



IES PAZO DA MERCÉ

*Ciclo Formativo de Grao Superior en Desenvolvemento
de Aplicacións Web.*

Bookbag

Proxecto presentado por:	Tomás Moroño Souto
Titor/a:	Héctor Gómez Vallejo
Data:	9 / 12 / 2025

Resumo

Na década dos 2000s, comezou a verse un forte interese por parte da población na dixitalización de todo tipo de ferramentas e utilidades coas que interactuamos diariamente, co obxectivo de facilitar a nosa vida cotiá e permitindo levar no bolsillo ou na mochila prácticamente calquer cousa, dende unha libreta ata unha biblioteca completa.

Con isto en mente, levado ó sector da lectura, Bookbag nace da tentativa de fusionar dúas cousas xa existentes coma son as aplicacións de rexistro de lectura e as poucas bibliotecas online que existen a día de hoxe.

Como resultado temos un software único que quere unificar nunha sola aplicación o mellor dos dous mundos, puidendo así facer búsquedas de libros e confeccionar “andeis” a medida que imos lendo, permitíndonos levar un control sinxelo das nosas lecturas.

Palabras clave: biblioteca online, rexistro de lectura, bookbag, lectura

Abstract

In the 2000s, there was a growing interest among the population in the digitalization of all kinds of tools and utilities we interact with on a daily basis. The goal was to make everyday life easier by allowing us to carry practically anything in our pocket or backpack—from a simple notebook to an entire library.

With this in mind, and applying it to the sector of reading, Bookbag was created from the attempt to merge two already existing concepts: reading-log applications and the few online libraries available today.

The result is a unique piece of software that aims to unify the best of both worlds in a single application. It allows users to search for books and create personalized “bookshelves” as they read, making it possible to keep simple and organized track of their reading activity.

Keywords: online library, reading log, bookshelves, bookbag, readings

Índice

Introducción.....	6
Contexto e xustificación.....	6
Obxectivos.....	7
Obxectivo xeral.....	7
Obxectivos específicos.....	7
Fases de traballo.....	8
Fase de estudo.....	8
Análise da contorna.....	8
O mercado.....	10
Forma xurídica.....	11
Marketing.....	11
Inversión e financiación.....	12
Viabilidade.....	12
Planificación.....	12
Análise e deseño.....	13
Desenvolvemento.....	16
Documentación.....	19
Conclusíons.....	20
Propostas de mellora e valoración.....	21
Melloras.....	21
Por facer.....	21

Bibliografía.....	23
-------------------	----

1. Introdución

Bookbag é unha aplicación web pensada para ofrecer unha biblioteca online e rexistro de lecturas. O motivo polo cal se decidíu por algo así non é outro que o resultado da búsqueda dunha aplicación simple e práctica que cubrise as necesidade de rexistrar as lecturas, cunhas funcións básicas pero dixitalizadas.

1.1. Contexto e xustificación

Na actualidade, moitas persoas buscan formas cómodas e centralizadas de acceso a bibliotecas e levar un seguimento das súas lecturas. Aínda que existen distintas plataformas, moitas non se adaptan as necesidades concretas dos usuarios que desexan xestionar as súas lecturas levando un rexistro de forma sinxela e tendo a man un amplo catálogo do que nutrirse de recomendacións, explorar posibles lecturas, etc.

Con Bookbag preténdese solucionar un problema de complexidade que hoxe habita en tóda-las plataformas de rexistro de lecturas que comezan a engadir funcións sen fin e abruman ó usuario final. Pretende ser sinxela, económica, accesible e agradable ó uso.

A elección deste tema xustifícase polo crecente interese en ferramentas dixitales para xestionar diversos problemas cotidianos entre a población e a oportunidade de crear unha plataforma útil, accesible e fácilmente ampliable. Ademáis permítenos aplicar e reforzar coñecementos técnicos desenvolvendo un proxecto práctico cunha aplicación real.

2. Obxectivos

2.1. Obxectivo xeral

Desenvolver unha aplicación web mediante a cal ofrecer unha biblioteca online e rexistro de lecturas que permita ós usuarios explorar, organizar e levar un control das súas lecturas.

