

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA
PARAÍBA**

**BOOKIFY: EXPLORANDO O USO DE SISTEMAS DE
RECOMENDAÇÃO.**

DANIEL SILVA NOGUEIRA

Cajazeiras - Paraíba, 2023

DANIEL SILVA NOGUEIRA

BOOKIFY: EXPLORANDO O USO DE SISTEMAS DE RECOMENDAÇÃO.

Trabalho de conclusão de curso apresentado junto ao programa de **Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas** do **Instituto Federal de Educação, Ciência e Tecnologia da Paraíba**, como requisito parcial à obtenção do título de **Tecnólogo em Análise e Desenvolvimento de Sistemas**.

Orientador:

Prof. Me. Diogo Dantas Moreira.

Cajazeiras - Paraíba, 2023

IFPB / Campus Cajazeiras
Coordenação de Biblioteca
Biblioteca Prof. Ribamar da Silva
Catalogação na fonte: Cícero Luciano Félix CRB-15/750

N778b Nogueira, Daniel Silva.
Bookify : explorando o uso de sistemas de recomendação / Daniel
Silva Nogueira.– 2023.

38f. : il.

Trabalho de Conclusão de Curso (Tecnólogo em Análise e
Desenvolvimento de Sistemas) - Instituto Federal de Educação,
Ciência e Tecnologia da Paraíba, Cajazeiras, 2023.

Orientador(a): Prof. Me. Diogo Dantas Moreira.

1. Desenvolvimento de sistemas. 2. Sistema de recomendação. 3.
Sistema intigente. 4. Gestão de acervo. I. Instituto Federal de
Educação, Ciência e Tecnologia da Paraíba. II. Título.

IFPB/CZ

CDU: 004.4(043.2)



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

DANIEL SILVA NOGUEIRA

BOOKIFY: EXPLORANDO O USO DE SISTEMAS DE RECOMENDAÇÃO

Trabalho de Conclusão de Curso apresentado junto ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba - Campus Cajazeiras, como requisito à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador

Prof. MSc. Diogo Dantas Moreira.

Aprovada em: **28 de Agosto de 2023.**

Prof. MSc. Diogo Dantas Moreira - Orientador

Prof. MSc. Ricardo de Sousa Job - Avaliador

IFPB - Campus Cajazeiras

Prof. Antonio Ricart Jacinto de Oliveira Medeiros - Avaliador
IFPB - Campus Cajazeiras

Documento assinado eletronicamente por:

- **Diogo Dantas Moreira**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 01/09/2023 07:50:54.
- **Ricardo de Sousa Job**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 01/09/2023 08:35:27.
- **Antonio Ricart Jacinto de Oliveira Medeiros**, PROF ENS BAS TEC TECNOLOGICO-SUBSTITUTO, em 01/09/2023 09:43:34.

Este documento foi emitido pelo SUAP em 30/08/2023. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código 468409
Verificador: e6c4f62b92
Código de Autenticação:



Rua José Antônio da Silva, 300, Jardim Oásis, CAJAZEIRAS / PB, CEP 58.900-000
<http://ifpb.edu.br> - (83) 3532-4100

*Dedico esse trabalho à minha mãe e meu pai,
pelo apoio e a torcida de sempre.*

AGRADECIMENTOS

Agradeço a Deus pelo dom da vida e com ela poder conquistar objetivos. Agradeço ao professor Diogo por sua orientação ao longo deste trabalho, à minha mãe e meu pai pela torcida nos bastidores, e também aos meus amigos que estiveram envolvidos comigo durante essa construção apoiando e incentivando.

"Você tem poder sobre sua mente – não sobre eventos externos. Perceba isso e você encontrará sua força."

Marco Aurélio, Meditações

RESUMO

Estamos cercados pela tecnologia que cresce exponencialmente cada vez mais dentro dos mais diversos contextos, o que nos submete a sobrecarga de informações e produtos cada vez que acessamos a Internet, com isso os usuários buscam itens e opções que se adéquem às suas preferências particulares e se valem da experiência de interação com outros usuários ou sistemas. Neste contexto entende-se a importância dos Sistemas de Recomendações que exercem um papel fundamental na decisão do usuário, de modo que sugerindo-o itens e conteúdos voltados para seus gostos, torna a experiência com o sistema personalizada. O presente trabalho busca explorar o uso de sistemas de recomendação ao desenvolver um aplicativo no contexto de gestão de livros, com o objetivo dos usuários cadastrarem seus livros pessoais e possibilitar o encontro com outras obras que estejam em acordo com suas preferências com base nos seus gostos pessoais, contribuindo assim com recomendações personalizadas ao usuário além de possibilitar a divulgação de obras pouco conhecidas.

Palavras-chave: Sistemas de recomendações, Sistemas Inteligentes, Gestão de livros.

ABSTRACT

We are surrounded by technology that grows exponentially more and more within the most diverse contexts, which subjects us to an overload of information and products every time we access the Internet, as a result, users look for items and options that suit their particular preferences and they are worth the experience of interacting with other users or systems. In this context, the importance of Recommendation Systems is understood, which play a fundamental role in the user's decision, so that by suggesting items and content geared to their tastes, the experience with the system becomes personalized. The present work seeks to explore the use of recommendation systems by developing an application in the context of book management, with the objective of users registering their personal books and enabling the encounter with other works that are in accordance with their preferences based on their tastes. personal information, thus contributing with personalized recommendations to the user, in addition to enabling the dissemination of little-known works.

Keywords: Recommender Systems, Intelligent Systems, Book Management.

LISTA DE FIGURAS

Figura 2.1 – Listas de recomendações na plataforma Netflix	12
Figura 3.1 – Arquitetura proposta	18
Figura 3.2 – Diagrama de Casos de Uso	20
Figura 3.3 – Bibliotecas	21
Figura 3.4 – População da base de dados	21
Figura 3.5 – Calculando a similaridade entre os gêneros	22
Figura 3.6 – Calculando a similaridade total entre os livros	22
Figura 3.7 – Calculando a similaridade entre os usuários	22
Figura 3.8 – Percorrendo os usuários do BD	23
Figura 3.9 – Gêneros Similares	23
Figura 3.10–Usuários Similares	24
Figura 3.11–Nível de similaridade do conteúdo	24
Figura 3.12–Nível de similaridade do usuário	24
Figura 3.13–Landing Page	25
Figura 3.14–Autenticação	25
Figura 3.15–Cadastro	26
Figura 3.16–Cadastro	26
Figura 3.17–Administração	27
Figura 3.18–Vitrine de Livros	27
Figura 3.19–Recomendações	28
Figura 3.20–Favoritos	28

SUMÁRIO

1	INTRODUÇÃO	6
1.1	Objetivos	7
1.1.1	Objetivos específicos	7
1.2	Organização do trabalho	7
2	REFERENCIAL TEÓRICO	8
2.1	Sistemas de recomendação	8
2.1.1	Histórico	9
2.1.2	Funcionamento dos SR	10
2.1.3	Objetivos dos SR	10
2.1.4	Coleta de informações	11
2.1.5	Estratégias de recomendação	12
2.1.6	Técnicas de recomendação	13
3	APLICAÇÃO BOOKIFY	17
3.1	Requisitos	17
3.2	Arquitetura	18
3.3	Casos de Uso	20
3.3.1	Diagrama de Casos de Uso	20
3.4	Desenvolvimento do Sistema	20
3.4.1	Importações de Bibliotecas	20
3.4.2	Leitura dos Dados	20
3.4.3	Métodos da Recomendação da API	21
3.4.4	Itens Recomendados	22
3.4.5	Métricas	23
3.5	Resultados Obtidos	25
3.5.1	Landing Page	25

3.5.2	Autenticação	25
3.5.3	Cadastro	26
3.5.4	Administração	26
3.5.5	Recomendações	27
3.5.6	Favoritos	28
4	CONSIDERAÇÕES FINAIS	29
	REFERÊNCIAS	30

1 INTRODUÇÃO

A tecnologia está presente a todo instante no cotidiano e com cada vez mais força os sujeitos estão cercados por ela. O mundo apresenta essa experiência desde contextos mais simples como os dispositivos móveis até chegar em contextos mais complexos como a Inteligência Artificial. Um dos modos mais próximos de se ter contato com as tecnologias é através do comércio, que envolve boa parte da vida da população.

De acordo com dados da Confederação Nacional do Comércio de Bens, Serviços e Turismo (CNC), a Intenção de Consumo das Famílias (ICF) – um indicador antecedente do consumo que pode ser utilizado para o planejamento do comércio e de outras atividades produtivas – atingiu 77,6 pontos no mês de fevereiro de 2022, com o segundo aumento mensal consecutivo (+0,4%), o maior nível desde maio de 2020 onde obteve resultado de 81,7 pontos (ICF, 2022). É possível destacar nesse aspecto, a expansão do comércio eletrônico, que cresce constantemente e se expandiu principalmente em tempos de pandemia e distanciamento social (MELO CRUZ, 2021).

