, l'usuari ha de disposar d'una "sortia d'emergència" clarament delimitada per a abandonar l'estat no volgut en què es troba sense haver de mantenir un diàleg amb el lloc o aplicació. També ha de disposar de la capacitat de desfer o repetir una acció duta a terme.*

La facilitat d'ús amb que l'aplicació ha estat desenvolupada rau en que l'usuari tingui poques opcions d'equivocar-se, però en el cas de que succeeixi a la capçalera sempre hi trobarà el logo de l'aplicació i l'opció del logout, que el permetran tornar a l'inici o sortir de la sessió immediatament.

- **Consistència i estàndards:** *Els usuaris no cal que sàpiguen que diferents paraules, situacions o accions signifiquen el mateix. És convenient seguir convencions.*

Tant el vocabulari com els elements d'interacció mantenen uns estàndards que permeten a l'usuari conèixer en tot moment el que pot i no pot fer. Tots els botons i els enllaços estan dissenyats de la mateixa manera per a que l'usuari pugui saber en tot moment que són clicables.

- **Prevenió d'errors:** *És important prevenir l'existència d'errors mitjançant un disseny adequat. Tot i així, els missatges d'error han d'incloure una confirmació abans d'executar les accions de correcció.*

Les pantalles de confirmació i els warnings poden ser molestos, però més molest pot arribar a ser eliminar una entrada a la base de dades. En el moment d'editar o eliminar un registre, sempre es demanarà confirmació, mitjançant contrasenya.

- **Reconeixement abans que record:** Fer visibles objectes, accions i opcions perquè l'usuari calgui que recordi informació entre diferents seccions o parts del lloc web o aplicació. Les instruccions d'ús han de ser visibles o fàcilment localitzables.

Les passes a seguir són clares. Registre, o Login en cas de que ja es sigui usuari, i a partir d'aquí, seguir les úniques quatre opcions que dona l'aplicació. Buscar, administrar, visionar o afegir.

- **Flexibilitat i eficiència en l'ús:** Els acceleradors o dreceres de teclat poden fer més ràpida la interacció per a usuaris experts, de tal manera que el lloc web o aplicació sigui útil per a usuaris no vells com avançats.

Shortcuts amb acceskey.

- **Disseny i estètica minimalista:** Les pàgines no han de contenir informació irrellevant o innecessària. Cada informació extra competeix amb la informació rellevant i en disminueix la visibilitat.

Tota la informació que apareix en tot moment és indispensable. L'aplicació ha estat concebuda partint d'un disseny minimalista basat en Material Design² i Flat Design³.

- **Ajuda als usuaris a reconèixer i diagnosticar els errors i recuperar-se'n:** Els missatges d'error s'han d'expressar en un llenguatge comú i senzill, que indiqui amb precisió el problema i sugereixi les possibles alternatives o solucions.

Tots els missatges d'error estàn redactats i s'emeten en un llenguatge que l'usuari pugui entendre i es trobi situat en tot moment. Els formularis mostren errors directes tipus : contrasenya incorrecta o usuari no vàlid.

- **Ajuda i documentació:** Encara que és millor que el lloc web o aplicació pugui ser usat sense documentació, pot ser necessari proveir cert tipus d'ajuda. En aquest cas, l'ajuda ha de ser fàcil de localitzar, ha d'especificar els passos necessaris i no ha de ser gaire extensa.

L'aplicaió consta d'un menú d'ajuda on es poden trobar les instruccions d'ús i les solucions als principals problemes que puguin surgir.

²Material Design. <https://material.io/design/>

³Flat Design. <https://www.departamentodeinternet.com/que-es-flat-design-diseno-plano/>

11. SEURETAT

11.1 AUTENTICACIÓ DELS USUARIS

La informació personal sempre és un tema que s'ha de tractar amb cura. Cal assegurar la seva seguretat. A Virtual Books s'ha fet servir el *middleware* d'Express *Passport*¹, que s'encarrega d'administrar les sessions d'usuari i no permetre accessos si no es té l'autorització pertinent. També es fa servir *bcrypt*² per a encriptar les contrasenyes.

Passport també permet l'autenticació d'usuaris mitjançant les xarxes socials, utilitzant les dades que l'usuari ha fet servir a Twitter i Facebook, entre d'altres.

11.2 'USE STRICT'

Les bones pràctiques a l'hora d'escriure codi són necessàries, i l'ús de 'use strict' és una d'elles. Fa referència al mode estricte de Javascript.

'Use strict' aporta una coherència i evita males pràctiques a l'hora de programar, obligant a que el codi compleixi uns requeriments i aplicant unes restriccions, com per exemple la obligatorietat de declarar variables.

12. FITXES

12.1 FITXA DELS LLIBRES

Per a la elaboració de la fitxa dels llibres s'han tingut en compte diversos aspectes. Omplir moltes dades acostuma a fer mandra i acaba desenvocant en un desús del producte. Per tant la primera premisa a tractar ha estat la obligatorietat dels camps a omplir. Per a decidir els obligatoris, m'he basat en en la informació que brinden dels seus llibres tres editorials: *Contra*¹, *Blackie Books*² i *Malpaso*³ :

- Títol*
- Autor*
- Editorial*
- Edició
- N^o de pàgines

A aquests mínims, s'han d'afegir algunes característiques pròpies del llibre en concret del propietari, que serviran, sobretot, per a la segona part del projecte, on es posi en marxa l'intercanvi de llibres i el mercat de segona mà:

- Estat del llibre*
- Estat de la coberta*
- Signat per l'autor*
- Portada il·lustrada per

Un cop establerts els camps obligatoris, l'usuari podrà entrar valors subjectius, que no han de ser públics, però poden enriquir l'experiència.

