

**IPB Digital Library**

## Redney Monteiro – a46398

Master in Informatics Software Architectures Prof. Paulo Alves

Bragança 2023-2024

# Abstract

The abstract in english.

**Keywords:** The keywords

# Resumo

O resumo em português.

**Palavras-chave:** palavras chave.

# Contents

1. [Introduction](#_bookmark0) 1
   1. [Document Structure](#_bookmark1) 1
2. [Problems and Goals](#_bookmark4) 3
   1. [Goal 10: Reduce inequality within and among countries](#_bookmark5) 5
3. [Background](#_bookmark6) 7
   1. [Main Concepts of the Project Theme](#_bookmark7) 7
   2. [State-Of-The-Art](#_bookmark8) 7
4. [Requirements](#_bookmark9) 11
   1. [Technologies/Software](#_bookmark10) 11
      1. [Visual Studio Code](#_bookmark11) 11
      2. [MongoDB Compass](#_bookmark12) 11
      3. [Python](#_bookmark13) 12
      4. [FastAPI](#_bookmark14) 12
      5. [JavaScript](#_bookmark15) 12
      6. [Node.js](#_bookmark16) 13
      7. [React](#_bookmark17) 13
      8. [Material UI 5](#_bookmark18) 13
      9. [JWT](#_bookmark19) 14
      10. [Axios](#_bookmark20) 14

v

* + 1. [Git](#_bookmark21) 15
    2. [GitHub](#_bookmark22) 15
  1. [User Stories](#_bookmark23) 15
  2. [Database Schema](#_bookmark25) 17
  3. [Mockups](#_bookmark27) 17

1. [Prototype Development](#_bookmark28) 19
   1. [Back-End Development](#_bookmark29) 19
      1. [Database](#_bookmark30) 19
      2. [API and relation with Database](#_bookmark31) 19
      3. [Authentication using JWT](#_bookmark32) 20
      4. [Handle with images](#_bookmark33) 22
      5. [Testing endpoints](#_bookmark34) 22
   2. [Front-End Development](#_bookmark37) 24
   3. [Donations and PayPal Integration](#_bookmark38) 24
   4. [Google Maps Integration](#_bookmark39) 24
   5. [Turning website dynamic](#_bookmark40) 25
2. [Conclusions](#_bookmark41) 27

# List of Figures

[Figure 1 - User stories 17](#_Toc166530665)

[Figure 2 - Use case nível 0 20](#_Toc166530666)

[Figure 3 - Gerir book, admin 20](#_Toc166530667)

[Figure 4 - Gerir book, student 21](#_Toc166530668)

[Figure 5 - Gerir booksh elf 21](#_Toc166530669)

[Figure 6 - Gerir borrowed 22](#_Toc166530670)

[Figure 7 - Gerir account, user 22](#_Toc166530671)

[Figure 8 - Database Schema 23](#_Toc166530672)

[Figure 9 – Mockup 24](#_Toc166530673)

[Figure 10 - FastAPI endpoints 26](#_Toc166530674)

[Figure 11 - Aceder base de dados 27](#_Toc166530675)

[Figure 12 - Metodo de encriptação 28](#_Toc166530676)

[Figure 13 - Estrututa de Pasta Python 29](#_Toc166530677)

[Figure 14 - Website image 30](#_Toc166530678)

# Chapter 1 Introduction

In today's digital age, transformation permeates every aspect of life, including the way we interact and access books. Libraries, as guardians of knowledge, are adapting to keep up with this change. This project focuses on developing a website for a library, with the aim of facilitating and making access to knowledge more efficient.

Users can search for specific titles, see what books are available, and even create private virtual shelves to organize the books they want to read in the future. Additionally, users can request books to be picked up from the library.

The IPB Digital Library serves as a digital catalog, allowing users to explore the library's collection efficiently and conveniently. We are committed to continually improving our website to meet our users' needs and promote a love of reading. Thank you for your understanding and support.

With advanced search functionality, users can easily find the books they are looking for, saving time and effort. Additionally, they can create private virtual bookshelves for better organization of their favorite books and can order books easily and conveniently.