Comumente, em sistemas de comércio eletrônico os consumidores se deparam com sugestões sobre itens que estão buscando ou que se adequem ao seu perfil de comprador. Isso pode ocorrer pelas ações dentro do sistema ou por compras de outros usuários com gostos que lhe sejam parecidos. Uma solução que envolva boa interação entre os usuários, e entre o usuário e a máquina, pode ser considerada como peça chave para o mercado de trabalho, pois quanto menos tempo o cliente passar para obter sua solução, mais eficaz se torna o processo de resposta para ele. Nesse contexto se destaca a importância dos sistemas de recomendações (SR).

Os Sistemas de Recomendação podem ser definidos como mecanismos de personalização de conteúdo, onde através de critérios como preferência do usuário, semelhança entre conteúdos acessados, avaliação de outros, etc., são feitas sugestões entre usuários (ADOMAVICIUS; TUZHILIN, 2005).

Considerando que as recomendações têm um papel fundamental na decisão de compra dos usuários ou até mesmo os influencia a consumir conteúdos que são semelhantes aos seus gostos, pode-se entender a dimensão dos SR, seja no mundo digital (onde se tem *e-commerces*, *aplicações com muitos usuários* que irão auxiliar os usuários a compreenderem melhor o que desejam, terem uma melhor experiência de navegação, e conseqüentemente obterem resultados personalizados às suas preferências), seja em plataformas de relacionamentos entre os usuários, como por exemplo as redes sociais, que

conectam pessoas com gostos, atividades, *hobbies* similares e etc.

O presente trabalho tem como objetivo desenvolver um sistema de recomendação de livros com o intuito de experimentar uma estratégia de sistemas de recomendação, num ambiente de usuários inseridos e novos usuários que irão se cadastrar no sistema.

Um sistema que possua uma estratégia de recomendação por trás possibilita maior facilidade para seus usuários, viabilizando o encontro de livros que estejam de acordo com seus gostos e interesses. O sistema contribui com alternativas personalizadas que são apropriadas para o perfil do usuário, gerando economia de tempo com as sugestões feitas com base nos seus gostos pessoais.

1.1 OBJETIVOS

O objetivo geral deste trabalho é desenvolver um sistema de recomendação que possibilite o cadastro de usuários e seus livros, e utilize-se de uma estratégia de recomendação híbrida para obtenção de outros livros da base de dados para o usuário.

1.1.1 Objetivos específicos

- Fazer um levantamento bibliográfico sobre SR;
- Realizar levantamento de requisitos do domínio;
- Planejar um modelo de recomendação para o domínio escolhido;
- Projetar e implementar aplicação seguindo a estratégia de recomendação adotada.
- Recomendar itens ao usuário, oriundos do resultado da estratégia abordada.

1.2 ORGANIZAÇÃO DO TRABALHO

As seções seguintes estão divididas em três capítulos. O capítulo 2 que irá descrever a fundamentação teórica do trabalho, discorrendo sobre a presença da tecnologia no meio social e a importância dos sistemas de recomendações neste contexto, bem como conceitos sobre sistemas de recomendações que fornecem uma base para construção do projeto proposto pelo trabalho. O capítulo 3 aponta a proposta do trabalho, indicando a definição dos escopo para a implementação do projeto, os requisitos funcionais, o projeto arquitetural e as tecnologias usadas. Por fim, o capítulo 4 expõe as considerações finais do projeto e o anseio almejado pelos seus resultados.

2 REFERENCIAL TEÓRICO

2.1 SISTEMAS DE RECOMENDAÇÃO

Na era da informação, é possível perceber uma gama cada vez maior de opções que se apresentam aos indivíduos. Em um sistema de *ecommerce*, por exemplo, o usuário é exaustivamente apresentado a uma sucessão de itens, pelos mais variados motivos: itens mais comprados, itens novos, etc. Cada pessoa em sua individualidade pode realizar suas escolhas diante dessas opções, porém, em situações novas é necessário usar da experiência particular - mais precisamente, de seu interesse - para chegar a essa decisão. Diversos são os meios pelos quais uma pessoa pode chegar ao seu interesse e, nesse contexto, se destaca a recomendação como uma das alternativas (CAZELLA; NUNES; REATEGUI, 2010).

A recomendação se mostra como um fator que envolve confiabilidade, posto que a opinião, avaliação, comentários e compartilhamento interpessoal de serviços ou produtos demonstram maior chance de aprovação. Isso pode ser exemplificado quando, na busca por um filme, se buscam as avaliações ou comentários: o fato de as pessoas recomendarem positivamente o produto acaba gerando maior interesse de consumo (ZYLBERSZTEJN, 2012). Assim, um Sistema de Recomendação é uma alternativa eficaz no momento de busca e escolha de um item.

Os Sistemas de Recomendação podem ser definidos como mecanismos de personalização de conteúdo, onde através de critérios como preferência do usuário, semelhança entre conteúdos acessados, avaliação de outros, etc., são feitas sugestões entre usuários (ADOMAVICIUS; TUZHILIN, 2005). Esses sistemas corroboram e impactam de maneira positiva na interação humano-computador, na medida que impulsionam a capacidade dos usuários de passarem menos tempo buscando por algum produto que seja compatível com suas preferências.

Estes sistemas auxiliam na eficácia do processo de indicação que já ocorre naturalmente fora dos espaços tecnológicos, além de ajudar os usuários a decidir um produto quando lhe falta experiência ou quando há um grande número de opções (SHRIVER, 2018). Segundo Cazella, Nunes e Reategui (2010), um típico sistema de recomendação funciona com o fornecimento de informações de entrada que o mesmo agrega e encaminha para indivíduos considerados potenciais dentro daqueles interesses. Porém, o desafio consiste neste mesmo ponto: realizar a combinação adequada entre a expectativa do usuário e os produtos e serviços a serem recomendados para o mesmo.

A evolução constante da *Web* e das tecnologias têm permitido avanços cada vez

maiores, e o desenvolvimento e aprimoramento dos sistemas de recomendação é uma realidade causada por essa força motriz dinamizadora de todo esse processo. Através desses sistemas é permitido ao usuário ter uma experiência personalizada de encontrar seu caminho em meio a grandes volumes de informações que estão espalhadas e dispostas ao seu redor, por um lado filtrando o que mais lhe agrada e por outro o sistema sugerindo itens considerando ou deduzindo suas preferências (gostos, interesses e prioridades). Exemplos não faltam, desde plataformas de streaming de vídeo, como por exemplo *Netflix*, *Youtube*, *Prime Vídeo*, passando por plataformas sociais como *Twitter*, *Facebook* e *Instagram* e até *e-commerces* como por exemplo, *Submarino* e *Amazon*. Alguns outros exemplos menos conhecidos podem ser encontrados em (ROCHA; Catarina, 2003) com o *RecDoc*, um sistema de recomendação para uma biblioteca digital na *Web*, um sistema desenvolvido para apoiar os usuários a encontrarem documentos relevantes na biblioteca baseado nas referências bibliográficas armazenadas em seu perfil.

2.1.1 Histórico

Primariamente, em 1990, foi desenvolvido um embrião do que mais tarde viria a se desenvolver de forma mais completa nos sistemas de recomendações, quando o sueco Jussi Karlgren elaborou um relatório relacionada a um trabalho acadêmico, onde neste relatório o intuito era melhorar a interação humano-computador com uma álgebra para recomendações (KARLGREN, 1990), modelo esse elaborado por Jussi, que em 1994 foi publicado depois de ser revisto e aprimorado (KARLGREN, 1994).

Contudo, um pioneiro sistema de recomendação completo foi desenvolvido e publicado em 1992 por David Goldberg, David A. Nichols, Brian M. Oki e Douglas B. Terry, denominado como *Tapestry* (GOLDBERG, 1992), com o intuito de fazer com que fosse aprimorado o rendimento no processo de indicação de itens aos usuários, seguindo a filosofia de que “os humanos fornecem a mais confiável avaliação de documentos” (GOLDBERG, 1992, p. 61-70). O *Tapestry* era um software projetado para fazer a recomendação de documentos que eram extraídos de coleções de notícias, essa recomendação era feita para um público restrito de usuários através dos seus emails. Os pioneiros em sistemas de recomendação forjaram o termo “filtragem colaborativa” para representar o sistema específico, no qual a colaboração dos usuários seria o ponto chave para a filtragem da informação, onde até então uma enorme quantidade de *emails* com notícias lhes eram entregue mas não era feita nenhuma filtragem para esse envio, e a partir dessa abordagem que foi criado o *Tapestry*.

2.1.2 Funcionamento dos SR

Os sistemas de recomendações agem filtrando o grande volume de informações que nos cercam para poder indicar aquilo que mais interessa ao usuário, mas para que esse filtro e essa sugestão ocorra dessa forma personalizada, ele se vale de um conjunto de dados para poder realizar a assimilação entre eles. Todo esse conjunto de dados, fazem referência a três objetos: Itens, Usuário e Transações (Ricci et al, 2011).