2.2. Obxectivos específicos

- Crear unha interface web funcional que permita engadir, editar e eliminar libros da biblioteca persoal.
- Deseñar unha base de datos que almacene a información dos usuarios e dos libros de forma estruturada.
- Implementar un sistema de rexistro de lectura que permite almacenar o estado dun libro (lido, lendo ou pendente) en diferentes andeis para cada usuario rexistrado na aplicación.

3. Fases de traballo

3.1. Fase de estudo

3.1.1. Análise da contorna

Análise PEST

- *Factores político-legais:* entre a normativa vixente en materia de protección de datos e servizos electrónicos destacan:
 - Reglamento Xeral de Protección de Datos (responsabilidade, protección de datos por defecto e dende diseño, transparencia).
 - Lei de Protección de Datos e Garantía dos Dereitos Digitais (LOPD-GDD) (rendición de contas, notificación de brechas de seguridade, rexistro de actividades de tratamiento).
 - Lei de Servizos da Sociedade da Información e o Comercio Electrónico (LSSI).
 - Dereitos ARSULIPO (acceso, rectificación, cancelación, supresión, limitación do tratamiento, portabilidad, oposición).

Tamén se debe ter en conta o instrumento normativo europeo *EU Cybersecurity Act*, que establece exixencias de certificación e estándares mínimos de seguridade para produtos e servizos digitais.

- *Factores económicos:* entre os factores económicos debemos ter en conta a inflación, xa que pode aumentar os costes operativos e de desenvolvemento. Tamen pode afectar ó poder adquisitivo dos clientes. As políticas fiscais tamen poderían influir negativamente xa que os impostos sobre ganancias, inversión e salarios teñen un impacto directo na rentabilidade da empresa.
- *Factores socioculturais:* os perfiles potenciais serán persoas que xa teñan uns hábitos de lectura asentados, que busquen simplicidade á hora de levar un rexistro de lecturas e que se poidan ver atraídos polas funcións xa

implementadas na aplicación. Un dos problemas actuais sería que moita xente non le ca mesma frecuencia que o facía antes.

- *Factores tecnolóxicos:* actualmente existe un crecemento cara a dixitalización globalmente, o cal impulsa a demanda de solucións tecnolóxicas más eficientes. Temos que ter en conta tamén que a protección de datos é un factor crítico no sector do software.

Análise da contorna específica

O sector no que se enmarca o proxecto presenta unha dinámica marcada por unha forte presión regulatoria, especialmente en materia de datos, seguridade e estándares tecnolóxicos. É dependente de manter certo grao de innovación, xa que no ámbito TIC, o sector caracterízase por ciclos de innovación curtos de alta esixencia.

Os competidores directos son empresas que ofrecen servizos similares, competindo por funcionalidade, prezo ou experiencia de usuario. En calquer momento startups poderían xurdir e introducir tecnoloxías disruptivas ou modelos de negocio más flexibles.

Análise DAFO e CAME

- DAFO:
 - Debilidades: a empresa é nova, por tanto terá que traballarse unha reputación dende o inicio. Existen outras empresas ofrecendo productos similares no mercado.
 - Ameazas: a situación política de inestabilidade, a non aprobación de Presupostos Xerais no goberno central condicionaría futuras políticas de pymes. Hai unha competencia intensa, con grandes bases de usuarios.
 - Fortalezas: proporciona valor engadido fronte as bibliotecas online tradicionais.
 - Oportunidades: escalabilidade global, ao ser dixital pode chegar a calquier usuario que dispoña dun dispositivo no que consumila.

- CAME:
 - Corrixir (debilidades): desenvolver unha estratexia de márketing dixital, centrada en redes sociais, comunidades lectoras e influencers literarios.
 - Afrontar (ameazas): tratar de fidelizar ó usuario a tracés dunha experiencia atractiva, notificacións útiles e interface coidada.
 - Manter (fortalezas): conserva-la simplicidade de uso e accesibilidade, evitando sobrecarga-la interface. Preserva-lo enfoque na experiencia do usuario.
 - Explotar (oportunidades): expandi-lo catálogo mediante integracións con APIs externas (como Open Library), ou bibliotecas públicas. Desenvolver funcionalidades sociais, coma reseñas, listas compartidas ou un sistema de mensaxería interna para compartir/recomendar lecturas.