This library website is a step towards a more informed and sustainable future. While we are proud of what we have achieved so far, we recognize that there is still room for improvement and expansion. We are committed to continuing to develop and improve our website to better meet the needs of our users and promote literature. Thanks.

# 

# Chapter 2

# Problems and Goals

## 2.1 Problems

As bibliotecas tradicionais, de fato, enfrentam vários desafios na era digital. A necessidade de espaço físico, a manutenção dos livros, a limitação dos horários de funcionamento e a dificuldade de encontrar livros específicos são alguns dos problemas comuns. Além disso, os utilizadores podem ter dificuldade em aceder à biblioteca devido à distância ou a outras restrições.

O nosso website de biblioteca foi desenvolvido para resolver esses problemas. Embora os utilizadores não possam ler os livros diretamente no website, eles têm a capacidade de pesquisar e visualizar quais livros estão disponíveis na biblioteca. Isso elimina a necessidade de espaço físico e mantém os livros em perfeitas condições. O website está disponível 24 horas por dia, 7 dias por semana, permitindo aos utilizadores aceder aos recursos da biblioteca a qualquer momento e em qualquer lugar.

Os utilizadores podem pesquisar facilmente livros específicos através da funcionalidade de pesquisa avançada. Além disso, eles podem requisitar livros para retirada na biblioteca, eliminando a necessidade de se deslocarem fisicamente até lá. Cada utilizador pode criar a sua própria estante virtual privada, permitindo-lhes organizar os livros de acordo com as suas preferências.

Este website de biblioteca é um passo em direção a um futuro mais informado e sustentável. Estamos comprometidos em continuar a desenvolver e aprimorar o nosso website para atender melhor às necessidades dos nossos utilizadores e promover o amor pela leitura. Obrigado pela sua compreensão e apoio.

**2.2 Goals**

O IPB Digital Library foi desenvolvido com o objetivo de facilitar o acesso ao conhecimento, melhorar a eficiência da pesquisa, promover a organização pessoal, facilitar o empréstimo de livros, promover a sustentabilidade e estar disponível 24/7. Ao digitalizar a coleção de livros da biblioteca, pretendemos tornar o conhecimento mais acessível a todos, independentemente da sua localização geográfica. Com a funcionalidade de pesquisa avançada, os utilizadores podem encontrar facilmente os livros que procuram, economizando tempo e esforço. Através da criação de estantes virtuais privadas, os utilizadores podem organizar os seus livros favoritos de acordo com as suas preferências. Os utilizadores podem requisitar livros para retirada na biblioteca de forma fácil e conveniente, eliminando a necessidade de se deslocarem fisicamente até lá. Ao reduzir a necessidade de transporte físico e manter os livros em perfeitas condições, estamos a contribuir para um futuro mais sustentável. O website está disponível 24 horas por dia, 7 dias por semana, permitindo aos utilizadores aceder aos recursos da biblioteca a qualquer momento e em qualquer lugar. Estes objetivos refletem o nosso compromisso em promover a literatura e o acesso ao conhecimento, ao mesmo tempo que utilizamos a tecnologia para superar as limitações das bibliotecas tradicionais. Estamos ansiosos para ver como o IPB Digital Library continuará a evoluir e a servir a nossa comunidade de utilizadores.

# Chapter 3 Background

O IPB já tem um Sistema identico.

# Chapter 4 Requirements

O capítulo 4 tem como objetivo apresentar as tecnologias e linguagens de programação utilizadas no decorrer deste trabalho. Além disso, aborda as aplicações já desenvolvidas que servirão como orientação para o desenvolvimento do nosso website de biblioteca.

## Technologies/Software

### Python

Python é uma linguagem de programação de alto nível, versátil e poderosa, que permite trabalhar de forma rápida e integrar sistemas de forma mais eficaz[3]. Sua sintaxe clara e legível torna Python fácil de aprender para novos programadores[3]. Além disso, Python é usado em muitas áreas, incluindo desenvolvimento web, ciência de dados, aprendizagem de máquina, automação, entre outros [3]. Python pode ser facilmente instalado em plataformas Windows, macOS ou Linux a partir do site oficial Python (python.org). O processo de instalação é simples e inclui opções para instalar a biblioteca padrão, conjunto de testes, lançador e pip também[4]. Você pode usar o pip para baixar pacotes de vários índices como o Índice de Pacote Python [4].