Itens Os itens podem ser quaisquer objetos (música, livro, produto de beleza, roupa, etc.) dependendo apenas de qual o contexto que o SR é utilizado. Cada item possui um valor positivo ou negativo, e esse valor é o que irá pesar na recomendação final para o usuário

Usuários Para o sistema recomendar itens para o usuário, é preciso obter dele informações relevantes para a análise, e assim gerar uma recomendação mais personalizada voltada para suas preferências. Quanto mais informações o sistema obtiver do usuário, mais personalizada será a recomendação.

Transações As transações são os registros de atividade do usuário quando ele interage com o sistema (compras, visualizações, curtidas, comentários, etc.) esse registro serve para que o algoritmo crie um comportamento do usuário.

2.1.3 Objetivos dos SR

Os sistemas de recomendação estão presentes nos mais variados segmentos: redes sociais, aplicativos, *e-commerce*, *streams*, etc, onde o objetivo é fazer com que o usuário tenha uma boa experiência com o conteúdo que é consumido. Para que se possa chegar à esse objetivo comum, há outros operacionais e técnicos ao qual um SR deve atender para sua melhor eficácia, são eles:

Relevância Os usuários ao terem contato com produtos e serviços, tendem a escolher aqueles que acham mais interessante. A relevância sozinha não é suficiente para o bom êxito de um SR apesar de sua primazia, por isso há outros objetivos operacionais que em consonância com ela trazem uma eficácia maior.

Novidade A novidade retrata a capacidade do SR de mostrar ao usuário algo que ele ainda não tenha visto anteriormente, evitando assim uma recomendação repetitiva e entediante, o que pode vir a causar um efeito contrário ao de lucro

Serendipidade Um outro objetivo capaz de surpreender e converter positivamente é o da serendipidade, embora seja um pouco diferente do que diz a novidade pelo fato

das recomendações serem surpresas para o usuário, ao invés de trazerem algo já conhecido. Podemos enxergar esse objetivo, quando em algum caso uma lista de itens específicos está sendo recomendada a um usuário e há nela um elemento novo não esperado por ele, o fato de conhecer algo novo pode fazê-lo perceber um interesse latente por determinadas preferências.

Aumento de diversidade A diversidade garante que haja alguns itens de tipos diferentes na recomendação para o usuário, fazendo assim com a recomendação repetitiva e ele não corra o risco de ficar entediado, conseqüentemente não venha ter um impacto negativo no lucro.

2.1.4 Coleta de informações

Para poder ser possível um sistema de recomendação atingir seus objetivos e entregar uma experiência personalizada para o usuário, é preciso que se obtenha dele informações pessoais e até mesmo comportamentais, a fim de que se possa filtrar as informações de entrada e gerar uma recomendação voltada para suas preferências em específico, ao invés de uma recomendação genérica. Para isso, habitualmente temos 2 métodos de identificação do usuário que nos permite coletar informações sobre o mesmo, são eles: Identificação no servidor e Identificação no cliente (Edgar Bisset et al, 2016).

No método de **identificação no servidor**, o usuário irá conceder suas informações pessoais (nome, idade, sexo, endereço, etc.) que serão persistidas em um base de dados para identificação posterior daquele usuário. Será associado a ele um login e uma senha para que o sistema com base nas suas informações cadastradas possa o reconhecer cada vez que solicitar acesso. Usualmente essas informações são captadas através de formulários de cadastros nas plataformas.

Na **identificação no cliente** as informações dos usuários são, geralmente, captadas através dos *cookies*, onde pequenos arquivos são criados e armazenados no computador do usuário com as informações que posteriormente o sistema usará como filtro para fazer a recomendação. Embora esse método não seja o mais seguro para identificar um usuário, pois ao salvar as informações o sistema compreende que elas estão associadas a determinado usuário, quando na verdade o funcionamento dos *cookies* armazena as informações concedidas pelo dispositivo, se houver um caso de vários usuários compartilharem o mesmo dispositivo, o filtro da informação será genérico para todos os usuários.

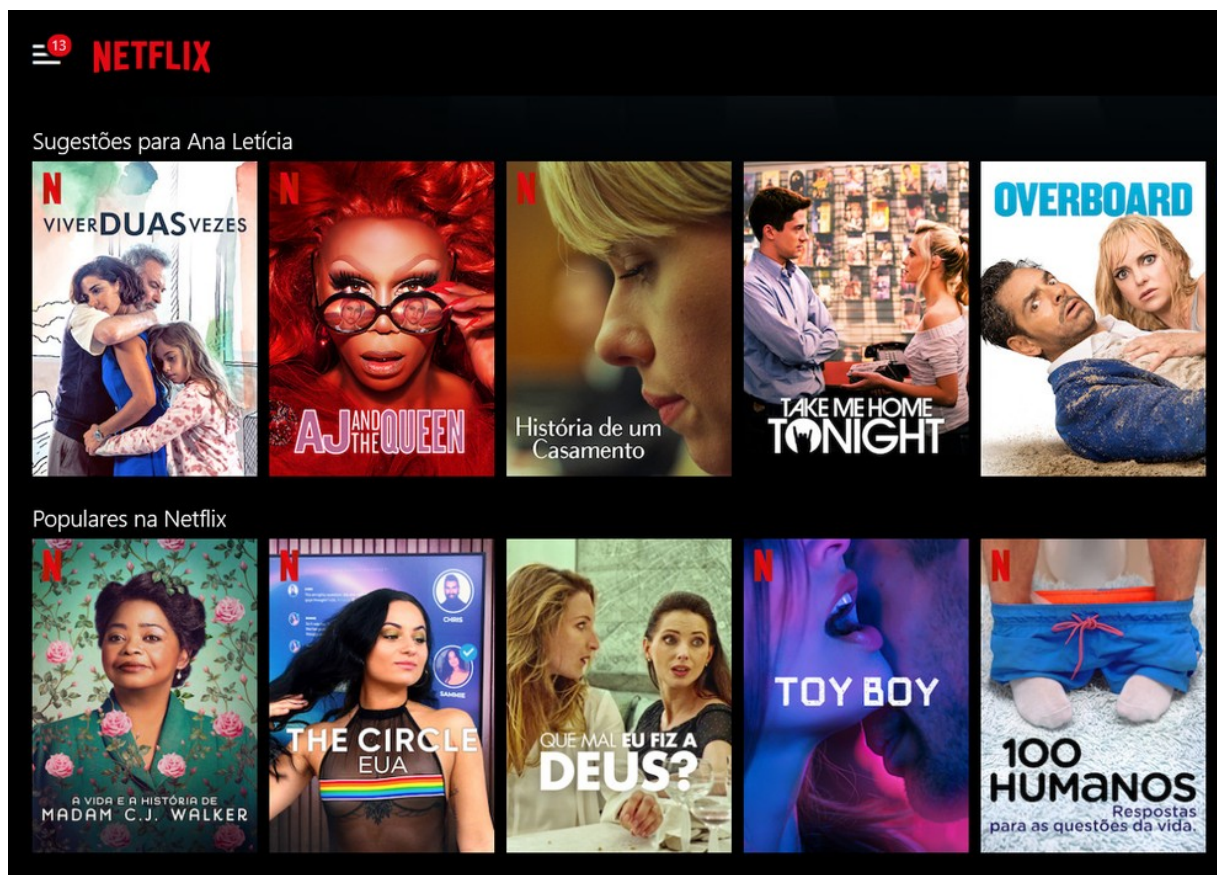
Uma vez que conseguimos coletar informações sobre o usuário, precisamos definir quais produtos/serviços serão mais adequados para a recomendação. Para isso precisaremos de estratégias e técnicas de recomendação, que abordaremos nos tópicos subsequentes.

2.1.5 Estratégias de recomendação

Uma vez que o sistema conseguiu identificar o perfil do usuário e filtrar suas informações necessárias, é o momento de se traçar uma estratégia de recomendação que servirá de base para a aplicação de uma ou mais técnicas para ter uma recomendação adequada ao perfil individual do usuário (Edgar Bisset et al, 2016). São várias as estratégias usadas por um sistema de recomendação, dentre as quais podemos citar:

Na estratégia de **Listas de recomendação** os itens a serem recomendados aos usuários são organizados por contextos específicos de interesses (ano novo, dia dos namorados, natal, etc.), trazendo assim à essa estratégia uma vantagem na hora da implementação tendo em vista dela não dispor de tanta robustez, porém como os itens são dispostos por contextos pré-determinados aos usuários não há uma recomendação personalizada, mas sim uma recomendação genérica a todos usuários. A Figura 2.1, retirada do site Techtudo¹ nos ilustra melhor sobre essa estratégia.

