

**IFPI – Instituto Federal de Educação, Ciência e Tecnologia do Piauí**  
**Campus Teresina Central**  
**Departamento de Informação, Ambiente, Saúde e Produção Alimentícia**  
**Curso: Tecnologia em Análise e Desenvolvimento de Sistemas**  
**Disciplina: Projeto Integrador I**

## **Projeto AcervBooks**

**Juliana Lima de Sousa**  
**Kastier Francisco Leite Junior**  
**Micherlane Rodrigues Machado da Silva**

**Responsáveis: Prof. M<sup>e</sup> Ely Miranda**

**3 de julho de 2023**

## **Resumo**

O AcervBooks consiste em um indexador de livros, desenvolvido para web, que visa agilizar o processo de busca de livros no acervo da biblioteca do Instituto Federal do Piauí – IFPI, que atualmente enfrenta problemas significativos, como a lentidão no processo de localização de títulos diversos, podendo muitas vezes gerar frustração e desestímulo ao usuário desse serviço. A proposta é trazer essa funcionalidade com um tempo de resposta menor do que o atual, tornando a busca agradável e intuitiva. Traz também alterações nos dados dos títulos, com uma descrição mais ampla e completa das informações contidas nos mesmos, auxiliando, assim, os usuários na escolha do livro que se adequa à sua necessidade.

## **1 Justificativa**

Na era que hoje a sociedade se encontra, a busca de informações relevantes é cada vez mais importante. Não é à toa que uma das grandes empresas de tecnologia do mundo seja um indexador de sites, Google, que tem como objetivo facilitar que o usuário encontre mais facilmente os sites de seu interesse. Essa ideia não é diferente para um sistema de biblioteca, pois esta tem como objetivo conectar leitores a livros de seus interesses. Entretanto, nem todas as bibliotecas atingem esse objetivo com êxito, como é o caso da biblioteca do Instituto Federal de Ciência e Tecnologia do Piauí(IFPI).

A biblioteca do Instituto Federal de Ciência e Tecnologia do Piauí (IFPI) enfrenta diversos desafios em relação à busca de livros em seu acervo. Um dos principais problemas é a demora significativa no processo de localização, que muitas vezes leva minutos para ser concluído. Essa lentidão pode causar frustração e desperdício de tempo para os usuários, chegando até mesmo a desencorajá-los na busca pelos livros desejados.

Além disso, o sistema atual da biblioteca carece de um eficiente sistema de pesquisa que ofereça recursos como autocomplemento nas informações digitadas. A falta desse recurso prejudica a agilidade e a precisão da pesquisa, tornando a experiência de busca menos satisfatória para os usuários. A possibilidade de receber sugestões relevantes enquanto digitam facilita a descoberta de obras relacionadas aos seus interesses e necessidades.

Outro ponto importante é a limitação das informações disponíveis sobre os livros no acervo da biblioteca do IFPI. Atualmente, são fornecidos apenas dados básicos e poucos detalhes relevantes sobre as obras, o que dificulta a seleção adequada de livros de acordo com os interesses individuais dos usuários. Essa falta de informações abrangentes pode resultar em escolhas menos assertivas e menos alinhadas às necessidades específicas dos leitores.

Diante desses desafios, é evidente a necessidade de aprimorar o sistema de busca da biblioteca do IFPI. O projeto AcervBooks surge como uma solução viável para enfrentar

essas questões. Por meio de um software de busca de livros aprimorado, o AcervBooks visa reduzir significativamente a demora no processo de localização, proporcionando resultados mais rápidos e eficientes.

Além disso, o AcervBooks oferecerá recursos avançados de pesquisa, incluindo auto-complemento, que tornarão a busca mais intuitiva e precisa para os usuários. A possibilidade de receber sugestões relevantes enquanto digitam facilitará a descoberta de novos títulos e ampliará o acesso a obras relacionadas aos interesses individuais de cada usuário.

Por fim, o AcervBooks fornecerá informações completas e relevantes sobre as obras, incluindo sinopses, detalhes do autor e edições, entre outros dados essenciais. Essa riqueza de informações auxiliará os usuários na seleção de livros adequados às suas necessidades e interesses específicos, melhorando consideravelmente a experiência de busca e escolha na biblioteca do IFPI.

Em suma, aprimorar o sistema de busca por meio do projeto AcervBooks é fundamental para atender às demandas e superar os desafios enfrentados pela biblioteca do IFPI. Essa iniciativa contribuirá para melhorar a experiência dos usuários, aumentar a satisfação e a utilização do acervo, além de incentivar a descoberta de novos livros e promover o hábito da leitura entre os membros da comunidade acadêmica.