### FastAPI

FastAPI é um framework moderno e de alto desempenho para a construção de APIs com Python, baseado em dicas de tipo padrão[1][2]. Ele oferece um desempenho muito alto, comparável ao NodeJS e Go, graças ao Starlette e Pydantic[1][2]. É um dos frameworks Python mais rápidos disponíveis[1]. Além de ser rápido para executar, o FastAPI também é rápido para codificar. Ele permite aumentos significativos na velocidade de desenvolvimento, reduzindo o número de erros induzidos pelo desenvolvedor [1]. O FastAPI é intuitivo, oferecendo um ótimo suporte ao editor, com conclusão em todos os lugares, reduzindo o tempo gasto na depuração[1].O FastAPI é projetado para ser fácil de usar e aprender, minimizando o tempo gasto na leitura da documentação[1]. Ele também minimiza a duplicação de código, obtendo múltiplas funcionalidades de cada declaração de parâmetro [1].

### MongoDB

MongoDB é um banco de dados de documentos que oferece a escalabilidade e a flexibilidade que você deseja com a consulta e indexação que você precisa[5]. Ele armazena dados em documentos flexíveis semelhantes a JSON, o que significa que os campos podem variar de documento para documento e a estrutura de dados pode ser alterada ao longo do tempo [5]. Além disso, o MongoDB é intuitivo para os desenvolvedores aprenderem e usarem, ao mesmo tempo que fornece todas as capacidades necessárias para atender aos requisitos mais complexos em qualquer escala[5]. As consultas ad hoc, a indexação e a agregação em tempo real fornecem maneiras poderosas de acessar e analisar seus dados [5]. Como um banco de dados distribuído em seu núcleo, o MongoDB oferece alta disponibilidade, escalabilidade horizontal e distribuição geográfica integradas e fáceis de usar [5]. Em resumo, o MongoDB é uma ferramenta poderosa para gerenciamento de dados, proporcionando alta performance, facilidade de uso e uma série de funcionalidades que facilitam o desenvolvimento[6].

### JavaScript

JavaScript é uma linguagem de programação de alto nível, dinâmica e interpretada, conhecida principalmente por sua função como linguagem de script para páginas da web[11][12]. Foi criada por Brendan Eich em 1995 e desde então evoluiu para uma das linguagens de programação mais populares do mundo[11]. A força do JavaScript reside em sua capacidade de permitir a criação de conteúdo dinâmico e interativo para website [11][12]. Ele permite que os desenvolvedores manipulem elementos da página da web e respondam a ações do usuário em tempo real, como cliques de mouse, pressionamentos de tecla e alterações de formulário[11][12]. Além disso, JavaScript é a única linguagem que combina HTML, CSS e JavaScript, tornando-a única em sua capacidade de criar interfaces de usuário ricas e interativas[11][12]. Além disso, JavaScript não se limita ao navegador. Com o advento do Node.js, JavaScript agora pode ser usado para desenvolver aplicações do lado do servidor, permitindo que os desenvolvedores usem uma única linguagem para desenvolver aplicações web completas[11].