- Sensacions del llibre
- Valoració del llibre
- Observacions

A més a més de les dades referents al llibre directament, hi haurà un camp adicional, que farà referència a la possessió del mateix.

- Qui el té? L'he deixat a algú?

12.2 FITXA DELS USUARIS

Les fitxes dels usuaris tindran els següents camps:

- Nom
- Cognom
- Email
- Nom d'usuari
- Contrassenya

^{1,2,3} Pàgines web dels editorials. BlackieBooks.org, Malpaso.es, EditorialContra.com

13. INSTAL·LACIÓ

13.1 REQUISITS D'INSTAL·LACIÓ

Per a fer córrer Virtual Books en local, és necessari tenir instal·lat NodeJS¹ i MongoDB². Per a fer-ho, s'ha d'anar a les seves pàgines oficials, i seguir els passos que allà es detallen.

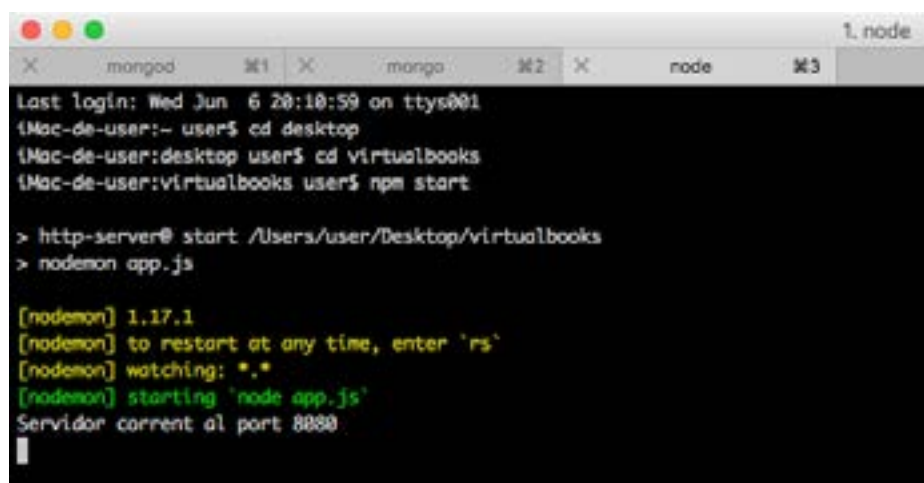
No és necessari cap altre requisit.

13.2 INSTRUCCIONS D'INSTAL·LACIÓ

Un cop instal·lats MongoDB i NodeJS, amb el terminal s'ha d'accedir a la carpeta on es troba el projecte, i executar l'ordre `- npm install -`. D'aquesta manera, s'instal·laran tots els mòduls i les llibreries que són necessàries, que es troben a l'arxiu package.json.

Un cop finalitzada l'aplicació (la consola dona la confirmació), s'ha de posar en marxa el servei de mongoDB. Per a fer-ho, en una altra vista del terminal, s'ha d'executar, dins de la carpeta on tenim el projecte l'ordre `- mongod -`. A continuació, en una vista diferent de les dues anteriors, i també dins de la carpeta del projecte, s'ha d'executar l'ordre `- mongo -`.

En aquest punt ja està tot el sistema preparat per a fer córrer l'aplicació, i nomès serà necessari la seva posada en marxa. Per a fer-ho, a la mateixa vista que s'ha executat `- npm install -`, s'ha d'executar l'ordre `- npm start -`.



```

Last login: Wed Jun 6 20:10:59 on ttys001
(Mac-de-user:~ user$ cd desktop
(Mac-de-user:desktop user$ cd virtualbooks
(Mac-de-user:virtualbooks user$ npm start

> http-server@ start /Users/user/Desktop/virtualbooks
> nodemon app.js

[nodemon] 1.17.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node app.js'
Servidor corrent al port 8080
  
```

Figura 7. Ready to start. El servidor ja corre al port 8080.

El servidor local ja està llest per a servir l'aplicació mitjançant <http://localhost:8080/>

Per a realitzar totes les proves necessàries, hi ha un usuari creat:

```

user: user@uoc.com
pass: useruoc
  
```

De tota manera, al realitzar un registre, ja es poden fer servir les dades que s'hagin dipositat.

14. BUGS

El fet de treballar mentre s'aprenia, ha comportat que la llista d'errors que anaven apareixent a mesura que s'anava desenvolupant l'aplicació no tingui fi.

El principal problema, ha estat el domini de nodeJS i ExpressJS, que permetés solucionar qualsevol problema senzill.

Un dels problemes més grossos que he trobat, ha estat el lligar les col·leccions d'usuaris i llibres, de tal manera que cada usuari visualitzés únicament la seva biblioteca, i no la biblioteca global.

Desde un principi tenia clar que hi havia d'haver alguna llibreria que s'encarregués de fer-ho, ja que sembla ser que hi ha llibreries per tot.

Finalment, es va solventar amb un mecanisme senzill, consistent en introduir un camp en la col·lecció dels llibres que guardés el nom d'usuari de l'usuari (podria haver estat qualsevol camp identificatiu), i posar un condicional a l'hora de mostrar la biblioteca de cadascun que comprovés si l'usuari que tenia la sessió activa coincidia amb l'usuari que havia creat el llibre.