Figura 2.1 – Listas de recomendações na plataforma Netflix



Fonte: Techtudo (2020)

Já na estratégia **Avaliações dos usuários** os usuários ao acessarem o sistema

¹ Como achar filmes na Netflix? | Áudio e Vídeo | TechTudo. Disponível em: <<https://www.techtudo.com.br/listas/2020/03/como-achar-filmes-na-netflix-cinco-dicas-para-encontrar-novos-titulos.ghtml>>

podem avaliar determinado produto e possivelmente fazer um breve comentário a respeito. Essa estratégia é comumente usada, principalmente em e-commerces, quando a validação dos demais usuários pode ser fator decisivo na compra de determinado produto ou serviço (Julia Fialho, 2016). Como as listas de recomendações, não possui uma implementação complicada, tendo em vista que não depende de dispositivos terceiros inteligentes e utilizando-se apenas do armazenamento das avaliações e de sua mostra no momento mais oportuno, porém a sua confiabilidade depende da veracidade das informações prestada pelos usuários, o que torna a recomendação boa a outrem um pouco dependente de quem avaliou determinado produto/serviço.

Na estratégia **Suas recomendações**, estas são organizadas em uma seção, onde são particularmente dispostas para aquele usuário específico. Essas recomendações são feitas a partir de informações filtradas implicitamente (através das ações do usuário dentro do sistema) ou explicitamente (através do perfil do usuário).

Na **Recomendação através de gostos em comum**, uma das mais complexas, consiste na recomendação a partir da associação, na base de dados, das avaliações dos itens realizados pelos usuários. Como a estratégia de avaliação de usuário, essa também é muito comum em *e-commerces*, podendo existir a possibilidade de se usarem as duas em conjunto.

Por fim, temos a **Associação por conteúdo**, nessa estratégia, as recomendações aqui são feitas baseadas no conteúdo do item, também habitualmente usado em *e-commerces*. Geralmente quando itens são vendidos em conjunto para uma grande parte dos usuários. Por exemplo, quando dois itens X e Y são comumente vendidos juntos, o sistema passará a recomendar o produto Y aos usuários que adquiriram o produto X .

2.1.6 Técnicas de recomendação

Após termos apresentado algumas das estratégias que são utilizadas para recomendação, abordaremos agora técnicas que são utilizadas para filtragem das informações, segundo Belvin e Croft (1992), a filtragem da informação descreve a diversidade dos processos envolvidos na entrega da informação para os usuários. Segundo Santana (2018), dentre as técnicas de recomendação podemos citar as mais conhecidas: Filtragem baseada em conteúdo, filtragem colaborativa, filtragem híbrida.

Filtragem baseada em conteúdo. As abordagens baseadas em conteúdo funcionam a partir do conhecimento que o sistema tem das características dos itens, com isso a recomendação será gerada através da similaridade com itens avaliados anteriormente pelos usuários de maneira positiva. Através das técnicas de recuperação da informação são

extraídas as características dos itens, porém há casos em que a informação textual sobre o item não é de fácil acesso como no caso de áudios, imagens e vídeos, em contextos assim a informação pode ser extraída de metadados, os metadados são atributos que caracterizam os itens que são atribuídos ou por especialistas na área ou então por usuários num ambiente *web*.

As recomendações na filtragem por conteúdo são geradas a partir do cálculo da similaridade entre os itens, umas das métricas mais conhecidas é a Similaridade por cosseno (Baeza-Yates, 2011), a escolha da equação dentre outras métricas possíveis a se utilizar, passa pelo critério da Similaridade do Cosseno quantificar a semelhança entre vetores em um espaço multidimensional, quanto mais próximos os vetores estão na direção maior será a similaridade por cosseno entre eles, é frequentemente usada para medir a similaridade entre documentos. Podemos ver a seguir na Equação 1.

$$\text{Cos}(x, y) = \frac{x \cdot y}{\|x\| \cdot \|y\|} \quad (1)$$

Após ter sido calculado a similaridade entre os itens é preciso ordenar os itens avaliados para obter listas de preferências, muitos sistemas que empregam a abordagem de conteúdo, utiliza-se da medida estatística TF-IDF (Equação 2), que indica a importância de um termo em um documento com relação à coleção de documentos (Rajaraman Ullman, 2011). Com o resultado dos cálculos obtêm-se um conjunto de itens ordenados pelo grau de similaridade entre eles. A escolha da medida para o presente trabalho apoia-se no critério do objetivo da mesma, de encontrar documentos semelhantes em um determinado contexto de consulta.

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right) \quad (2)$$

Apesar de ser uma técnica muito conhecida e usada, a filtragem por conteúdo apresenta alguns desafios, no que tange a recomendação, são eles:

Novo usuário Para que o sistema entenda as preferências do usuário e assim gere recomendações personalizadas ao seu perfil, o usuário precisa avaliar ou ter um número suficiente de itens. Um novo usuário com poucos itens receberá recomendações menos personalizadas.

Especialização Pode haver uma limitação no leque de opções do usuário, caso ele avalie somente itens de um determinado contexto.

Por outro lado, há uma maior transparência, de modo que o usuário entenda o porquê aquele conteúdo está sendo recomendado a ele aliado a uma independência do *feedback* de outros usuários.

Filtragem Colaborativa. A filtragem colaborativa surgiu com a proposta de preencher lacunas abertas na filtragem baseada em conteúdo (CAZELLA, 2006), essa técnica se diferencia da anterior pelo fato de não ser necessário compreender ou reconhecer o conteúdo dos itens. Nessa abordagem o ponto chave consiste na troca de experiências entre usuários com gostos em comum. Se um usuário A interagir e avaliar os itens X , Y e Z , é bem provável que um usuário B que avaliou os itens X e Y também goste do item Z .

O processo de abordagem da técnica abrange, calcular a similaridade entre os usuários e estimar se um determinado item está atrelado a suas preferências. Sistemas mais avançados que utilizam essa abordagem são capazes de descobrir a relação entre usuários fundamentado na descoberta de padrões comuns de comportamento, fazendo assim uso de métricas para medir a similaridade de um usuário alvo com os seus vizinhos. As técnicas de filtragem colaborativas podem ser divididas em:

Filtragem Baseada em Memória Consideram métricas como avaliações dos usuários sobre os itens para calcular a similaridade entre os usuários ou itens, bem como também similaridade entre usuários.

Filtragem Baseada em Modelo Os algoritmos são baseados em modelos que atuam nas avaliações dos usuários sobre os itens, de forma a treinar um modelo capaz de efetuar predições.

Filtragem Baseada em Vizinhança É utilizado a similaridade usuário-usuário ou item-item para fazer recomendações a partir da matriz usuário-item. Na similaridade usuário-usuário é computada uma matriz de similaridade entre todos os usuários, considerando cada usuário como um vetor de tamanho n , e depois computando a similaridade entre todos os usuários. Na similaridade item-item considera-se itens que sofreram avaliações semelhantes por um mesmo grupo de usuários, semelhante à usuário-usuário, cada item é considerado como um como um vetor de tamanho n , e depois computa-se a similaridade entre os itens.

Como vimos a filtragem colaborativa parte da premissa de gerar recomendações de itens desconhecidos a um usuário, levando em consideração que tais itens foram relevantes para outros usuários com preferências em comum. Para gerar as recomendações é preciso calcular a similaridade entre os usuários, e para tal usa-se uma métrica de similaridade

que meça a distância entre os usuários para poder prever uma recomendação, uma das métricas mais conhecidas é a Similaridade por cosseno (Equação 1).

Apesar de sua robustez, a filtragem colaborativa apresenta alguns desafios em seu uso, como por exemplo:

Esparsidade dos dados Em base de dados com um volume muito grande de dados, teria que ser feita a similaridade de um usuário alvo com todos os outros.

Memória À medida que aumenta o número de itens avaliados pelo usuário ou itens de uma coleção pessoal, mais recurso computacional será exigido para gerar recomendações.

Mas ao contrário da Filtragem Baseada em conteúdo, essa abordagem traz a vantagem não depender do conteúdo do item para gerar suas recomendações, já que parte das avaliações de outros usuários para tal ou comportamentos e ações anteriores de outros usuários semelhantes.

Filtragem Híbrida. Sistemas com filtragem híbrida buscam conciliar os pontos fortes das outras duas técnicas para criar um sistema que melhor atenda as necessidades dos usuários. Essa abordagem busca unir as vantagens de cada uma das demais vantagens, para assim poder eliminar as fraquezas do sistema.

Os sistemas de recomendações, conforme foi demonstrado na seção 2.1, se demonstram como ferramentas que estão presentes em nosso cotidiano com o objetivo de auxiliar-nos em meio a tanta demanda de informações que somos submetidos, para assim nos mais diversos meios se ter uma experiência personalizada. Mas a recomendação se mostra para nós como um fator que envolve a confiabilidade posto que somos apresentados ao até então desconhecido que veio até nós por meio da experiência e ações de outrem, para isso é importante que os sistemas de recomendações gerem e entreguem recomendações compatíveis com as preferências de cada um em particular e assim poder nortear o usuário em meio à sobrecargas de informações.