3.1.2. O mercado

O mercado das plataformas de lectura dixitais e servizos asociados á xestión de hábitos lectores experimentou un crecemento sostido na última década impulsado pola transformación dixital e pola evolución dos hábitos culturais da poboación.

Dito mercado mostra unha tendencia de crecemento constante por diversas razóns:

- Incremento do uso de dispositivos electrónicos.
- Maior dispoñibilidade de contidos dixitais, gratis ou de pago.
- Tendencia global cara a plataformas online para a organización de tarefas cotiás.
- Aumento da alfabetización dixital, especialmente en poboación nova.

A estratexia de segmentación será indiferenciada. Esta baséase en usuarios xeráis con interese na lectura, estudiantes e comunidades/clubs de lectura. O público obxectivo engloba idades dende que un usuario aprende a ler ata usuarios de idades entre os 55 e 60 anos, xa que os maiores é habitual que carezan de tales hábitos dixitais.

As oportunidades de mercado residen na demanda crecente de ferramentas dixitais que cobran necesidades cotiás e a falta de plataformas sinxelas para o rexistro de lecturas.

Criterios de segmentacion demograficos e xeograficos

3.1.3. Forma xurídica

A forma xurídica ideal considérase “Empresario de Responsabilidade Limitada”, xa que non hai intención inmediata de incorporar traballadores, é un proxecto pequeno, nunha fase inicial, cun baixo investimento e un risco moderado.

3.1.4. Marketing

Tendo en conta o anteriormente mencionado, o proxecto debería orientarse tanto a captación como fidelización dos usuarios, poñendo o foco na proposta de valor, na experiencia de usuario e na creación de comunidade arredor da lectura. Isto fai fundamental deseñar unha estratexia.

Dita estratexia virá dada por un producto centrado no rexistro de lectura con andeis personalizados, favoritos e seguimento de libros, unha posible implementación cara futuro que recomende libros (via mail) e unha interface amigable e pulida. O prezo terá varios plans dispoñibles:

- Freemium: acceso gratuíto básico + algunas opcións premium que posteriormente serán exclusivas do plan premium.
- Premium: acceso completo a tóda-las funcións da aplicación, incluíndo andeis personalizados, recomendacións, etc.

A plataforma de momento será únicamente unha aplicación web, aínda que a futuro poderíase contemplar o desenvolvemento dunha aplicación android nativa.

En canto á promoción utilizaríase unha combinación entre difusión en redes sociais, comunidades lectoras e a contratación de influencers do sector literario.

A estratexia de fidelización levaríase a cabo mediante unha actualización constante do catálogo apoiándonos dunha integración con APIs e bibliotecas públicas online para mantelo rico e diverso. Ter en conta que esta estratexia depende polo momento de integracións futuras da aplicación que non teñen peso actualmente xa que primeiro teríamos que entrar no mercado e iso cómpranos tempo para desenvolveras funcións faltantes actualmente.

3.1.5. Inversión e financiación

O orzamento para un MVP (Producto Mínimo Viable) estímase que sería, orientativamente:

Custos directos

- Desenvolvemento de software: a cargo do promotor.
- Hosting: 45,84€ (con IVA) os primeiros 12 meses (con OVHCloud).
- Dominio: “bookbag.gal” 9.49€ os primeiros 12 meses (con OVHCloud).
- Ferramentas de desenvolvemento: PHPStorm 131.89€ (con IVA).

Custos indirectos

- Promoción e marketing:
 - Influencers: 200€ (menos de 20 mil seguidores).
 - Campañas en redes sociais: 2400€ os primeiros 12 meses (por red social).

O total sitúase en torno os **2787,22€** para o primeiro ano.

Inicialmente a fonte de financiamento será propia, mediante a aportación do socio único. Nun futuro pódese estudar unha ampliación do capital.

3.1.6. Viabilidade

fdsa

Avalía se o proxecto é viable economicamente (proxección de ingresos/gastos).