Al cap i a la fi, una solució que poc tenia a veure amb NodeJS.

Ha estat una solució parcial, ja que de ben segur es pot millor, fent la base de dades més eficient, evitant duplicitats (per exemple, amb l'ISBN, identificatiu únic de cada llibre).

La decisió sobre els bugs que han anat sorgint ha seguit el següent llistat de prioritats:

1. És necessari per al funcionament de l'aplicació?
2. És qüestió de disseny o de funcionament?
3. Conec una alternativa en el cas de no trobar la solució que busco?
4. Tinc temps per aplicar-ho, sense deixar de banda altres funcionalitats?

Així, per exemple, per a solucionar que el buscador diferenciés entre |SI| |Si| |sI| |si| (o les mateixes opcions amb no), a l'hora de filtrar per Llegits/No llegits, s'ha optat per l'opció d'inserir al HTML/JADE la funció onkeyup, que s'activa quan es detecta que s'ha deixat d'escriure alguna cosa i passar-li que totes les lletres es convertissin en majúscula.

```
div(class="form-item")
  label(for="read")
  input(type="search" name="read" placeholder="L'he llegit? Si/No" onkeyup="this.value=this.value.toUpperCase()")
```

Figura 8. Fragment de codi on es troba la funció onkeyup.

15. PROJECCIÓ A FUTUR

És possiblement l'atractiu més important d'haver realitzat aquesta aplicació. El temps per a desenvolupar un projecte d'aquestes característiques no és potser el més adequat per a engrescar-se a fer-lo amb unes eines desconegudes, que encara delimiten més el temps productiu.

Així doncs, la intenció es dedicar-li més temps a aquest projecte, amb les següents funcionalitats:

- Desplegar l'aplicació a internet, activant així el registre mitjançant Twitter i Facebook.
- Administrar diverses biblioteques per usuari.
- Solucionar els registres de llibres doblats a la BBDD, filtrant per ISBN.
- Potenciar el buscador, amb la possibilitat de que l'usuari pugui filtrar per camps propis de recerca.
- Descarregar llistat de llibres, en format pdf.
- Crear una pàgina d'estadístiques, com per exemple els llibres que es tenen de cada editorial, els llegits/no llegits...
- Enllaçar els llibres, un cop creats, amb la pàgina dels editorials, si en té, per a poder veure la informació allà detallada.
- Incloure suggeriments, com ara si d'una pel·lícula s'ha realitzat un llibre.
- Puntuar els llibres i així poder obtenir recomanacions segons les puntuacions.
- Fer un refactoring del codi.
- Dedicar-li temps al SEO.
- Incloure avís de Cookies.
- Desplegar una plataforma de venda i intercanvi de llibre, mitjançant la biblioteca de cada usuari. Cada usuari podrà marcar quins llibres vol posar en venda o intercanvi, i li apareixeran a la resta d'usuaris que estiguin interessats en fer servir aquesta part de l'aplicació, que seria una espècie de Wallapop de llibres.

16. ESTUDI DE VIABILITAT

16.1 PRESSUPOST

Per a la realització d'aquest projecte, traslladat a un projecte real, serien necessaris els següents perfils professionals, partint de la base que el projecte ja ha estat venut i acceptat:

- **Cap de projecte**, encarregat d'establir un equip, establir les tasques a realitzar, i realitzar la coordinació de totes les àrees que hi formen part.
- **Dissenyador Gràfic**, per al disseny del logotip, el disseny de l'aplicació i establir una guia d'estil.
- **Full-stack developer**, encarregat de la part tècnica de l'aplicació. En comunicació amb els altres departaments, decidir quines són les tecnologies més adients per al seu desenvolupament, i encarregar-se del front-end i el back-end. Si el projecte creix segons les expectatives realitzades, és possible que el més adient sigui contractar un especialista en back-end.

A partir d'aquest punt, per a un bon tracte amb el client, s'han d'establir una sèrie de condicions per a que el tracte sigui satisfactori.

- **Abast del projecte**, és a dir, fins a on queda cobert el projecte dins el pacte econòmic al que s'ha arribat. A partir d'aquí, tot el que es vulgui s'haurà de negociar a part.
- **Manteniment**, en mesura temporal. Temps que el client podrà tenir manteniment i el cost d'aquest. Així com qué entra dins el manteniment.

Aquestes serien les característiques que s'haurien de donar en un projecte professional, però que, donat a la tipologia del mateix, no es compleixen. El cap de projecte, el dissenyador gràfic i el full-stack developer recauen en la mateixa persona, així que el preu hora, cau en una mitja, i obviament, tira a l'alça per a compensar totes aquelles hores que no estan pressupostades però que si que s'hi dediquen. Així doncs, per una previsió de 260 hores, amb un preu de **30euros/hora**, dona un total de **7.800 euros** bruts, a falta de sumar-hi el manteniment.

El manteniment, és adient tipificar-lo també en hores, per vèries raons. La primera, es perquè el temps dedicat, també es marca en hores, de la mateixa manera que el temps dedicat al desenvolupament en si. La segona, i més important, es que si les hores es paguen, s'eviten trucades que es puguin respondre amb el tan estimat '*has mirat que x estigui encés?*'. Així doncs, un manteniment de per vida, amb el preu de les hores a 35euros/hora, a no ser que el problema sigui demostrable que no és del client.