3 APLICAÇÃO BOOKIFY

No presente trabalho será abordado uma aplicação contando com o sistema de recomendação de livros, objetivando explorar uma estratégia de recomendação híbrida. Para tal, foi pensado que o sistema permita que o usuário faça seu cadastro e adicione livros à sua lista pessoal, com base em um banco de dados de livros, populado utilizando-se da abordagem *ETL* para evitar um cadastro manual de, por exemplo, mais de 1000 exemplares. Embasado no cálculo de similaridade, o sistema irá recomendar para o usuário livros que não estão presentes em sua coleção, com base em gostos filtrados no momento do cadastro, abordando assim o método de identificação no servidor e evitando o *cold start*.

3.1 REQUISITOS

O quadro 3.1 descreve os requisitos funcionais do sistema proposto. Foi utilizada a seguinte nomenclatura: [RFXX], para enumerar e descrever os requisitos. Os requisitos funcionais elencados descrevem em um alto nível a implementação das funcionalidades do sistema que se almeja implementar neste trabalho.

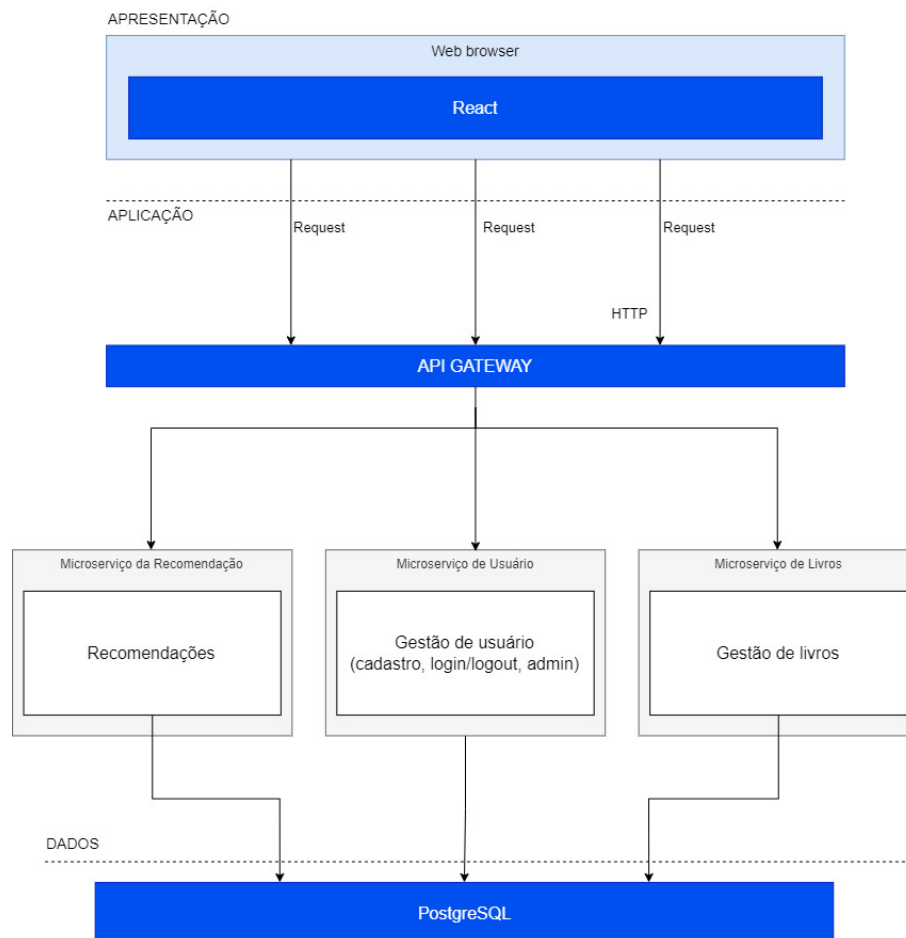
Quadro 3.1 – Requisitos funcionais da proposta

Código	Nome	Descrição
RF01	Cadastro de usuário	O sistema deve permitir o cadastro dos usuários e dos seus livros à coleção, pelo menos três livros.
RF02	Captação de preferência	O sistema deve submeter o usuário a uma pré-lista de livros para que o mesmo adicione à sua lista no momento do cadastro, tendo em vista suprimir o <i>cold start</i> .
RF03	Livros Populares	O sistema deve exibir ao usuário livros populares cadastrados na base de dados, tomando por métrica as melhores avaliações dos livros.
RF04	Adição de livros à coleção	O sistema deve permitir a adição dos livros que são exibidos nas vitrines à lista pessoal de livros do usuário.
RF05	Adicionar livros à coleção	O sistema deve permitir a adição de livros à lista de Favoritos
RF06	Recomendações de livros	O sistema deve realizar recomendações de livros com base na(s) estratégia(s) abordadas.

3.2 ARQUITETURA

Nesta seção será apresentado o projeto arquitetural que foi utilizado para a implementação do sistema, apresentado na Figura 3.1. O mesmo está dividido em três camadas: **Apresentação**, **Aplicação** e **Dados**.

Figura 3.1 – Arquitetura proposta



Fonte: Elaborado pelo autor (2022)

Na **camada de apresentação** é encapsulada toda lógica da interface construída utilizando *React Js*, uma biblioteca *JavaScript* para criação de UI (*User Interface*), a escolha desta tecnologia passa pelo seu conceito de sincronização e facilidade de conexão entre as páginas. Nesta camada serão consumidos os serviços que serão enviados da camada de aplicação via protocolo HTTP e expostos para o usuário.

A **camada de aplicação** trata de resolver as regras de negócios, ela é responsável por receber e enviar requisições para camada de Apresentação através de uma *API REST*, que foi desenvolvida seguindo uma arquitetura de microserviços e um gateway que recebe as requisições e as distribui para o serviço responsável. Para tal foi utilizado o *Express* um framework para aplicativo da web do Node.js, e *FastAPI*, um *framework Python* moderno

de código aberto utilizado para criação de APIs Restful. Dentro desta camada teremos os seguintes módulos:

Recomendações Este módulo engloba o endpoint do sistema responsável pela recomendação, para tal foi adotada uma estratégia *híbrida*. Tendo em vista a adoção da estratégia, utilizou-se de uma Filtragem Baseada em Conteúdo responsável por calcular a similaridade entre dois livros com base em suas características como gênero, autor, título e avaliação; e uma Filtragem Colaborativa responsável por calcular a similaridade entre dois livros envolvendo a análise de padrões de comportamento entre diferentes usuários, a partir da semelhança entre estes.

Gestão de usuários Este módulo descreve a administração das tarefas dos usuários, bem como o cadastro no sistema e autenticação por meio de email e senha.

Gestão de livros O módulo de gestão de livros trata da administração por parte do usuário acerca da adição de livros à sua coleção, adição de livros aos favoritos, e acesso a área de recomendação.

Por fim, a **camada de dados** responsável por fazer a persistência dos dados, que serão enviados da camada de Aplicação, para o *PostgreSQL* - um sistema gerenciador de banco de dados objeto relacional - que em conjunto com a API, irá fazer a persistência e para a recomendação para os usuários.

Logo abaixo algumas serão descritas algumas das tecnologias usadas para o desenvolvimento do estudo.

Python Desenvolvida por Guido Van Rossum, matemático holandês, no início dos anos 90, o *Python* é uma linguagem de programação de alto nível - *High Level Language* -, dinâmica e orientada a objetos. Podendo ser usada em diversos contextos dos quais podemos citar *Scripting* e automação, desenvolvimento web (com o uso de *frameworks* como *Django*, *Flask* e *Fast API*) e *Big Data*.

Numpy *Numpy* é uma poderosa biblioteca da linguagem de programação Python, que consiste em objetos chamados de arrays (matrizes), que são multidimensionais. Além disso, essa biblioteca vem com uma coleção de rotinas para processar esses arrays.

FastAPI *FastAPI* é uma estrutura da Web moderna para criar APIs RESTful em Python, de forma leve e eficaz.

scikit-learn Uma biblioteca de aprendizado de máquina de código aberto para a linguagem de programação Python.

Node.js Software de código aberto, multiplataforma, baseado no interpretador V8 do Google que permite a execução do JavaScript fora de um navegador web.

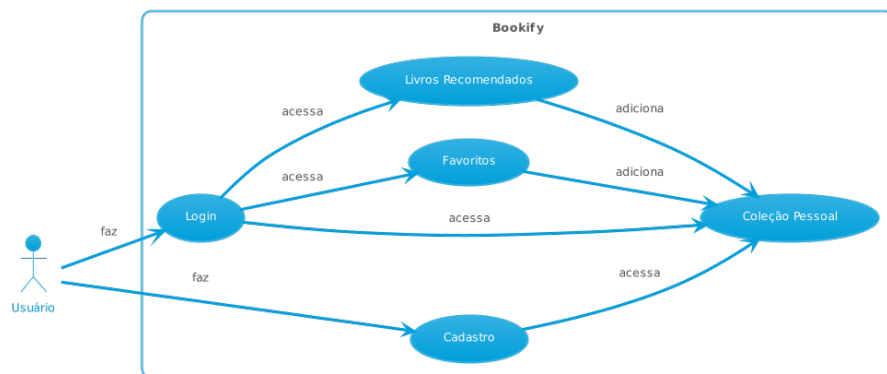
React Biblioteca JavaScript de código aberto para criação de interfaces de usuário componentizadas e reutilizáveis em páginas web.