Analiza a viabilidade técnica e operativa (recursos, capacidades, tempos).

Analiza riscos e como se mitigarán.

Conclusión sobre a sustentabilidade do proxecto.

3.2. Planificación

O proxecto abrangue o deseño, desenvolvemento e implementación dunha plataforma dixital orientada á lectura, composta por:

- Unha biblioteca online con catálogo consultable.
- Un sistema de rexistro de lectura.

- Perfís de usuario nos cales xestionar as súas configuracións e datos persoais.
- Interface responsive e accesible vía web.
- Panel de administración para xestionar o catálogo.

A execución do proxecto presenta diversos riscos que deben ser considerados dende o inicio. Entre os más relevantes destacan os riscos técnicos, como posibles fallos de infraestrutura ou errores no desenvolvemento; os riscos legais asociados ao cumprimento da normativa de protección de datos; e os riscos organizativos derivados de atrasos, cambios no alcance ou limitacións de recursos. Para afrontar estes desafíos, o proxecto deberá sopesar a futura incorporación dun equipo humano composto por un xefe ou xestora de proxecto encargado da coordinación xeral, desenvolvedores de front-end e back-end responsables da implementación técnica, un deseñador ou deseñadora UX/UI para garantir unha boa experiencia de usuario e, finalmente, un responsable de calidade e un asesor legal para supervisar as probas e o cumprimento normativo. Actualmente, cóntase soamente co promotor do proxecto para facerse cargo de tódalas tarefas.

As tarefas fundamentais inclúen a análise inicial de requisitos, o deseño da arquitectura e das interfaces, o desenvolvemento das distintas funcionalidades da plataforma, a integración e probas exhaustivas, así como o despregamento final e a documentación correspondente. Todo o proceso estrutúrase arredor de fitos clave, como a definición dos requisitos, a realización do prototipo, a construcción dos módulos principais (autenticación, catálogo e rexistro de lectura), a superación das fases de probas e, finalmente, o lanzamento da versión mínima viable da plataforma, seguido dunha avaliación continua para introducir melloras posteriores.

3.3. Análise e deseño

Tendo en conta as necesidades do proxecto, considéranse requisitos recomendados de hardware para a elaboración da aplicación:

Equipo:

- Placa: ASUS TUF B-550-Plus AM4.
- CPU: AMD Ryzen 5 5600 AM4.

- Kit G.Skill Trident Z 2x16GB DDR4 RAM.
- Disco Samsung 980 NVMe PCIe 3.0 M.2 de 1TB.
- Fonte e caixa acorde (recoméndase fonte con certificación 80 plus).

Periféricos:

- Recoméndase doble pantalla de 24" antireflectora para desenvolver con comodidade con documentacións a nun lado e código no outro.
- Teclado e ratón ó gusto, usaráse un Logitech K120 e un Razer Deathadder V2.

Server:

- Empregaráse un servidor Nginx (dockerizado por comodidade para desenvolvimento).

Sería recomendable ter en conta os plans actuais de hosting á hora de despregala aplicación definitivamente para servi-la ó cliente final. Tívose en conta que unha aplicación de este tipo (Laravel con necesidade mediana de almacenamento), nun futuro escalará cara a implementación de Redis para o manexo de caché (para evitar cargas iterativas das listaxes de libros e das estanterías de forma que sexa máis rápida e eficiente) e require dunha CLI para o seu manexo xeral.

Os plans que se sopesan son os seguintes:

- Hostinger (VPS): 5€/mes - 1 vCPU - 4 GB RAM - 4 TB de tráfico.
- Hetzner Cloud (VPS): 3.49€/mes - 2 vCPU - 4 GB RAM - 20 TB de tráfico.
- OVHCloud (VPS): 3.82€/mes - 4 vCPU - 8GB RAM – 75 GB NVMe de tráfico.

Sacamos en limpio que según o idóneo sería comezar cun plan como o que ofrece Hetzner Cloud, que é o mais económico ademais do mais moderado en prestacións e nos primeiros trimestres ir valorando as necesidades que vaian xurdindo cara unha migración a un servicio maior como podería ser o ofrecido por OVHCloud no plan seguinte (4 vCore, 12 GB RAM, 100NVMe por 5.95€/mes), en

xeral os de OVHCloud están moi ben xa que ofrecen flexibilidade á hora aumenta-lo espacio de almacenamento.