### React

ReactJS, também conhecido como React.js ou simplesmente React, é uma biblioteca JavaScript de código aberto para a construção de interfaces de usuário baseadas em componentes [9]. É mantida pela Meta (anteriormente Facebook) e uma comunidade de desenvolvedores individuais e empresa s[9]. ReactJS permite que você construa interfaces de usuário a partir de peças individuais chamadas componentes¹. Cada componente em ReactJS é uma entidade modular e reutilizável, o que torna o código mais organizado e fácil de manter[9]. Além disso, ReactJS segue o paradigma de programação declarativa, o que significa que os desenvolvedores projetam visualizações para cada estado de uma aplicação, e o React atualiza e renderiza os componentes quando os dados mudam[9]. Uma das principais vantagens do ReactJS é que ele só renderiza novamente as partes da página que mudaram, evitando a renderização desnecessária de elementos DOM inalterados[9]. Isso melhora significativamente a eficiência e o desempenho das aplicações web. ReactJS também pode ser usado para desenvolver aplicações de página única, móveis ou renderizadas no servidor com frameworks como Next.js[9]. Como o ReactJS está apenas preocupado com a interface do usuário e a renderização de componentes para o DOM, as aplicações React muitas vezes dependem de bibliotecas para roteamento e outras funcionalidades do lado do cliente[9].

### Boostrap

Bootstrap é uma biblioteca popular de HTML, CSS e JavaScript para o desenvolvimento de sites responsivos e aplicações web[14][15]. Foi desenvolvido pela equipe do Twitter e é agora mantido por desenvolvedores em todo o mundo[14][15]. Bootstrap oferece uma variedade de componentes reutilizáveis que ajudam os desenvolvedores a criar interfaces de usuário consistentes e atraentes de forma rápida e eficiente[14][15]. Estes componentes incluem botões, formulários, navegação, janelas modais, tooltips, carrosséis e muito mais[14][15]. Além disso, Bootstrap inclui um sistema de grid responsivo que permite aos desenvolvedores criar layouts flexíveis e adaptáveis para uma variedade de tamanhos de tela[14][15]. Isso é especialmente útil na era moderna, onde os sites precisam ser acessíveis e funcionais em uma variedade de dispositivos, desde desktops até smartphones [14],[15]. Bootstrap também inclui uma série de plugins JavaScript que adicionam funcionalidades interativas aos componentes[14],[15]. Estes plugins permitem aos desenvolvedores adicionar comportamentos dinâmicos aos seus sites sem ter que escrever muito código JavaScript [14],[15].

## User Stories

As User Stories são uma parte fundamental do desenvolvimento ágil de software [17],[18]. Elas são descrições informais e gerais de um recurso de software, escritas a partir da perspectiva do usuário final [17],[18]. O objetivo de uma User Story é articular como uma tarefa específica pode oferecer um determinado valor ao cliente[17],[18]. As User Stories são expressas em linguagem simples e não técnica, o que ajuda a dar contexto à equipe de desenvolvimento e suas iniciativas[17],[18]. Depois de ler uma User Story, a equipe sabe por que está desenvolvendo o que está desenvolvendo e qual o valor que isso cria[17],[18]. As User Stories são uma das principais componentes de um programa ágil. Elas fornecem uma estrutura centrada no usuário para o trabalho diário, o que impulsiona a colaboração, a criatividade e um produto melhor em geral[17],[18].



Figure 1 - User stories

## Descrição do Sistema

O IPB Digital Library, destinado aos estudantes e funcionários da escola IPB, é uma plataforma digital robusta que visa facilitar o acesso e a interação com os recursos da biblioteca. Após um processo de registo simples e seguro, os usuários podem fazer login no website. Este processo de autenticação garante que cada usuário tenha uma experiência personalizada e segura na plataforma.

Uma vez autenticados, os usuários têm a capacidade de pesquisar livros disponíveis na biblioteca. Esta pesquisa pode ser feita por nome do livro ou categoria, permitindo aos usuários encontrar facilmente o que procuram. Além disso, eles podem visualizar detalhes desses livros, como o autor, o ano de publicação, a sinopse e muito mais. O sistema também oferece a funcionalidade de criar estantes virtuais privadas. Estas estantes permitem aos usuários organizar os seus livros favoritos de acordo com as suas preferências. É importante notar que cada livro só pode ser adicionado a uma estante de cada vez, garantindo que a organização seja mantida. Além disso, os usuários têm a opção de requisitar livros. Eles podem indicar quais livros desejam requisitar e o sistema irá processar o pedido. Se o livro não for devolvido até a data limite, o sistema marcará automaticamente o livro como atrasado. Isso ajuda a manter a ordem e a eficiência na gestão dos livros da biblioteca.