## **2 Fundamentação Teórica**

Para uma melhor compreensão dos motivos para a criação de um sistema buscador de livros para biblioteca, é necessário analisar dois conceitos importantes que se baseia esse projeto: sistema de busca e biblioteca 2.0 e também a técnica de full-text search.

### **2.1 Sistema de busca e biblioteca 2.0**

Para compreender o que significado de biblioteca 2.0, é necessário se entender o que é Web 2.0. O termo Web 2.0 foi um termo utilizado por Tim O'Reilly para descrever modelos de negócios que se baseavam em uma rede como plataforma, ou seja, que envolviam todos os dispositivos conectados. As aplicações voltadas para esse estilo de negócio são aquelas que entregam softwares que podem ser continuamente serem atualizados, com melhorias que ocorrem à medida que os usuários interagem com o sistema, seja consumindo ou gerando novos conteúdos de diversas fontes e aproveitando a inteligência coletiva.[1]

A biblioteca 2.0 é uma extensão do conceito de Web 2.0 aplicados às bibliotecas. Essa extensão pode ser definida como uma aplicação interativa de colaboração e tecnologia de multimídias baseadas em web para serviços e coleções de bibliotecas na web. Segundo Jack M. Maness[2], a biblioteca 2.0 pode ser definida com quatro elementos essenciais: é

centrada no usuário, oferece uma experiência multimídia, é socialmente rica, e é comunitariamente inovadora. Quando o autor refere-se a "centrado no usuário" significa que tanto o consumo como a criação de conteúdo é dinâmica, onde não há uma distinção clara entre usuário e bibliotecário. E "socialmente rica" porque permite a comunicação entre os usuários e bibliotecários quer na forma de mensagens síncronas ou assíncronas. A outra característica representa que diferentes mídias podem estar presentes no sistema; como vídeos, imagens, dentre outras formas de informações digitais. E por fim, "comunitariamente inovadora" pois admite a possibilidade que o sistema de biblioteca possa mudar se os usuários também mudarem. Permitindo sempre a evolução contínua do sistema.

Outro conceito importante intrínseco à biblioteca 2.0 é o sistema de busca ou sistema de busca bibliográfica(SBB). Esse recurso responsável por facilitar a recuperação de informações do acervo das bibliotecas. O SBB é dividido em duas camadas: a de interface de descoberta e o sistema de descoberta[3]. O primeiro retrata a área em que o usuário poderá interagir com o sistema, isto é através de formulários intuitivos onde poderão fornecer informações necessárias para as buscas no acervo da biblioteca. É comum recursos como tolerância à diferentes pontuações e pesquisa de digitação antecipada (caixa de textos com sugestões de pesquisa). Quanto ao segundo, refere-se à técnica utilizada para retornar as informações salvas no acervo.

Uma técnica relevante, que foi aplicada nesse projeto, para o sistema de busca é o full-text search que consiste em procura de algum termo contido em textos extensos, retornando como resultado alguma ou todas as palavras da pesquisa. É uma busca que se vale de um valor aproximado ao termo digitado o qual utiliza a indexação. A indexação pode ser feita de duas maneiras: em lote ou incremental. O índice é semelhante ao um grande dicionário, onde um termo pode ser encontrado em várias concepções. Com esse tipo de associação, o tempo de resposta reduz consideravelmente. Embora consuma um certo espaço do disco rígido.

A pesquisa de texto completo tem como raiz a criação do índice em que cada campo de texto possui um conjunto de dados. Ou seja, de cada palavra terá removido sinais diacríticos, por exemplo, as acentuações serão removidas. Além disso, as palavras de ligação como preposições ou identificadores como artigos não serão consideradas. Ademais letras maiúsculas e minúsculas possuem o mesmo valor, não havendo uma diferenciação se o usuário escreveu "pESSoa" ou "pessoa", pois na fase final, todas as letras das palavras passarão a ser maiúsculas. Com esse processo, haverá uma otimização na pesquisa de texto. Apesar de que ela será muito mais eficiente se recursos adicionais foram usados em conjunto como sugestão de preenchimento automático, pesquisa de sinônimos e correções no texto (como sugestões de "você quis dizer..."). [4]

Com esta base teórica, foi possível a criação de uma solução tecnológica para um buscador que otimize o retorno de informações do acervo da biblioteca.