A partir d'aquí, s'hauria de contractar un domini i un hosting, que en aquest cas, a **one.com**, queda tot per **40 euros anuals**. Es tria one.com per fiabilitat i bon servei postvenda.

16. ESTUDI DE VIABILITAT

16.2 ANÀLISI DE MERCAT

En aquesta primera fase del projecte, que és la que abarca el treball de fi de grau especificat en aquesta memòria, l'aplicació tractarà de donar resposta a la Biblioteca virtual individual. En aquesta camp, la competència amb la que ens podem trobar, és bàsicament els gustos i tradicions que hom pugui tenir ja interioritzats.

L'ús de fulles de càlcul, llibretes, l'amor per l'"analògic".

És en una segona fase, on la competència entra en joc.

En un primer esglaó, es troben les grans plataformes de venda de segona mà entre usuaris, com *Vibbo*¹, *Wallapop*² i en menor mesura *milanuncios*³.

Virtual Books no té com a objectiu convertir-se en referència de venda i intercanvi de llibres de segona mà, però si arribar a ser la plataforma més important en el camp exclusiu de particulars aficionats a la lectura.

Hi ha plataformes com *iberlibro*⁴, *libleista*⁵ o *uniliber*⁶, que ofereixen llibre de segona mà, però provinent de llibreries.

Totes aquestes plataformes tenen en comú que ofereixen un perfil molt detallat del llibre, amb tota la informació que Virtual Books ofereix guardar a la biblioteca individual.

Virtual Books ofereix aquesta possibilitat, i afegeix la facilitat de Wallapop per a cercar i poguer vendre o intercanviar el llibre desitjat. Fins i tot, va un pas més enllà, en el que hauria de dictar la lògica, on posar el títol del llibre al buscador hauria de ser suficient per a que totes les entrades resultessin llistades.

^{1,2,3} Pàgines de venda de producte de segona mà entre usuaris. Vibbo.com, wallapop.es, milanuncios.es

^{4,5,6} Pàgines exclusives de venda de llibre nou i de segona mà, provinent de llibreries. iberlibro.com, libelista.com, uniliber.es

16. ESTUDI DE VIABILITAT

16.1. CONCLUSIONS

El projecte és completament viable i es demostra amb el fet de que ja és una realitat. Escrit això, completament necessari per a l'autoestima del creador, Virtual Books és una plataforma que de ben segur pot ser útil, en aquesta primera fase embrionaria.

No és una plataforma que sigui rentable, ja que no admet publicitat per a ser atractiva. Té una arquitectura minimalista, molt clara, en la qual la publicitat aniria en un sentit diametralment oposat.

En una segona fase, on les prestacions ja s'incrementarien, i es traslladaria de forma indefinida a terminals mòbils, s'hauria d'estudiar la seva rendibilitat a través de la publicitat, però no seria descartable, en format no abusiu, com per exemple la que podem trobar a Wallapop, on editorials o llibreries podrien apostar per promocions personalitzades segons les preferències dels lectors.

El cost total, es podria donar ja per pagat, ja que l'única manera de recuperar les hores invertides seria vendre l'explotació, i quedar-se amb els beneficis de manteniment.

17. CONCLUSIONS

Els objectius principals que m'havia marcat el dia que vaig decidir engegar aquest projecte, eren aprendre a desenvolupar-me en un entorn desconegut per mi, amb una base no excessivament sòlida de Javascript, i aconseguir un producte que pogués ser útil per a algú més que per mi.

A la baralla entre temes a escollir, hi havia un denominador comú. Assolir algun coneixement nou que em deixés amb ganes de més, ja que el tfg permetia començar un camí en que la etapa final no seria la darrera.

I així ha estat. He descobert que allò que buscava amb l'aprenentatge de **NodeJS** i **MongoDB** s'ha complert, i per el camí he trobat **Jade**, un postprocessador d'HTML que m'ha semblat molt potent i molt útil per a mantenir un codi ordenat i elegant.

He comprovat que la majoria d'idees que jo tenia, i que partien d'inici en la carrera de convertir-se en Treball de fi de grau, es poden desenvolupar amb aquesta tecnologia.

A nivell temporal, he trobat a faltar molt de temps. Molt de temps d'assentament del projecte, de poder fer-lo créixer dins l'entorn del TFG, amb un suport universitari si fos necessari. He trobat a faltar molt de temps que trobo que hauria de ser necessari per a realitzar un TFG d'aquestes característiques, que penso que es pot anar alimentant amb el pas del temps.

Per a mi, un tfg ideal, dins aquest àmbit, hauria de tenir la durada mínima de dos semestres, i poder compaginar-se amb diferents assignatures, que en permetessin la retroalimentació però que a la vegada ajudessin a evadir-se en moments de col·lapse, que no han estat pocs.

És possible, però, que aquesta sensació sigui produïda per a voler avançar amb dues tasques alhora: el treball i l'aprenentatge.

En sóc conscient, és clar, que els marges temporals adequen un projecte a unes circumstàncies laborals que cada vegada més, demanden uns temps d'execució impossibles per projectes d'aquestes característiques.

A l'hora d'afrontar el projecte, el que més em va repetir el meu voltant, és que era molt important establir i tancar les tasques i l'abast del projecte desde un principi.

I ha resultat clau, davant dels dubtes inicials, definir fins a quina fase havia d'arribar el projecte dins del TFG. Va ser una dels maldecaps més importants, sobre tot perquè en cap moment veia suficient el que podia establir com a límit

Finalment, s'entrega una versió que permet substituir qualsevol suport o '*manera*' que cadascú tingui per a gestionar la seva biblioteca, l'altre dels objectius principals juntament amb l'aprenentatge.