Por fim, o sistema também permite que os usuários enviem comentários. Esta funcionalidade proporciona uma plataforma para feedback e discussão, permitindo aos usuários partilhar as suas opiniões e experiências.

## Use cases

Os casos de uso são uma ferramenta poderosa utilizada na análise de negócios para definir os requisitos funcionais de um sistema ou software[27]. Um caso de uso descreve como um usuário interage com um sistema para alcançar objetivos específicos[24],[25]. Ele delineia o fluxo de entradas do usuário, estabelecendo caminhos de sucesso e falha para atingir metas¹. Isso permite que as equipes de produto compreendam melhor o que um sistema faz, como ele funciona e por que ocorrem erros [24]. Os casos de uso são frequentemente empregados em ambientes de desenvolvimento de software para simplificar conceitos complicados, mas eles podem ser igualmente importantes na gestão de projetos para a coleta de requisitos e definição do escopo de um projeto[25]. Eles são usados em domínios de gestão de produtos, desenvolvimento de produtos e testes de produtos[25]. Os gerentes de produto documentam casos de uso focados no usuário, enquanto os desenvolvedores documentam casos de uso focados no produto[25]. Os casos de uso focados no usuário estão principalmente preocupados com o usuário e seus objetivos. Estes são então passados para os desenvolvedores para orientar a tomada de decisões durante o processo de desenvolvimento do produto[25]. Os casos de uso são projetados para revelar as demandas do sistema no início do processo[25]. Eles se concentram nos usuários do sistema, em vez do próprio sistema[25]. Um caso de uso deve ser compreensível para todos os stakeholders, não apenas desenvolvedores e testadores, pois eles são principalmente prosa narrativa[25].

Uma imagem com texto, captura de ecrã, diagrama, círculo

Descrição gerada automaticamente

Figure 2 - Use case nível 0

Uma imagem com texto, captura de ecrã, diagrama, círculo

Descrição gerada automaticamente

Figure 3 - Gerir book, admin

Uma imagem com texto, diagrama, círculo, captura de ecrã

Descrição gerada automaticamente

Figure 4 - Gerir book, student

Uma imagem com texto, captura de ecrã, círculo, diagrama

Descrição gerada automaticamente

Figure 5 - Gerir bookshelf

Uma imagem com texto, captura de ecrã, círculo, diagrama

Descrição gerada automaticamente

Figure 6 - Gerir borrowed

Uma imagem com texto, diagrama, círculo, captura de ecrã

Descrição gerada automaticamente

Figure 7 - Gerir account, user

## Database Schema

The Database was defined based on the functionalities of the application and showed in a table schema for better understatement, as showed in figure [4.1](#_bookmark26).

Uma imagem com texto, captura de ecrã, Tipo de letra, Retângulo

Descrição gerada automaticamente

Figure 8 - Database Schema

## Mockups

Ao criar o mockup para o nosso projeto, optamos por um design limpo e simples, utilizando a cor e o ícone do IPB. Acreditamos que essa abordagem não só reflete a identidade visual do IPB, mas também facilita a usabilidade e a compreensão do projeto pelos usuários.

O design limpo e simples ajuda a destacar as funcionalidades mais importantes do nosso projeto, evitando distrações desnecessárias. A cor do IPB foi usada para dar um toque de familiaridade e consistência ao projeto, enquanto o ícone do IPB serve como um lembrete constante da associação do projeto com a instituição.

Para desenvolver o mockup, utilizamos o Figma, uma ferramenta de design de interface colaborativa[17]. O Figma permite que os designers trabalhem juntos em tempo real, tornando o processo de design mais eficiente e integrado[17]. Além disso, o Figma suporta a criação de protótipos realistas, permitindo-nos testar a experiência interativa e obter feedback mais cedo no processo de design [17]. Para aceder ao mockup - [Link do Mockup](https://www.figma.com/design/5NjDTsigGIR8ZCMyYPtjnY/Untitled?node-id=0%3A1&t=PcSitJbESRhB3LEZ-1)

**Uma imagem com texto, captura de ecrã, software, Website

Descrição gerada automaticamente**

Figure 9 – Mockup

# Chapter 5

**Prototype Development**

## Back-End Development

The back-End development was divided into the Database, the API, authentication and test.