En canto ó software empregado durante o desenvolvemento contamos cas seguintes ferramentas:

- **Laravel** v12.0: framework principal da aplicación, é o encargado de todo o backend, usa PHP v8.4.
- **Spatie Laravel Permission** v6.21: será a libraría encargada de xestionar os permisos (non os roles).
- **Livewire** (Flux v2.1.1 e Volt v1.7.0): como framework principal de frontend, usa Tailwind CSS v4.0.7 e Vite v7.0.4, apoiámonos puntualmente en WireUI 2.4 para certos componentes, en SweetAlert2 v11.0 para xestiona-las ventanas modais de confirmación e en AlpineJS v3.x.x para a animación do dropdown/botón hamburguesa da barra superior de navegación.
- Servidor web: **Nginx** v1.29.3.
- Base de datos: **PostgreSQL** v16.0.
- Infraestructura: Docker y Docker Compose con Nginx v1.29.3, PostgreSQL v16.0 e PHP-FPM v8.4.
- Testing: **Pest PHP** v4.0.
- Control de versións: **Git** (remoto en [Github](#)).
- Xestores de paquetes: **Composer** v2.8.6 e **NPM** v10.8.2 e Node v20.12
- Integración Continua (CI): Github (Github Actions).
- Cache: Redis, non implementado por completo (comentado no .env e no compose de Docker).
- Utilidades: **Make**, Bash.
- En canto a editores usouse **Penpot** para o deseño de interfaces e mockups, **PHPStorm** para o desenvolvemento da aplicación (código), **LibreOffice Writer** para as memorias (e **Impress** a presentación).

O uso das linguaxes/frameworks mencionados ven motivado pola intención do equipo de desenvolvemento por aprender Laravel e a súa contorna. O framework é moi potente, open-source, cobre moi ben as necesidades da aplicación que queremos levar á cabo e ofrécenos cantidade de ferramentas que a futuro serán interesantes para engadir novas features ou melloras.

A mención de Redis como sistema de cacheado ven dada polo uso habitual en plataformas parecidas a que se quere lograr, axudando a manter unha estabilidade e un rendemento maiores en contornas de produción. Nun principio non se pensa contar ca súa implementación pero é interesante tela en conta a futuro xa que nos aforrará posibles problemas.

Os diagramas da base de datos pódense atopar no cartafol “diagramas_db”.

Os deseños das interfaces pódense atopar na carpeta “diseño_interfaces”.

3.4. Desenvolvemento

Despregamento

Inicialmente partiráse da idea de desenvolver a aplicación nun entorno Docker (con docker compose), pola súa facilidade de uso e practicidade á hora de despregar e visualizar os avances feitos. Dito compose será composto polos seguintes contedores:

- Nginx: como servidor web, accesible en localhost no porto 8080.
- PHP-FPM: como servidor PHP, o cal compilará o código e devolverá ó servidor web para ser mostrado.
- Postgres: como base de datos, o cal se encargará da persistencia de datos nun volume da nosa máquina local. Este será accesible no porto 33006 se usamos un cliente de SQL.
- PgAdmin: como cliente web PGSQQL, o cal servirá para acceder a base de datos sen necesidade dun cliente SQL. Este será accesible en localhost no porto 8081.
- Workspace: é un contedor PHP-FPM personalizado, o cal conterá librerías extra típicas de desenvolvemento como por exemplo Xdebug, acceso o debugging de Redis, ou á consola de Laravel (Artisan). Só se usará en modo desenvolvemento, non en produción.

- Redis: sería utilizado para xestionar a memoria cache da aplicación e ir comunicándose coa base de datos cada certo tempo para manter persistente a información en cola. Como xa se mencionou anteriormente este servizo ven deshabilitado inicialmente, existe no código pero estará comentado polo de agora xa que non lle sacaríamos partido mentres non migremos a un modelo de alta concurrencia.