3.3 CASOS DE USO

3.3.1 Diagrama de Casos de Uso

Na Figura 3.2 observa-se o Diagrama de Casos de Uso descrevendo as funcionalidades e interações do usuário com o sistema.

Figura 3.2 – Diagrama de Casos de Uso



Fonte: Elaborado pelo autor (2023)

3.4 DESENVOLVIMENTO DO SISTEMA

O sistema de recomendação desenvolvido se baseia na abordagem de uma estratégia híbrida, usando a Filtragem Baseada em Conteúdo para calcular a similaridade entre livros com base em suas características, e a Filtragem Colaborativa para calcular a similaridade entre o usuário atual e outros usuários do banco de dados, tomando por base os gêneros dos livros que ele possui.

3.4.1 Importações de Bibliotecas

Na Figura 3.3 estamos importando as bibliotecas: *sklearn* comumente usada em Machine Learning, de onde importamos as funções de cálculo da *Similaridade do Cosseno* e *TF-IDF*, e a *FastAPI* usada para criação da API RESTful.

3.4.2 Leitura dos Dados

O conjunto de dados foi extraído do *Kaggle* consistindo em um documento .csv, onde através da propriedade *COPY* do SQL foi feito um processo de *ETL* a fim de carregar

Figura 3.3 – Bibliotecas

```

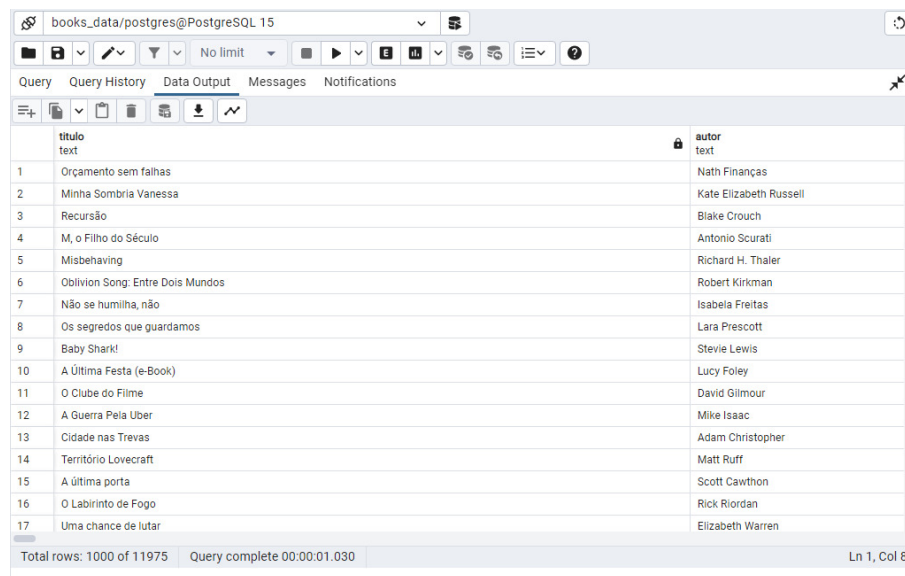
from fastapi import FastAPI, Request
from fastapi.middleware.cors import CORSMiddleware
import numpy as np
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.feature_extraction.text import TfidfVectorizer

```

Fonte: Elaborado pelo autor (2023)

os dados do arquivo de forma automática em uma Tabela no PostgreSQL, conforme mostra a Figura 3.4

Figura 3.4 – População da base de dados



	titulo text	autor text
1	Orçamento sem falhas	Nath Finanças
2	Minha Sombria Vanessa	Kate Elizabeth Russell
3	Recursão	Blake Crouch
4	M, o Filho do Século	Antonio Scurati
5	Misbehaving	Richard H. Thaler
6	Oblivion Song: Entre Dois Mundos	Robert Kirkman
7	Não se humilha, não	Isabela Freitas
8	Os segredos que guardamos	Lara Prescott
9	Baby Shark!	Stevie Lewis
10	A Última Festa (e-Book)	Lucy Foley
11	O Clube do Filme	David Gilmour
12	A Guerra Pela Uber	Mike Isaac
13	Cidade nas Trevas	Adam Christopher
14	Território Lovecraft	Matt Ruff
15	A última porta	Scott Cawthon
16	O Labirinto de Fogo	Rick Riordan
17	Uma chance de lutar	Elizabeth Warren

Fonte: Elaborado pelo autor (2023)

3.4.3 Métodos da Recomendação da API

Na Figura 3.5 é mostrada a primeira função que calcula a similaridade entre dois conjuntos de gêneros de livros, utilizando a medida de similaridade do cosseno, ajustada com base na densidade do usuário utilizando a técnica de interpolação da média, para evitar o impacto desproporcional de usuários com menos dados nas recomendações.

A próxima função calcula a similaridade do conteúdo do livro com base em características adicionais, como gênero, autor, título e avaliação. Após isso é feita uma normalização dos valores entre 0 e 1 para garantir que as características adicionais tenham o mesmo peso na similaridade total entre dois livros. Então a similaridade total entre os livros é calculada como uma média ponderada das similaridades individuais, conforme mostra a Figura 3.6

Figura 3.5 – Calculando a similaridade entre os gêneros

```
def genre_similarity(genres1, genres2, user_density, mean_similarity):
    all_genres = sorted(list(set(genres1 + genres2)))
    vector1 = [1 if genre in genres1 else 0 for genre in all_genres]
    vector2 = [1 if genre in genres2 else 0 for genre in all_genres]
    similarity = cosine_similarity([vector1], [vector2])[0][0]
    correction_factor = user_density / mean_similarity
    similarity_corrected = similarity * correction_factor
    return similarity_corrected
```

Fonte: Elaborado pelo autor (2023)

Figura 3.6 – Calculando a similaridade total entre os livros

```
def content_similarity(book1, book2, user_density, mean_similarity):
    genres_similarity = genre_similarity(book1["genero"],
                                        book2["genero"], user_density, mean_similarity)
    author_similarity = 1 if book1["autor"] == book2["autor"] else 0
    title_similarity = 1 if book1["titulo"] == book2["titulo"] else 0
    # Normalizando a classificação para um valor entre 0 e 1
    rating_similarity = abs(book1["rating"] - book2["rating"]) / 5.0
    similarity = (genres_similarity + author_similarity +
                 title_similarity + rating_similarity) / 4.0
    return similarity
```

Fonte: Elaborado pelo autor (2023)

Na Figura 3.7 é calculada a similaridade entre dois usuários utilizando do *TF-IDF* (*Term Frequency-Inverse Document Frequency*) e tomando por base os gêneros dos livros dos usuários. O *TF-IDF* é comumente usado em processamento de linguagem natural para determinar a importância relativa de cada palavra em um documento ou corpus de documentos.

Figura 3.7 – Calculando a similaridade entre os usuários

```
def tfidf_similarity(genres1, genres2):
    genre_corpus = [book["genero"] for book in genres1] + [book["genero"] for book in genres2]
    vectorizer = TfidfVectorizer()
    tfidf_matrix = vectorizer.fit_transform(genre_corpus)
    similarity_matrix = tfidf_matrix[:len(genres1)].dot(tfidf_matrix[len(genres1):].T)
    similarity = np.mean(similarity_matrix.diagonal())

    return similarity
```

Fonte: Elaborado pelo autor (2023)

3.4.4 Itens Recomendados

Logo após ter sido calculada a similaridade entre os gêneros do usuário com os demais gêneros, e a similaridade entre o usuário com outro usuário, os livros serão adicionados às respectivas listas que serão retornadas.

A Figura 3.8 mostra o loop responsável por percorrer cada usuário do banco e

então fazer à chamada às funções, para que fosse evitado duplicatas é verificado se em um determinado momento o usuário verificado não é ele mesmo.

Figura 3.8 – Percorrendo os usuários do BD

```

for entry in database:
    if entry["email"] != user_email:
        similarity_genre = genre_similarity(
            [book["genero"] for book in user_books],
            [book["genero"] for book in entry["books"]],
            user_density,
            mean_density
        )
        similarity_user = tfidf_similarity(user_books, entry["books"])
        similarities_genre.append((entry["email"], similarity_genre))
        similarities_user.append((entry["email"], similarity_user))

```

Fonte: Elaborado pelo autor (2023)

A Figura 3.9 mostra o código que percorre a lista de usuários ordenados por similaridade em relação aos gêneros dos livros, e se o nível de similaridade for superior à 0.1 (para garantir o peso da similaridade em valores entre 0 e 1 e assegurar-se da não esparsidade em valores por exemplo: "0.01"), ele é adicionado a lista.