### Database

The Database was built using MongoDB. The figure [4.1](#_bookmark26) in the previous chapter the database is composed by 4 collections being them: User, News, Volunteering and Map.

### API and relation with Database

Na API do projeto, foi implementado vários endpoints e foi utilizado os quatro métodos HTTP principais: GET, POST, DELETE e PUT [20]:

* GET: Este método é usado para recuperar dados do servidor¹². É um método somente de leitura, o que significa que não tem risco de alterar ou corromper os dados [20],[21].
* POST: O método POST é usado para enviar dados ao servidor para criar um novo recurso. Ele é usado para enviar dados ao servidor e criar um novo recurso [20],[21].
* DELETE: Este método é usado para remover um recurso existente no servidor[20],[21].
* PUT: O método PUT é usado para atualizar um recurso existente no servidor ou criar um novo recurso se ele não existir [20],[21].

Esses métodos são fundamentais para permitir operações CRUD (Create, Read, Update, Delete) na sua API, permitindo aos usuários interagir efetivamente com os dados [20],[21],[22],[23]

Uma imagem com texto, captura de ecrã, número, Paralelo

Descrição gerada automaticamente

Figure 10 - FastAPI endpoints

#### Connectando à base de dados

Para conectar à base de dados, foi utilizado o pymongo, uma biblioteca Python para interagir com o MongoDB. O pymongo permite que os desenvolvedores Python usem o MongoDB como um sistema de armazenamento de dados de maneira eficiente e intuitiva. A conexão com a base de dados é estabelecida através do MongoClient, uma classe do pymongo. A string de conexão passada para o MongoClient especifica que a base de dados está armazenada localmente, no endereço ‘mongodb://localhost:27017/’. Isso significa que o servidor MongoDB está rodando na mesma máquina onde o código Python está sendo executado. Portanto, todas as operações de leitura e gravação de dados são realizadas localmente, o que pode resultar em um desempenho mais rápido e seguro.

Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente

Figure 11 - Aceder base de dados

#### Encriptando a password

Em qualquer sistema de informação, a segurança dos dados é de extrema importância. Uma das principais formas de garantir essa segurança é através da encriptação de senhas. Foi utilizado a biblioteca passlib para encriptar senhas. Esta biblioteca permite transformar a senha original em uma versão encriptada, conhecida como hash. A encriptação de senhas é importante por várias razões. Primeiro, protege os dados sensíveis do usuário. Mesmo que um invasor obtenha acesso ao banco de dados, ele não será capaz de descobrir a senha original a partir do hash. Segundo, a `passlib` também fornece uma função para verificar se a senha inserida pelo usuário corresponde ao hash armazenado. Isso é feito sem a necessidade de desencriptar o hash, contribuindo para a segurança e eficiência do sistema.

Uma imagem com texto, captura de ecrã, Tipo de letra, software

Descrição gerada automaticamente

Figure 12 - Metodo de encriptação

#### Estrura de Pastas

A estrutura de pastas é um aspecto crucial de qualquer projeto de programação. Ela serve como a espinha dorsal do projeto e desempenha um papel vital na organização dos arquivos e diretórios. Uma estrutura de pastas bem planejada e organizada pode tornar o desenvolvimento do projeto mais eficiente e gerenciável.Uma imagem com texto, captura de ecrã, software

Descrição gerada automaticamente

Figure 13 - Estrututa de Pasta Python

## Front-End Development

In the Front-End part, we used Material UI 5 as User Interface (UI). We started creating several components that can be reused in the several pages existed in the project, such as Navigation Bar, Footer and some forms. Firstly, the website was developed totally static, showing all the functions of the website. Navigation between pages was done using *react-router-dom*.