18. ANNEXOS

18.1. LLIURABLES DEL PROJECTE

El lliurament final consta de:

- Memòria del projecte, amb tota la informació referent a l'aplicació desenvolupada.
- Codi font de l'aplicació, en una carpeta amb el nom del projecte, VirtualBooks.
- Instruccions de funcionament, en un arxiu .html
- Vídeo de presentació del projecte, amb demostració de funcionament del mateix.
- Vídeo de presentació del projecte per al públic en general.

18. ANNEXOS

18.2 CODI FONT

Virtual Books està basat en rutes **GET** i **POST**. Les rutes **GET** permeten obtenir respostes del servidor, i s'usen per a servir les pàgines i les dades dels formularis, i les rutes **POST** donen la possibilitat d'inserir dades a la base de dades.

En l'extracte de codi següent, es poden veure diverses peticions **GET** que serveixen diferents pàgines estàtiques. Quan es demana un enllaç, es serveix el desitjat mitjançant l'ordre `res.render`, i se li passen els paràmetres a demanda, en aquest cas, `name`, que retorna el nom de l'usuari per a poder mostrar-lo en la part de client.

```
49 app.get("/books/new", function(req, res){
50   res.render("books/new", {
51     name: req.user.name
52   });
53 });
54
55 app.get("/books/search", function(req, res){
56   res.render("books/search", {
57     name: req.user.name
58   });
59 });
60 app.get("/footer/about", function(req, res){
61   res.render("footer/about", {
62     name: req.user.name
63   });
64 });
```

El paràmetre `name`, que s'ha demanat retornar mitjançant el mètode `res.render`, es passa a jade de la següent manera:

```
h1(class="fjalla") Benvingut,
  span(class="myblue") #{name}
  | !
```

18. ANNEXOS

18.2 CODI FONT

El mètode GET, a més a més de servir pàgines estàtiques, també serveix per a editar llibres. Se li passa al mètode l'identificatiu del llibre, que el serveix mongoose, i amb l'ordre findOne de mongoDB, que seria equivalent a un SELECT amb condicions de SQL, es troba el llibre que s'està editant per a actualitzar les dades.

Aquestes dades, han estat introduïdes mitjançant un formulari per POST.

```
219 // EDITAR LLIBRES
220 app.put("/books/:id",function(req,res){
221     var id = req.params.id;
222     if(req.body.confirm == confirm){
223         var data = {
224             title: req.body.title,
225             author: req.body.author,
226             editorial: req.body.editorial,
227             read: req.body.read,
228             pages: req.body.pages,
229             category: req.body.category
230         };
231         Book.update({"_id": id},data,function(book){
232             res.redirect("/books");
233         })
234     }else{
235         res.redirect("/");
236     }
237 });
238
239 app.get("/books/edit/:id",function(req,res){
240     var id_book = req.params.id;
241     //buscar un llibre en concret findOne
242     Book.findOne({_id: id_book},function(error,book){
243         console.log(book);
244         res.render("books/edit",{
245             book: book,
246             name: req.user.name
247         });
248     });
249 });
250
```

18. ANNEXOS

18.2 CODI FONT

El mètode **POST** és l'encarregat d'inserir les dades a la base de dades, mitjançant els formularis. En l'extracte següent, es pot observar com es creen les entrades dels llibres, on una variable tipo array(*data*) reb tots els valors inserits al formulari, i crea un nou objecte *Book*, que les reb, per després guardar l'objecte en una variable i guardar-lo dins la base de dades amb l'ordre de nodeJS **.save**

```
76 //Definir ruta a través de la que es creen els productes
77 app.post("/books", function(req, res){
78     console.log(req.bodyParser);
79     var data = {
80         title: req.body.title,
81         author: req.body.author,
82         editorial: req.body.editorial,
83         read: req.body.read,
84         pages: req.body.pages,
85         category: req.body.category,
86         usuari: req.user.username
87     }
88     var book = new Book(data);
89     book.save(function(err){
90         console.log(book);
91         res.redirect("/books");
92     });
93 });
94
```

18. ANNEXOS

18.2 CODI FONT

Un altre dels extractes que penso que val la pena destacar, és el buscador.

Com que són dades que s'han d'inserir, encara que en aquest cas, no s'han de guardar, es fa servir el mètode **POST**.

Mitjançant un condicional es comprova que l'usuari hagi introduït algun paràmetre de recerca, i així poguer donar un error en el cas de que no ho hagi fet.

A l'altre banda del condicional, si l'usuari ha introduït informació, es realitza una recerca de tot allò que hagi introduït, mitjançant el query **\$or** de mongoDB.

També es pot observar el funcionament de la llibreria **connect-flash**, encarregada de mostrar els missatges d'error a tota l'aplicació. **req.flash** s'encarrega de crear el missatge, i preparar-lo per a que sigui cridat quan sigui necessari.