## 3 Objetivo Geral

O objetivo geral do projeto AcervBooks é a criação de um software de código-fonte aberto que facilite a busca de informações do acervo do Instituto Federal de Ciência e Tecnologia do Estado do Piauí(IFPI). O software atuará como um buscador de livros, tornando mais fácil para os usuários encontrarem informações sobre os livros disponíveis no acervo do IFPI.

### 3.1 Objetivos específicos

1. Diminuir o tempo de espera do carregamento das informações do banco de dados: isso significa otimizar o tempo de resposta das informações no banco de dados. Isto reduzirá o tempo de espera dos usuários e proporcionará uma experiência mais agradável para o usuário.
2. Permitir pesquisas mais amplas, no sentido de com menos informações emitir resultados que se aproximem da informação procurada. Isto é, o software é equipada para retornar informações relevantes do livro procurado mesmo que as informações passadas sejam limitadas. Isso facilitará a busca e ampliará as chances de encontrar os livros desejados.
3. Oferecer uma interface intuitiva que propicie uma experiência mais agradável para o usuário, ou seja, será projetado para ser fácil de compreender e de utilizar. Para que assim, a busca de informações no acervo seja mais acessível e conveniente.

Dessa forma, este projeto tem a ambição de promover uma experiência agradável na busca de informações sobre os livros disponibilizados no acervo do IFPI.

## 4 Metodologia da Execução do Projeto

A técnica utilizada pelos membros da equipe para execução do dado projeto segue os conceitos do modelo Agile(Ágil), que pode ser definido como um princípio de gestão de projetos. Dentro desse modelo, tem-se como foco a metodologia Scrum, tido como um método para a criação de ciclos por meio de sprints que duram entre uma e duas semanas. Em meio as definições mencionadas, a execução do projeto segue as seguintes etapas:

1. Fase de diagnósticos;
2. Concepção;

3. Levantamento e análise de requisitos;
4. Fase de desenvolvimento;
5. Fase de testes;
6. Entrega.

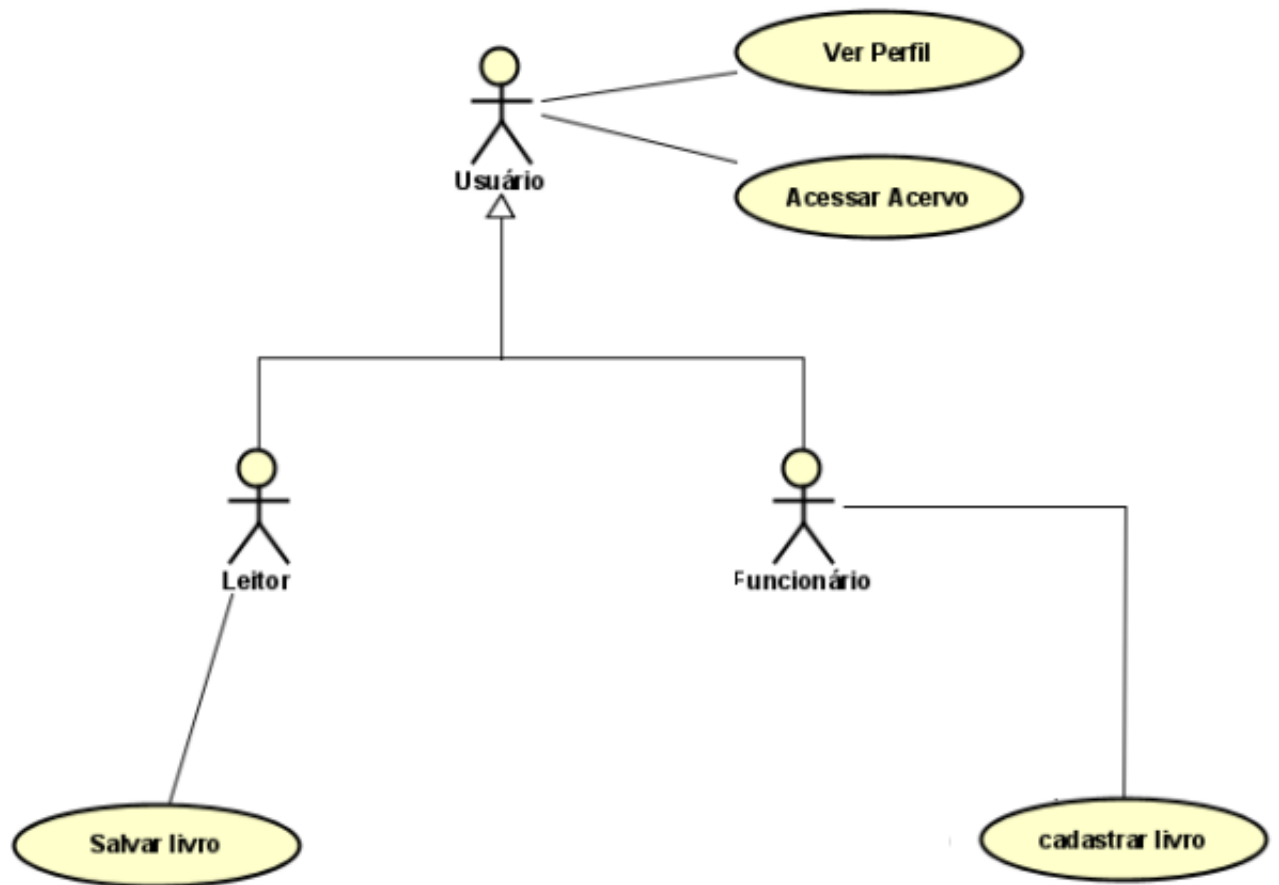
A ferramenta de gerenciamento de projeto escolhida para acompanhar a execução das etapas mencionadas acima foi o Trello, o qual está dividido em “a fazer”, “fazendo”, “teste” e “concluído”. Para o desenvolvimento da aplicação web, foi definido a utilização de React e Node.js como tecnologias, bem como a utilização de Typescript para a codificação da mesma. A plataforma de hospedagem de código escolhido foi o GitHub, sendo o repositório compartilhado entre os membros da equipe, possibilitando, assim, o trabalho conjunto entre os mesmos.