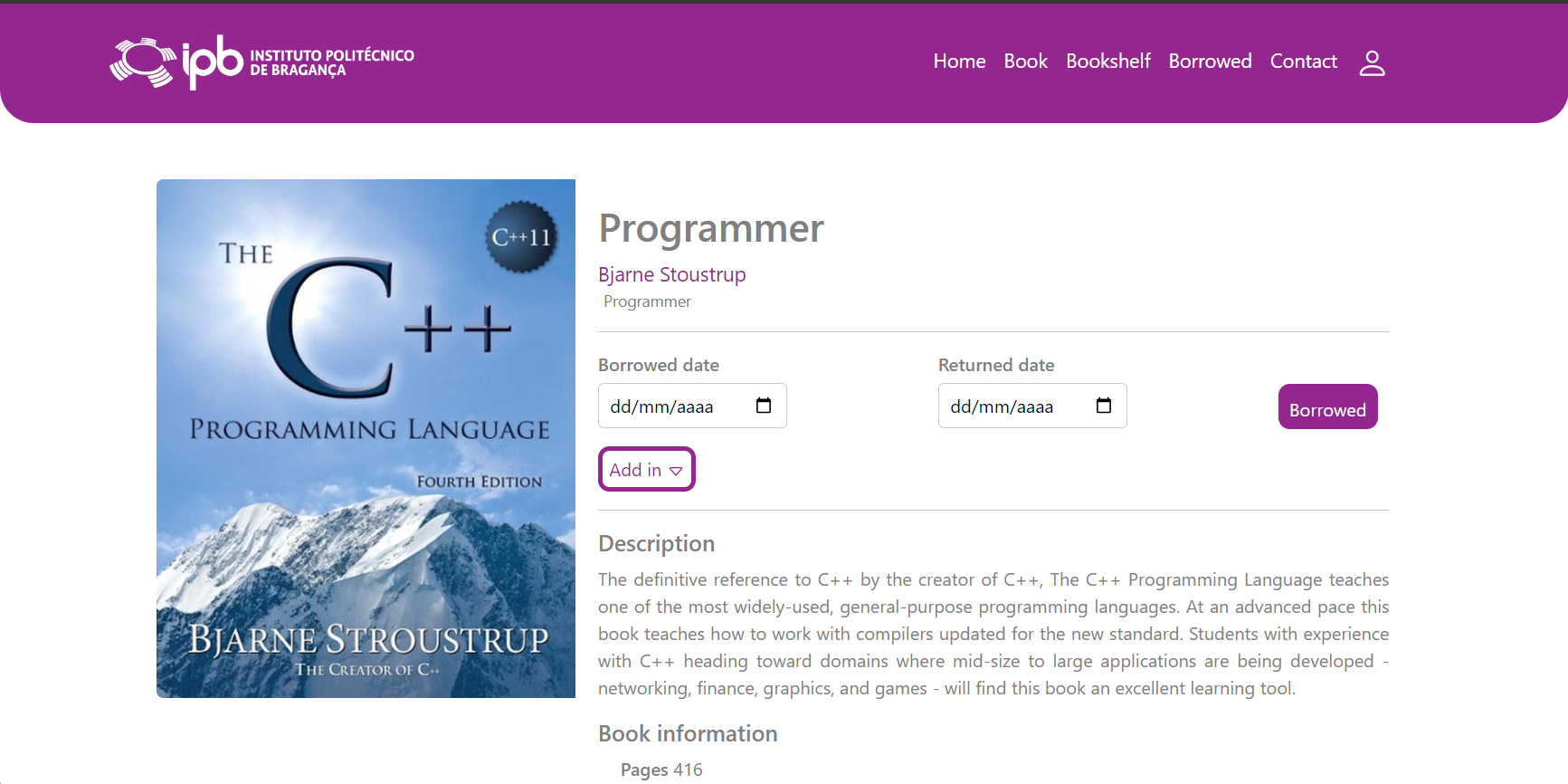


Figure 14 - Website image

# Chapter 6 Conclusions

O IPB Digital Library representa um marco significativo na promoção da leitura e do acesso à informação. Os utilizadores podem facilmente visualizar os livros disponíveis na biblioteca, requisitar livros e criar estantes virtuais privadas, proporcionando uma experiência personalizada e enriquecedora.