S'observa també, que el mètode `res.render`, pot passar més d'un paràmetre quan renderitza una pàgina. En aquest cas, el missatge creat per **connect-flash** i el nom de l'usuari que té iniciada sessió en aquesll moment.

```
// Buscador
app.post("/search",function(req,res){
  var editorial = req.body.editorial;
  var title = req.body.title;
  var author = req.body.author;
  var read = req.body.read;
  var pages = req.body.pages;
  var category = req.body.category;
  if(editorial == "" && title == "" && author == "" && read == "" && category == "" && pages == ""){
    req.flash('searchMessage', 'No has introduït res a buscar');
    res.render("books/search", {
      message: req.flash('searchMessage'),
      name: req.user.name
    });
  }else{
    Book.find({ $or: [{'editorial': editorial},{'title': title},{'author': author},{'read': read},{'category': category},{'pages': pages}]
    }, function(error,documents){
      if(error){console.log(error);}
      console.log(documents.length);
      if(documents.length == 0){
        console.log('No hi ha coincidències amb les recerques que has introduït');
        req.flash('noMessage', 'No hi ha coincidències amb les recerques que has introduït');
        res.render("books/search", {
          books: documents,
          name: req.user.name,
          message: req.flash('noMessage')
        });
      }else{
        res.render("books/library", {
          books: documents,
          name: req.user.name
        });
      }
    });
  }
});
```

18. ANNEXOS

18.2 CODI FONT

L'ordre dels llibres, com ha d'obtenir una informació, i el sistema està creat per a demanar un ordre mitjançant un enllaç, no una entrada d'informació, també s'executa mitjançant el mètode **GET**. Fa servir una operació de **READ** del **CRUD** de mongoDB (Create, Read, Update i Delete), que es `.find` per a trobar tots els llibres que pertanyen a l'usuari que en aquell moment està amb la sessió iniciada, i els ordena amb l'operador `.sort`, que té el paràmetre `1`, per a ordenar Aa-Zz i `-1` per a Zz-Aa.

```

133 //Ordenar
134 app.get("/sbyauthor", function(req,res){
135     Book.find({usuari: req.user.username}),function(error,documento){
136         if(error){console.log(error);}
137         res.render("books/library",{
138             books: documento,
139             name: req.user.name,
140             email: req.user.email
141         })
142     }).sort({author:1});
143 });
144 app.get("/sbytitle", function(req,res){
145     Book.find({usuari: req.user.username}),function(error,documento){
146         if(error){console.log(error);}
147         res.render("books/library",{
148             books: documento,
149             name: req.user.name,
150             email: req.user.email
151         })
152     }).sort({title:1});
153 });

```

El document `.jade` conté els enllaços a cada criteri de recerca en un formulari mitjançant **action**.

```


form(method="get" action="/sbytitle")
input(type="submit" class="arvo button" value="Títol")



form(method="get" action="/sbyauthor")
input(type="submit" class="arvo button" value="Autor")



form(method="get" action="/sbyeditorial")
input(type="submit" class="arvo button" value="Editorial")



form(method="get" action="/sbyread")
input(type="submit" class="arvo button" value="Llegits")



form(method="get" action="/sbypages")
input(type="submit" class="arvo button" value="Pàgines")



form(method="get" action="/sbycategory")
input(type="submit" class="arvo button" value="Temàtica")


```

18. ANNEXOS

18.2 CODI FONT

Els llibres es creen amb un camp al Schema, que recull l'usuari que en aquell moment ha iniciat sessió. D'aquesta manera, cada cop que l'usuari fa una recerca, o vol llistar la seva biblioteca, es comprova que l'usuari que ha iniciat sessió, i el llibre a retornar, coincideixen en un camp, que és l'**username**.

```
6  const bookSchema = new mongoose.Schema({
7    title:String,
8    author:String,
9    editorial: String,
10   read: String,
11   pages: Number,
12   category: String,
13   usuari: String,
14   llegir: ["Si", "No"],
15   date: {
16     type: Date,
17     default: Date.now
18   }
19 });
```

```
76 //Definir ruta a través de la que es creen els productes
77 app.post("/books",function(req,res){
78   console.log(req.bodyParser);
79   var data = {
80     title: req.body.title,
81     author: req.body.author,
82     editorial: req.body.editorial,
83     read: req.body.read,
84     pages: req.body.pages,
85     category: req.body.category,
86     usuari: req.user.username
87   }
```

A l'extracte de codi anterior, es pot observar com quan s'assignen els valors que es faràn servir per a després crear el llibre, al camp **usuari**, se li passa l'**username** de l'usuari, que en aquell moment té iniciada la sessió.

18. ANNEXOS

18.3 LLIBRERIES

NodeJS compta amb el gestor de paquets NPM per a gestionar les llibreries necessàries. A continuació es detallen les que s'han fet servir a Virtual Books. A nivell de servidor:

body-parser <https://www.npmjs.com/package/body-parser>

S'encarrega de retornar, mitjançant la variable req.body els objectes enviats per POST.

mongoose <https://www.npmjs.com/package/mongoose>

S'encarrega de connectar NodeJS i la base de dades en mongoDB i gestionar-la.

cookie-parser <https://www.npmjs.com/package/cookie-parser>

S'encarrega de la gestió de cookies.

express <https://www.npmjs.com/package/express>

Framework de NodeJS, que facilita l'interpretació de NodeJS.

passport <https://www.npmjs.com/package/passport>

S'encarrega dels registres i les sessions dels usuaris.

passport-local <https://www.npmjs.com/package/passport-local>

Registre local del middleware Passport, mitjançant usuari i contrasenya.

passport-twitter <https://www.npmjs.com/package/passport-local>

Registre mitjançant Twitter del middleware Passport.

passport-facebook <https://www.npmjs.com/package/passport-local>

Registre mitjançant Facebook del middleware Passport.

connect-flash <https://www.npmjs.com/package/connect-flash>

Gestor de missatges del modul Express.

moment <https://www.npmjs.com/package/moment>

Gestor de dates, fet servir en els registres d'usuari.