## **4.1 Requisitos**

O levantamento de requisitos foi feito com base nas observações feitas do sistema de busca de livros no acervo atual.

1. O sistema deverá permitir que o usuário possa buscar informações sobre os livros que deseja encontrar;
2. O sistema permitirá que um usuário autorizado possa cadastrar , editar e excluir livros no banco de dados;
3. O usuário poderá salvar um livro como favorito;
4. O sistema deverá se desenvolvido para a plataforma web;

## 4.2 Casos de Uso



1. Acessar Acervo

<b>Nome da Funcionalidade</b>	<b>Acessar Acervo</b>
<b>Resumo</b>	Esta funcionalidade descreve etapas necessárias para que o usuário possa acessar o acervo.
<b>Pré-condições</b>	O leitor deverá estar logado no sistema
<b>Pós-condições</b>	
<b>Ações do ator</b>	<b>Ações do sistema</b>
1. O leitor acessa a tela de acessar acervo.	
2. O leitor digita o termo e/ou aplica o filtro sobre o que deseja encontrar.	
3. O leitor seleciona a opção pesquisar.	
	4. O sistema faz uma busca na base de dados com os termos e/ou filtros correspondentes às informações do(s) livro(s) e os livros são encontrados.
	5. O sistema exibe na tela uma lista dos livros encontrados
<b>Cenário Alternativo I</b>	
1. O leitor não digitou nenhum termo e não aplicou nenhum filtro	
2. O leitor seleciona a opção pesquisar.	
	3. O sistema alerta que nenhum termo ou filtro foi aplicado.
<b>Cenário Alternativo II</b>	
1. O leitor digita o termo e/ou aplica o filtro do livro que ele quer encontrar	
2. O leitor seleciona a opção pesquisar.	
	3. O sistema faz uma busca na base de dados com o termo e/ou filtro aplicado com as informações do livro e este não é encontrado.
	4. O sistema exibe uma mensagem informando que o livro não foi encontrado.

## 2. Acessar Perfil



<b>Nome da Funcionalidade</b>	<b>Acessar Perfil</b>
<b>Resumo</b>	Esta funcionalidade descreve etapas necessárias para que o leitor acesse seu perfil
<b>Pré-condições</b>	O leitor deverá estar logado no sistema
<b>Pós-condições</b>	
<b>Ações do ator</b>	<b>Ações do sistema</b>
1. O leitor acessa a tela de perfil.	
	2. O sistema mostra as informações do leitor ( <div>           Nome,            data de nascimento,            endereço,            gênero,            matrícula,            email,            telefone e livros salvos )         </div>