Backend

Estará composto por unha aplicación completa en Laravel, a cal conterá as seguintes entidades: *Users*, *Roles*, *Books*, *Bookshelves*, *Bookshelf_types* e *Review*.

Ditas entidades poderán interactuar coa aplicación como se indica nos diagramas que temos no cartafol ‘*diagramas_db*’.

Laravel noso noso dispor unha serie de facilidades como as migracións, as cales serán utilizadas pola velocidade que estas nos brindan á hora de probar funcionalidades e acceder en modo desenvolvemento á aplicación. Posteriormente, estas servirían en caso de querer despregar a base de datos noutra plataforma seguindo o mesmo esquema que tiñamos ou en caso de posibles errores. Facilitan tamen a xestión de táboas e columnas en caso de simplemente querer engadir/eliminar algunha delas, simplificando o proceso.

Frontend

Estará composto por unha interface feita con Livewire e Blade según necesidade, intentando sacar partido da integración que ten Laravel con estes. Intentaremos organizar o código en componentes facendo a aplicación comprensible e fácil de manter na medida do posible.

Permisos

Os permisos serán implementados usando Spatie (Laravel Permission), xa que propón unha forma sinxela e rápida de xestionarlos. Esta libraría creará as táboas necesarias para almacenar e relacionar os usuarios cos roles e permisos que nos queramos na aplicación e poderemos posteriormente engadir ou eliminar os que precisemos mediante o uso da súa respectiva migración.

Testing

Usarase Pest para implementar os test unitarios e de integración. Estes estarán automatizados via CI usando Github Actions cun workflow (fluxo de traballo) que cada vez que se faga un push contra o repositorio levantará os contedores para executar os tests, executarás e devolverá un tick verde na páxina do repositorio en Github en caso de éxito ou unha cruz vermella en caso de erro.

Estudouse a posibilidade de desenvolver a aplicación mediante a metodoloxía TDD (*Test Driven Development* ou *Desenvolvemento Dirixido por Tests*) pero descartouse a idea no momento de plantexar o tempo que implicaría desenvolvela baixo esa metodoloxía.

Tooling

Usaremos Make como ferramenta de scripting para facilitar o uso de docker, a instalación de dependencias, a posta en marcha do proxecto cando se desprega, a limpeza unha vez rematadas as tarefas oportunas... etc. Os comandos dispoñibles estarán listados no README principal e requieren ter instalado o paquete ‘make’.

Backups

Teremos dispoñible un script que fai automáticamente un backup (por defecto a ruta destino será un cartafol chamado ‘/backups/bookbag’ na raiz do proxecto) cun dump da base de datos completa. En caso de realizalo, débese ter en conta que este respetará os anteriores backups e usará sempre un timestamp para os nomes co seguinte formato: ‘Ymd-HMS.dump’

3.4.1. Documentación

A documentación realizouse mediante insercións na parte de código, tanto na parte de lóxica (mediante PHPDoc), como nas vistas (mediante comentarios HTML/Blade) e nos ficheiros complementarios como Dockerfiles e Makefiles (mediante comentarios convencionais).

4. Conclusíons

Tendo en conta os obxectivos plantexados inicialmente, considérase que os seguintes foron implementados apropiadamente, salvando distancias con certas prácticas que hoxe en día se esperan neste tipo de aplicacións a día de hoxe:

- Conseguíuse desenvolver unha aplicación web mediante a cal ofrecer unha biblioteca online e un rexistro de lecturas que permita ós usuarios explorar, organizar e levar un control das súas lecturas.
- Elaborouse unha base de datos que almacena a información dos usuarios e dos libros de forma estruturada.
- Creóuse unha interface web funcional que permite engadir e eliminar libros da biblioteca persoal dun usuario.
- Conseguíuse (mediante un control por roles) engadir, editar e eliminar libros da base de datos co fin de manter a biblioteca actualizada.
- Implementouse un sistema de rexistro de lectura que permite almacenar o estado dun libro (lido, lendo ou pendente) en diferentes andeis para cada usuario rexistrado na aplicación.