A nivell client, s'han fet servir els següents:

Bootstrap 4 <https://getbootstrap.com/>

Llibreria encarregada de maquetar l'aplicació, mitjançant CSS i JS de manera àgil.

18. ANNEXOS

18.4 CAPTURES DE PANTALLA

Login, registre i missatges d'error.



Figures 9 i 10. Captures de pantalla del registre i l'Inici de Sessió.



Figura 11. Missatges d'error.

18. ANNEXOS

18.4 CAPTURES DE PANTALLA

Benvinguda i opcions de la biblioteca personal.



Figura 12 . Pàgina benvinguda usuari.



Figura 13. Biblioteca i quadre d'ordenació.

18. ANNEXOS

18.4 CAPTURES DE PANTALLA

BENINGUT, UOC!

Biblioteca Buscar Afegeix Llibre Compartir

.BUSCAR

Títol del llibre _____

Nom de l'autor _____

Editorial _____

L'ha llegit? (Sí/No) _____

Nº de pàgines _____

Temàtica _____

BUSCAR

POLÍTICA DE PRIVACITAT SORRE VIRTUAL BOOKS AJUDA

© 2018 Copyright: Virtual Books

Figura 14. Buscar un llibre.

BENINGUT, UOC!

Biblioteca Buscar Afegeix Llibre Administrar la biblioteca

.NOU LLIBRE

Nom Del Llibre _____

Nom De L'autor _____

Editorial _____

L'ha llegit? (Sí/No) _____

Nº de pàgines _____

Temàtica (Problema/Esports/Biografia...) _____

GUARDAR

POLÍTICA DE PRIVACITAT SORRE VIRTUAL BOOKS AJUDA

© 2018 Copyright: Virtual Books

Figura 15. Crear un nou llibre.

18. ANNEXOS

18.4 CAPTURES DE PANTALLA

ADMINISTRACIÓ DE LA BIBLIOTECA

[← Tornar a la meua llibreria](#)

Títol	Autor	Editorial	Cuigi	Nº Pàgines	Temàtica	Accions
Prohibido Nacer	Trevor Noah	Blackie Books	30	295	Biografia	Editar - Eliminar
Zorobab	José Plaats	Clavis Textos	81	300	Cuixa	Editar - Eliminar
Intercèl·lular	Franco B. Berger	Alagona	30	433	Novela	Editar - Eliminar
Buena Robótica	José María	Ediciones JC	81	490	Esport	Editar - Eliminar
Cuatro amigos	David Trosch	Alagona	81	295	Novela	Editar - Eliminar
Talles en las gradas	Wick Hensley	Palma de lectura	81	404	Novela	Editar - Eliminar
¿Por qué me comprometo con María Tullio si es tan tonta?	Sara Blakes	Principales de los libros	30	542	Novela	Editar - Eliminar
Los barbones	Santiago Lorenzo	Blackie Books	81	541	Novela	Editar - Eliminar
Las gatas	Santiago Lorenzo	Blackie Books	81	221	Novela	Editar - Eliminar
Los voladores	Santiago Lorenzo	Blackie Books	81	443	Novela	Editar - Eliminar
Fargo Book City	Chuck Kleemann	Di Pop	30	584	Biografia	Editar - Eliminar

[POLÍTICA DE PRIVACITAT](#) [SOBRE VIRTUAL BOOKS](#) [AJUDA](#)

© 2018 Copyright: Virtual Books

Figura 16. Administració de la llibreria.

UOC, ESTÀS EDITANT EL LLIBRE PROHIBIDO NACER

.EDITAR LLIBRE

Prohibido Nacer

Trevor Noah

Blackie Books

NO

30

295

Biografia

Editar aquí: SI/NO

ACTUALITZAR

[POLÍTICA DE PRIVACITAT](#) [SOBRE VIRTUAL BOOKS](#) [AJUDA](#)

© 2018 Copyright: Virtual Books

ESTÀS ELIMINANT

Prohibido Nacer de Trevor Noah de Editorial Blackie Books

.SI, ESTIC D'ACORD

Editar aquí: SI/NO

ELIMINAR

[POLÍTICA DE PRIVACITAT](#) [SOBRE VIRTUAL BOOKS](#) [AJUDA](#)

© 2018 Copyright: Virtual Books

Figures 17 i 18. Editar i borrar un llibre.

18. ANNEXOS

18.5.1 GUIA D'ESTIL - NOTACIONS I CONVENCIONS MEMÒRIA

TÍTOL DE CAPÍTOL : BEBAS NEUE REGULAR, 16PT, RGB(0,0,0).

SUBTÍTOL DE CAPÍTOL: BEBAS NEUE REGULAR, 16PT, RGB(164,167,173).

Cos de text: Merriwather Light, 10pt, RGB(0,0,0).

Cos de text, apartats principals: Merriweather Black, 10pt, RGB(0,0,0).

Cos de text: notacions, literals, noms de marca: Merriweather Light Italic, 10pt, RGB(0,0,0).

Figures i taules: Courier New Regular, 10pt, RGB(0,0,0).

Peus de pàgina: Merriweather Light, 7pt, RGB(0,0,0).