Figura 3.9 – Gêneros Similares

```

for entry in similarities_genre:
    user_entry = next(item for item in database if item["email"] == entry[0])
    similar_books = []
    for book in user_entry["books"]:
        for user_book in user_books:
            content_sim = content_similarity(book, user_book, user_density, mean_density)
            if content_sim > 0.1:
                similar_books.append(book)
                break
    recommended_books_genre.extend(similar_books)

```

Fonte: Elaborado pelo autor (2023)

Já na Figura 3.10 é mostrado o código que percorre a lista ordenada de usuários que são similares ao usuário atual, calculada utilizando a abordagem do método do TF-IDF para isso. A métrica de 0.05 é um valor arbitrário que devido ao corpus dos dados torna o sistema mais inclusivo. Valores maiores tornarão o sistema mais restritivo na recomendação de livros com base na similaridade, enquanto valores menores permitirão que mais usuários e seus livros sejam recomendados.

3.4.5 Métricas

Tomando um usuário da base de dados como exemplo para envio da *Request*, após a execução das funções são obtidos valores de dois tipos de métricas que são importantes

Figura 3.10 – Usuários Similares

```

for entry in similarities_user:
    if entry[1] >= 0.05:
        user_entry = next(item for item in database if item["email"] == entry[0])
        similar_books = user_entry["books"]
        recommended_books_user.extend(similar_books)

```

Fonte: Elaborado pelo autor (2023)

para leitura da aplicação - o nível de similaridade do conteúdo e o nível de similaridade do usuário, obtido nas funções *content similarity* e *tfidf similarity* respectivamente, conforme mostrado nas Figuras 3.11 e 3.12.

Figura 3.11 – Nível de similaridade do conteúdo

```

PROBLEMAS  SAÍDA  CONSOLE DE DEPURACÃO  TERMINAL

[3] -----
[3] CONTENT SIMILARITY LEVEL
[3] 0.3613483225498993
[3] -----
[3] CONTENT SIMILARITY LEVEL
[3] 0.4249001794597505
[3] -----
[3] CONTENT SIMILARITY LEVEL
[3] 0.400304365287082
[3] -----
[3] CONTENT SIMILARITY LEVEL
[3] 0.34670173600188875
[3] -----
[3] CONTENT SIMILARITY LEVEL
[3] 0.34670173600188875
[3] -----

```

Fonte: Elaborado pelo autor (2023)

Figura 3.12 – Nível de similaridade do usuário

```

PROBLEMAS  SAÍDA  CONSOLE DE DEPURACÃO  TERMINAL

○ [3] TFIDF SIMILARITY LEVEL
[3] 0.05924047308066366
[3] -----

```

Fonte: Elaborado pelo autor (2023)

3.5 RESULTADOS OBTIDOS

Nessa seção será exposto os resultados obtidos a partir de capturas de tela das interfaces da aplicação

3.5.1 Landing Page

Na Figura 3.13 é mostrada a tela inicial da aplicação onde o usuário pode fazer seu Login caso já tenha cadastro no sistema, ou então se Cadastrar.

Figura 3.13 – Landing Page

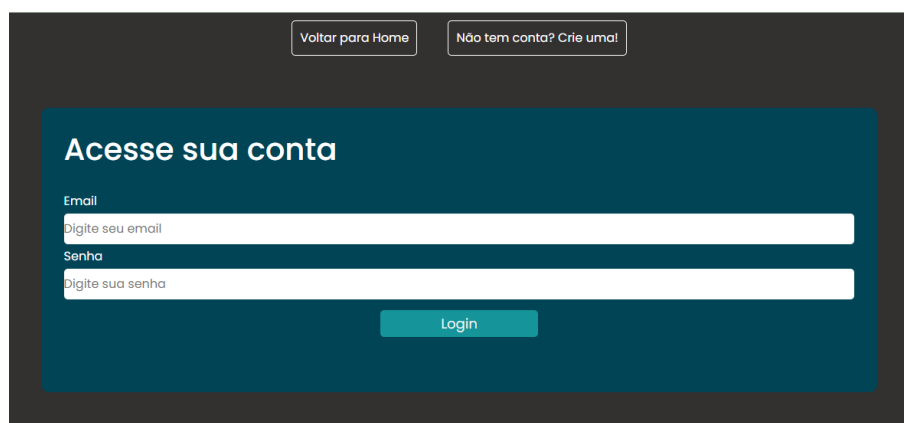


Fonte: Elaborado pelo autor (2023)

3.5.2 Autenticação

Na Figura 3.14 é mostrada a tela de Login para que o usuário acesse o sistema com suas credenciais.

Figura 3.14 – Autenticação

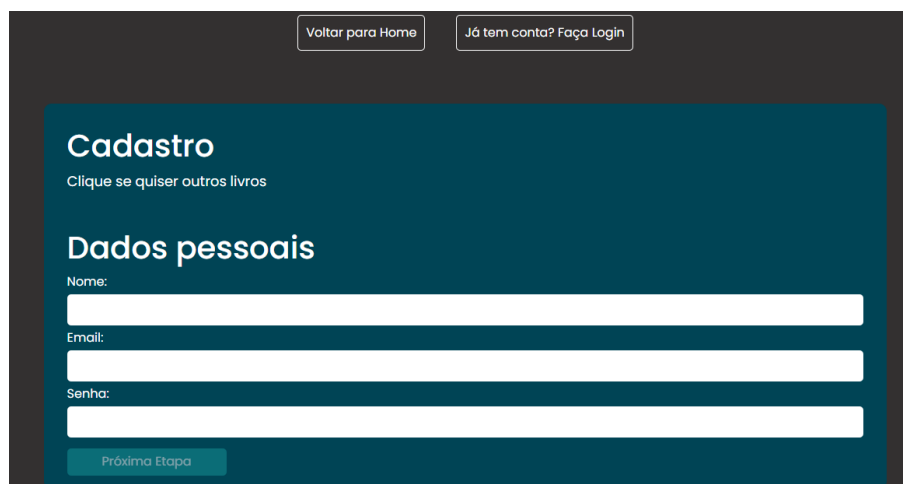


Fonte: Elaborado pelo autor (2023)

3.5.3 Cadastro

Nas figuras 3.15 e 3.16 é mostrado as telas das duas etapas do Cadastro, onde nesse momento o usuário precisa selecionar ao menos 3 livros para adicionar à sua coleção, coletando assim possíveis preferências iniciais para evitar o *cold start*.

Figura 3.15 – Cadastro



Voltar para Home Já tem conta? Faça Login

Cadastro

Clique se quiser outros livros

Dados pessoais

Nome:

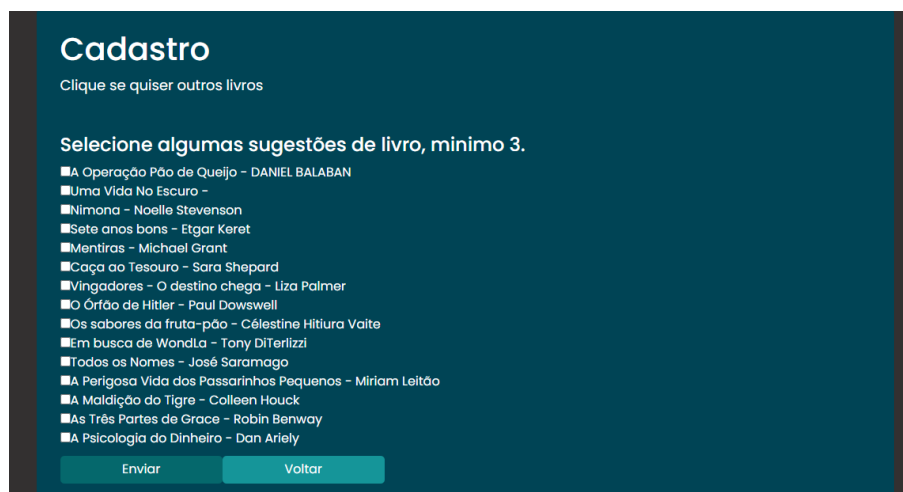
Email:

Senha:

Próxima Etapa

Fonte: Elaborado pelo autor (2023)

Figura 3.16 – Cadastro



Cadastro

Clique se quiser outros livros

Selecione algumas sugestões de livro, mínimo 3.

- A Operação Pão de Queijo - DANIEL BALABAN
- Uma Vida No Escuro -
- Nimona - Noelle Stevenson
- Sete anos bons - Etgar Keret
- Mentiras - Michael Grant
- Caça ao Tesouro - Sara Shepard
- Vingadores - O destino chega - Liza Palmer
- O Órfão de Hitler - Paul Dowswell
- Os sabores da fruta-pão - Célestine Hitiura Vaite
- Em busca de WondLa - Tony DiTerlizzi
- Todos os Nomes - José Saramago
- A Perigosa Vida dos Passarinhos Pequenos - Miriam Leitão
- A Maldição do Tigre - Colleen Houck
- As Três Partes de Grace - Robin Benway
- A Psicologia do Dinheiro - Dan Ariely

Enviar Voltar

Fonte: Elaborado pelo autor (2023)

3.5.4 Administração

Na figura 3.17 é mostrado a tela de Administração após o usuário entrar com suas credenciais. Nessa tela temos o menu de *Favoritos* que o usuário pode salvar livros que futuramente venha adicionar à sua coleção e o menu de *Recomendações* que leva para a tela dos livros recomendados pelo endpoint da API.