### 3. Cadastrar Leitor

<b>Nome da Funcionalidade</b>	<b>Cadastrar Leitor</b>
<b>Resumo</b>	Esta funcionalidade descreve etapas necessárias para cadastrar um livro.
<b>Pré-condições</b>	
<b>Pós-condições</b>	
<b>Ações do ator</b>	<b>Ações do sistema</b>
1. O leitor acessa a tela de login	
2. Escolha a opção "Primeiro Acesso?".	
	3. O sistema mostra a tela de cadastro
4. O leitor informa os seguintes dados: nome completo, matrícula, email, senha, telefone.	
5. O leitor aperta o botão “cadastrar-se”	
	6. O sistema notifica que o usuário foi cadastrado com sucesso.
	7. O sistema redireciona o usuário para a tela de login
<b>Cenário Alternativo I</b>	Usuário não pode ser cadastrado
	1. O sistema notifica que o usuário não pode se cadastrar.

#### 4. Cadastrar Livro

<b>Nome da Funcionalidade</b>	<b>Cadastrar Livro</b>
<b>Resumo</b>	Esta funcionalidade descreve etapas necessárias para que o funcionário cadastre um livro no sistema
<b>Pré-condições</b>	O funcionário deverá estar logado no sistema e ter utilizado a funcionalidade acessar acervo.
<b>Pós-condições</b>	
<b>Ações do ator</b>	<b>Ações do sistema</b>
1. Na tela de acesso ao acervo, o funcionário aperta o botão “adicionar novo livro”	
	2.O sistema abre uma tela modal de cadastro de livros.
3. O funcionário informa as informações do livro (imagem, sinopse, gênero, ano, volume, título, linguagem, editora e autor).	
4. Ele aperta o botão salvar	
	5. O sistema salva o livro no banco de dados.
	6. O sistema notifica que o livro foi salvo com sucesso
	7. O sistema retorna para a tela de acesso ao acervo.
<b>Cenário Alternativo I</b>	O livro não pode ser cadastrado
	1. O sistema informa que o livro não pode ser salvo e retorna para a tela de acesso ao acervo.

### 4.3 Priorização das Atividades

As atividade foram classificadas dado ao nível de complexidade dentro do sistema. Na tabela a seguir há os campos nome da funcionalidade em que apresenta a ideia fundamental da funcionalidade, tipo em que classifica se ela é funcional ou não funcional e complexidade exigida para se feita.

Nome da Funcionalidade	Tipo	Complexidade
Acessar Acervo	funcional	alta
Cadastrar Leitor	funcional	média
Cadastrar Livro	funcional	média
Favoritar Livro	funcional	baixa
Ver perfil	funcional	baixa

Tabela 1: Tabela de Priorização de atividades

## 5 Implementação

### 5.1 Tecnologias Utilizadas

O projeto AcervBooks é dividido em duas partes Back-End e Front-End. O Back-End é responsável por prover uma API REST, em que está as conexões com o banco de dados e regras do negócio. Quanto a Front-End, ele é responsável por ter uma apresentação de dos dados para o usuário. Cadas uma dessas partes possui um conjunto de tecnologias envolvidas que serão apresentadas a seguir:

- Back-end:

- Node.js: O Node.js foi usado como ambiente de execução do servidor.
- Express.js: o framework Express.js foi usado para a criação da estrutura do servidor e a utilização da API REST.
- Postgresql: o banco de dados escolhido foi o Postgresql para armazenar os dados da aplicação.
- Prisma: foi utilizado o ORM (mapeamento objeto-relacional) para interagir com o banco de dados. O Prisma permite mapear as entidades e relações do banco de dados para objetos e consultas, tornando mais fácil e eficiente o trabalho como o banco de dados.