Para rematar, este proxecto favoreceu moito o aprendizaxe de diferentes tecnoloxías e procesos que durante o curso non poden terse en conta dada a súa magnitud e agradécese que pese o verde que todavía se atopa a aplicación poidérase aprender tanto en tan pouco tempo.

5. Propostas de mellora e valoración

5.1.1. Melloras

Como se indicaba no apartado anterior, a aplicación todavía se atopa nun estadío temperán. Entre os diferentes aspectos que poderíamos salientar como posibles candidatos a mellorar, considéranse os seguintes:

- Repaso profundo o funcionamiento xeral das interfaces en móvil e escritorio (o responsiveness xeral).
- Adición de tests de integración que cobren por completo a navegación da aplicación.
- Mellora na sección do perfil de usuario, o cambio de apariencia precisa ser adaptado apropiadamente.
- Revisión e probas de navegación con teclado ou mediante comandos de voz.

5.1.2. Por facer

Certas partes do código da aplicación quedan baleiras, non rompendo a aplicación pero como guiño do que se pretendería facer a futuro, como por exemplo:

- A reseña exemplo da vista detalle dos libros, mostra como quedaría se tiveramos a lóxica da clase “Review” rematada, permitindo así ós usuarios autenticados crear, editar e eliminar (soamente as súas) reseñas nos libros.
- O controlador de Edición, indicando que realmente cada libro debería ter un conxunto de edicións distintas e son estas as que se poderían listar en lugar dos propios libros porque dunha edición a outra un mesmo libro cambia e por ende quizáis sería oportuno este cambio, de feito sería un bo lugar no que aplicar herencia. Por outra parte poderíase simplemente engadir detalles acerca de cada edición dun mesmo libro non cambiando a lóxica dos libros senón mostrándoas no detalle do libro en cuestión.
- O servizo de Redis comentado no docker compose de desenvolvemento, permitindo a adición dun sistema de cacheado sen apenas configurar nada,

xa que soamente teríamos que descomenta-lo servicio, descomentar a sección oportuna no .env e cambiar os valores acorde ó hosting.

- Algún componente que está ben modularizado respecto ós outros, non había tempo abondo para facer o mesmo con todos e comprobar o seu correcto funcionamento pero non se respeta SoC (“Separation of Concerns”) todo o que se podería. Facéndoo, ademáis, sería moito máis sinxelo testear ditas partes de forma independente ó facer tests de fluxo.

6. Bibliografía

1. Comparativa de especificaciones de hosting [Internet]. OVHCloud. [citado 24 de noviembre de 2025]. Recuperado a partir de: <https://www.ovhcloud.com/es-es/vps/configurator/>
2. Comparativa de especificaciones de hosting [Internet]. Hostinger. [citado 24 de noviembre de 2025]. Recuperado a partir de: <https://www.hostinger.com/vps/laravel-hosting>
3. Comparativa de especificaciones de hosting [Internet]. Costgoat. [citado 24 de noviembre de 2025]. Recuperado a partir de: <https://costgoat.com/pricing/hetzner>
4. Documentación de Laravel [Internet]. Laravel. [citado el 22 de noviembre de 2025]. Recuperado a partir de: <https://laravel.com/docs/12.x>
5. Separation of concerns (SoC) [Internet]. Wikipedia. [citado el 21 de noviembre de 2025]. Recuperado a partir de: https://en.wikipedia.org/wiki/Separation_of_concerns
6. Redis [Internet]. Laravel. [citado el 24 de noviembre de 2025]. Recuperado a partir de: <https://laravel.com/docs/12.x/redis>
7. Laravel Docker examples [Internet]. Github. [citado el 19 de noviembre de 2025]. Recuperado a partir de: <https://github.com/dockersamples/laravel-docker-examples>
8. Learn Makefiles [Internet]. Makefile Tutorial. [citado el 18 de noviembre de 2025]. Recuperado a partir de: <https://makefiletutorial.com/>
9. EU Cybersecurity Act [Internet]. European Comission. [citado o 24 de novembro de 2025]. Recuperado a partir de: <https://digital-strategy.ec.europa.eu/en/policies/cybersecurity-act>

Anexo A. Título do anexo