Figura 3.17 – Administração



Fonte: Elaborado pelo autor (2023)

Ainda na tela de Administração do usuário conforme mostra a Figura 3.18, consta com uma vitrine que sugere opções do livros com maiores avaliações da base de dados, estes livros são mostrados somente com base em mais avaliados, a recomendação de livros se dá de fato na tela de *Recomendações*

Figura 3.18 – Vitrine de Livros



Fonte: Elaborado pelo autor (2023)

3.5.5 Recomendações

Na figura 3.19 é mostrado a tela da página em que as recomendações geradas são renderizadas no *Frontend*, constando com a descrição do livro, a avaliação que ele tem, a listagem dos gêneros, link para buscar sobre o Livro no Google, e o botão que o adiciona à coleção pessoal do usuário.

Figura 3.19 – Recomendações

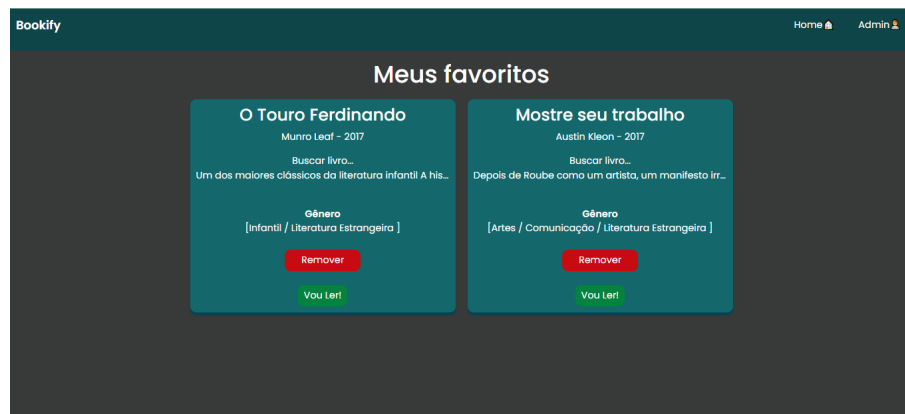


Fonte: Elaborado pelo autor (2023)

3.5.6 Favoritos

Ainda há também a página de Favoritos conforme mostra a Figura 3.20, onde estão presentes os livros que o usuário adiciona para posteriormente avaliar se adicionará à sua coleção ou não. Essa parte do *Frontend* utiliza-se da *Context API* do *React* para fazer com que os livros favoritados sejam acessíveis por toda aplicação.

Figura 3.20 – Favoritos



Fonte: Elaborado pelo autor (2023)

4 CONSIDERAÇÕES FINAIS

Como vimos no presente trabalho estamos cercados pela tecnologia, que cresce exponencialmente cada dia mais e torna mais possível ações que outrora ainda não eram pensadas. E a nossa experiência com ela permeia contextos dos mais diversos e dentro destes contextos podemos compreender a dimensão da importância no que diz respeito aos sistemas de recomendações, que em toda nossa troca de experiência com o meio e interação social se faz presente, seja nos e-commerces que acessamos diariamente, seja nas redes sociais, ou até mesmo nos aplicativos.

Foi visto ainda, que os sistemas de recomendações, definidos como mecanismo de personalização de conteúdo, auxiliam os usuários na hora da tomada de suas decisões, tendo em vista que o permite ter uma experiência particular às suas preferências, contribuindo assim para o processo de decisão do mesmo e mostrando-lhe sugestões assertivas em conformidade com seu perfil.

Inicialmente, é importante salientar que o desenvolvimento e implementação do sistema enfrentaram algumas restrições temporais significativas, aliado às responsabilidades acadêmicas e profissionais dos envolvidos, limitou a alocação de tempo necessário para a condução de um estudo de caso com grupo reduzido de maneira abrangente. Embora tenha sido esta uma limitação encontrada, ressalta-se que a mesma não invalida os resultados e as contribuições técnicas deste trabalho, que se concentrou na concepção e implementação do sistema.

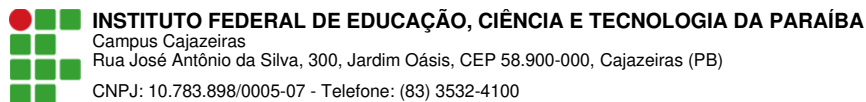
Contudo, diante do tema abordado, é esperado que os resultados possam se demonstrar contundentes no que tange a acurácia das recomendações proporcionando ao usuário final uma experiência personalizada pautada na confiabilidade de outrem e em conformidade com suas preferências particulares. Espera-se ainda que este estudo reflita eficácia na geração das sugestões aos usuários finais, tornando assim a interação humano-computador mais rápida, acertada e menos dispendiosa de tempo.

Ainda é esperado que o presente estudo possa aprimorar a experiência na prática de experimentos de técnicas de recomendações e estratégias de geração de sugestões, e implementação de sistemas de recomendações, bem como agregar conhecimentos acadêmicos e metodológicos de pesquisa, a fim de estudo a respeito do tema que foi abordado.

REFERÊNCIAS

- ADOMAVICIUS, G.; TUZHILIN, A. **Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions**. In IEEE Transactions On Knowledge and Data Engineering, vol. 17, no. 6, 2005, p. 734-749.10
- ZYLBERSZTEJN, Vivian Suslik. **Relação entre identificação consumidor-empresa, experiência com o serviço e o comportamento de recomendação do consumidor**. 2012.
- CAZELLA, Sílvio; NUNES, Maria Augusta; REATEGUI, Eliseo. **A Ciência da Opinião: Estado da arte em Sistemas de Recomendação**. CONGRESSO DA SOCIEDADE DE COMPUTAÇÃO, 30, 20 a 23 de julho de 2010. Belo Horizonte. Anais... Belo Horizonte: SBC, 2010.
- MENTI, Julia Fialho. **Fatores motivadores da participação em social commerce: um estudo com usuários brasileiros**. 2016.
- ALVAREZ, Edgar Bisset et al. **Os sistemas de recomendação, arquitetura da informação e a encontrabilidade da informação**. Transformação, v. 28, p. 275-286, 2016.
- DE MELO CRUZ, Wander Luis. **Crescimento do e-commerce no Brasil: desenvolvimento, serviços logísticos e o impulso da pandemia de Covid-19**. GeoTextos, 2021.
- DE CASTRO, ARMANDO ANTONIO MONTEIRO; DO PRADO, PEDRO PAULO LEITE. **Algoritmos para reconhecimento de padrões**. Revista Ciências Exatas, v. 8, n. 2002, 2002.
- KARLGREN, Jussi. **The Systems Development and Artificial Intelligence Laboratory**. Stockholm University, Stockholm, 1990. <Disponível em: <<https://jussikarlgren.files.wordpress.com/1990/09/algebrawp.pdf>>>
- KARLGREN, Jussi. 1994. **Newsgroup Clustering Based On User Behavior - A Recommendation Algebra**. Technical Report. European Research Consortium for Informatics and Mathematics at SICS
- GOLDBERG, David et al. **Using collaborative filtering to weave an information tapestry**. Communications of the ACM, v. 35, n. 12, p. 61-70, 1992.
- HUG, Nicolas. **Surprise: A Python library for recommender systems**. Journal of Open Source Software, v. 5, n. 52, p. 2174, 2020.
- Rajaraman, A., Ullman, J. (2011). **Data Mining**. In Mining of Massive Datasets (pp. 1-17). Cambridge: Cambridge University Press. doi:10.1017/CBO9781139058452.002
- HU, Jinku. **Calcule a distância euclidiana em Python**. Delfstack, 2019. <Disponível em: <https://www.delftstack.com/pt/howto/numpy/calculate-euclidean-distance/>>. Acesso em: 17 de mai. de 2022.
- MAKLIN, Cory. **TF IDF | TFIDF Python Example**. Towards Science, 2019. <Disponível em: <https://towardsdatascience.com/natural-language-processing-feature-engineering-using-tf-idf-e8b9d00e7e76>>. Acesso em: 16 de mai. de 2022.

Baeza-Yates, R. (2011). *Modern Information Retrieval: The Concepts and Technology Behind Search* (2nd ed.). New York: Addison Wesley. OCLC 683591535.



Documento Digitalizado Ostensivo (Público)

Trabalho de Conclusão de Curso

Assunto: Trabalho de Conclusão de Curso
Assinado por: Daniel Nogueira
Tipo do Documento: Anexo
Situação: Finalizado
Nível de Acesso: Ostensivo (Público)
Tipo do Conferência: Cópia Simples

Documento assinado eletronicamente por:

- Daniel Silva Nogueira, ALUNO (201722010014) DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - CAJAZEIRAS, em 28/09/2023 08:35:18.

Este documento foi armazenado no SUAP em 28/09/2023. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 956232
Código de Autenticação: 1296f39c7e