Este website não é apenas uma ferramenta para facilitar o acesso aos livros, mas também um espaço onde os amantes de livros podem interagir e partilhar as suas paixões literárias. Através da criação de estantes virtuais privadas, os utilizadores podem organizar os seus livros favoritos e partilhar as suas recomendações com outros.

Apesar de já termos alcançado muitos dos nossos objetivos iniciais, reconhecemos que ainda há espaço para melhorias e expansão. Estamos empenhados em continuar a desenvolver e aprimorar o nosso website para melhor atender às necessidades dos nossos utilizadores e contribuir para a promoção da literatura.

# Bibliography

1. FastAPI. https://fastapi.tiangolo.com/.
2. Using FastAPI to Build Python Web APIs – Real Python. https://realpython.com/fastapi-python-web-apis/.
3. Welcome to Python.org. https://www.python.org/.
4. Download Python | Python.org. https://www.python.org/downloads/.
5. What Is MongoDB? | MongoDB. https://www.mongodb.com/company/what-is-mongodb.
6. MongoDB: The Developer Data Platform | MongoDB. https://www.mongodb.com/.
7. What Is A Technology Stack? Tech Stacks Explained | MongoDB. https://www.mongodb.com/basics/technology-stack.
8. React (JavaScript library) - Wikipedia. https://en.wikipedia.org/wiki/React\_%28JavaScript\_library%29.
9. React. https://react.dev/.
10. Quick Start – React. https://react.dev/learn.
11. An Introduction to JavaScript. https://javascript.info/intro.
12. JavaScript technologies overview - JavaScript | MDN - MDN Web Docs. https://developer.mozilla.org/en-US/docs/Web/JavaScript/JavaScript\_technologies\_overview.
13. What is JavaScript? - Learn web development | MDN. https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First\_steps/What\_is\_JavaScript.
14. Bootstrap · The most popular HTML, CSS, and JS library in the world. https://getbootstrap.com/.
15. Introduction · Bootstrap. https://getbootstrap.com/docs/4.0/getting-started/introduction/.
16. What Is Bootstrap: A Beginner's Guide to Bootstrap - Codecademy. https://www.codecademy.com/resources/blog/what-is-bootstrap/.
17. User Stories | Examples and Template | Atlassian. https://www.atlassian.com/agile/project-management/user-stories.
18. Veja templates e exemplos de histórias de usuário - Atlassian. https://www.atlassian.com/br/agile/project-management/user-stories.
19. User story - Wikipedia. https://en.wikipedia.org/wiki/User\_story.
20. HTTP Methods - REST API Tutorial. https://restfulapi.net/http-methods/.
21. What are GET, POST, PUT, PATCH, DELETE? A walkthrough with ... - Medium. https://medium.com/@9cv9official/what-are-get-post-put-patch-delete-a-walkthrough-with-javascripts-fetch-api-17be31755d28.
22. HTTP Methods for RESTful Services - REST API Tutorial. https://www.restapitutorial.com/lessons/httpmethods.html.
23. HTTP REQUEST Methods-GET, POST, PUT, PATCH, DELETE. | The Startup - Medium. https://medium.com/swlh/restful-api-design-get-post-put-patch-delete-a-walkthrough-with-javascripts-fetch-api-e37a8416e2a0.
24. Use Cases and Scenarios - A Comprehensive Guide. https://www.adaptiveus.com/blog/technique/use-case-and-senarios/.
25. What is a use case? How to write one, examples, + template. https://www.figma.com/resource-library/what-is-a-use-case/.
26. What is a use case? Definition, template, and how to write one. <https://blog.logrocket.com/product-management/what-is-a-use-case-template-how-to-write/>.
27. What Is a Use Case & How To Write One | Wrike. https://www.wrike.com/blog/what-is-a-use-case/.United Nations, *United nations sustainable development goal 1: No poverty*, Website. [Online]. Available: <https://www.un.org/sustainabledevelopment/poverty/>.