18. GUÍA D'ESTIL

18.5.2 PALETA TIPOGRÀFICA I MIDA DE FONTS - APLICACIÓ

TITULARS PRINCIPALS

Tipus de lletra: fjalla

Color: #000000

<https://fonts.google.com/specimen/Fjalla+One>

COS DE TEXT

Tipus de lletra: arvo

Color: #000000

<https://fonts.google.com/specimen/Arvo>

18.5.3 PALETA DE COLORS - APLICACIÓ

El disseny senzill i minimalista, evoca en una paleta de colors escassa. Els colors principals seràn els següents:

RGB (255,255,255)
#ffffff

RGB (0,0,0)
#000000

RGB (85,171,255)
#55abff
Color secundari

18. GUÍA D'ESTIL

18.5.3 RETÍCULA- APLICACIÓ

Per versatilitat s'ha decidit fer una retícula de 12 columnes. *Bootstrap* està basat en un grid de 12 columnes, que esdevé en una organització de la informació molt còmode i amb molta coherència.

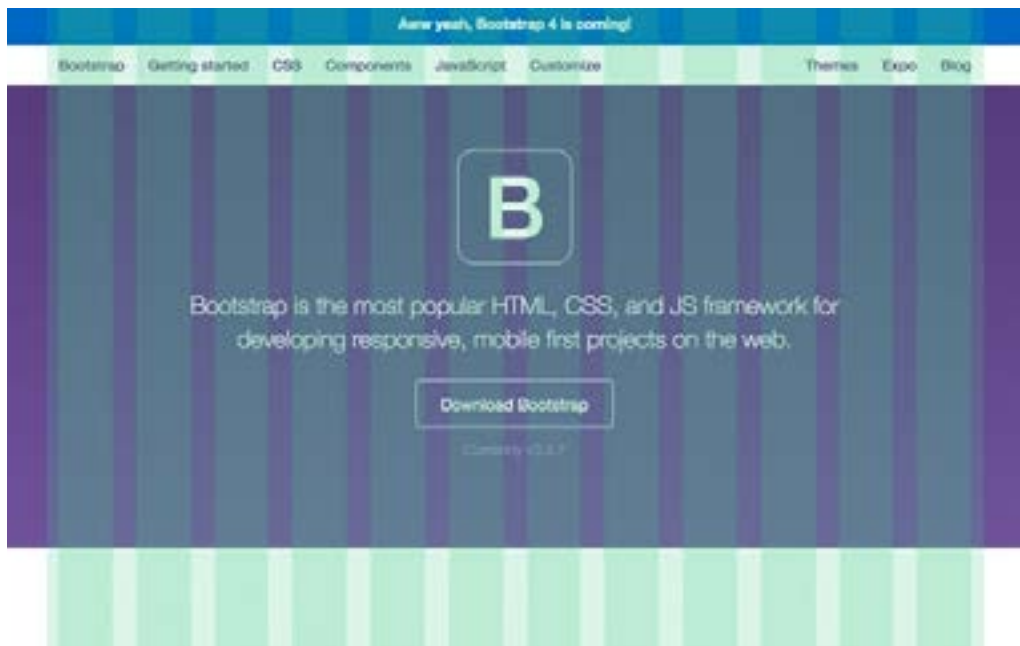


Figura 19. Grid de 12 columnes de Bootstrap sobre la seva

18.5.5 ELEMENTS D'INTERACCIÓ

Botons

Buscar un llibre

BUSCAR UN LLIBRE

Figura 20. Estat dels botons. A l'esquerra normal, a la dreta hover.

18. GUÍA D'ESTIL

18.5.5 ELEMENTS D'INTERACCIÓ

.BUSCAR

Títol del llibre

Nom de l'autor

Editorial

L'he llegit? Si/No

Nº de pàgines

Temàtica

Figura 21. Elements de formulari.

18. ANNEX

18.6 GLOSSARI

API Interfície de programació d'aplicacions.

Middleware Software que ajuda a la interconnexió entre aplicacions

NoSQL Sistema de bases de dades no relacional.

Terminal Consola de comandes per executar ordres.

18. ANNEX

18.7 BIBLIOGRAFIA

UOC BLOG MULTIMEDIA. *Disseny Centrat en l'Usuari (DCU)*. [online]
<http://multimedia.uoc.edu/blogs/dii/disseny-centrat-en-lusuari-dcu/>

UOC. *Presentació i elaboració de documents*. [online]
http://materials.cv.uoc.edu/cdocent/HKZB5UG6XS130_6R5O43.pdf

Hanson, J. *PassportJS documentation*. [online]
<http://github.com/jaredhanson/passport>

18.8 RECURSOS

A més a més de la bibliografia, també s'han fet servir diverses fonts de recursos que han ajudat a entendre i desenvolupar les diferents tecnologies que s'han emprat en el desenvolupament de l'aplicació, així com a mitigar i solucionar els diferents errors que han anat sorgint.

codigofacilito <https://codigofacilito.com/>

Videotutorials en castellà.

StackOverflow <https://stackoverflow.com/>

Fórum de desenvolupadors. Bíblia de la solució de problemes.

MongoDB <https://www.mongodb.com/>

Documentació de MongoDB

Bootstrap <https://getbootstrap.com/>

Documentació de Bootstrap.

NPM <https://www.npmjs.com>

Repositori de llibreries d'NPM.

w3schools <https://www.w3schools.com/>

Referències sobre HTML5, CSS3, Javascript, Bootstrap 4...

Unspalsh <https://unsplash.com/>

Excel·lent repositori d'imatges lliures de drets.

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

Documentació de NodeJS

ExpressJS <http://expressjs.com/es/>

Documentació de NodeJS