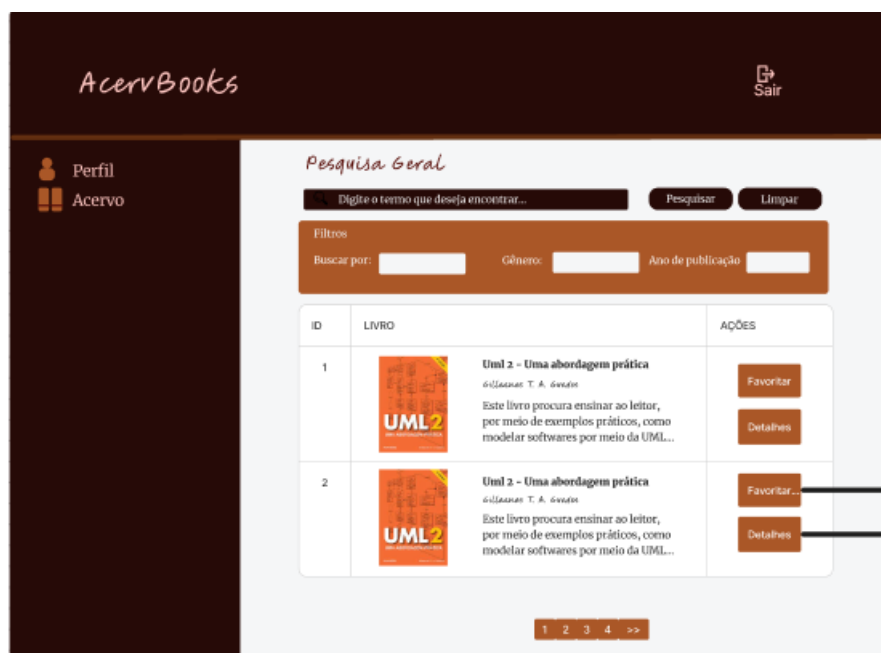
- Front-end:

- React: foi utilizado o React como biblioteca JavaScript de código aberto para construir a interface do usuário do aplicativo de forma modular e reutilizável.

- HTML5: foi utilizado HTML5 para estruturar e marcar o conteúdo do site, trabalhando em conjunto com os componentes do React.
- CSS3: Foi aplicado CSS3 para estilizar e posicionar os elementos da página, garantindo uma aparência visual atraente e responsiva.
- . Autenticação e autorização:
  - JSON Web Tokens(JWT): foi implementado a autenticação baseada em tokens usando JWT para autenticar o usuário e proteger rotas específicas.
  - bcrypt: foi usando também o bcrypt para criptografar e verificar as senhas dos usuários.
- . Controle de versão e colaboração:
  - Git: o Git foi usado para o controle de versão do código e colaboração com outros membros da equipe de desenvolvimento.
  - Github: o Github foi usado como plataforma de hospedagem para o repositório Git, permitido o compartilhamento e colaboração eficientes de código.

## 5.2 Protótipo

Nesta seção será apresentada de forma ilustrativa a solução desenvolvida neste trabalho.

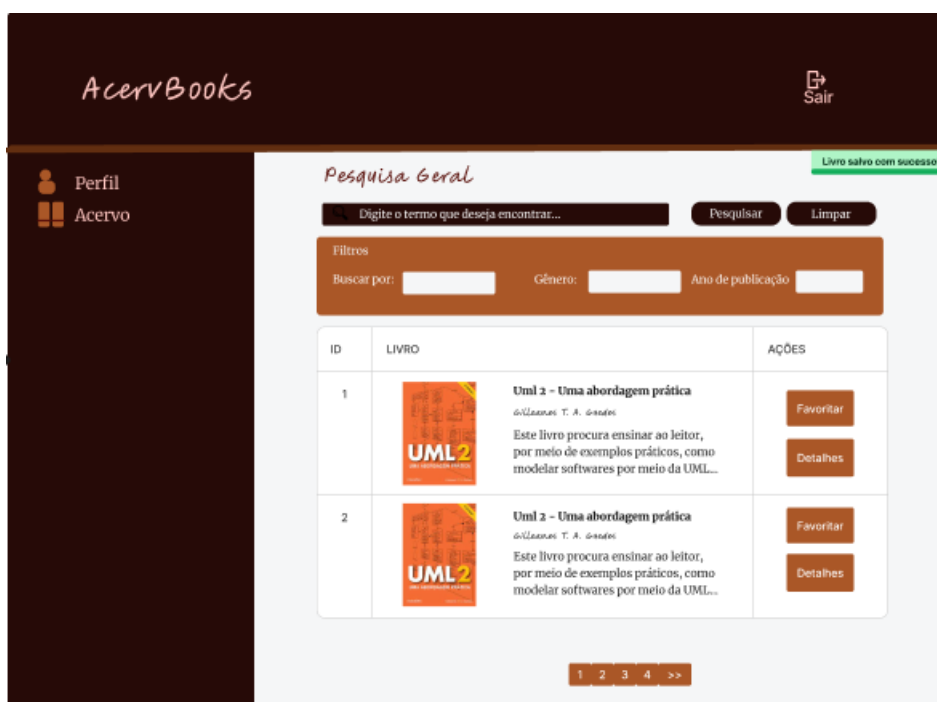


Esta é a página Home da aplicação. Ela é acessível após o usuário estar devidamente autenticado no sistema. A composição das páginas seguem padrão desta página. A estrutura

é composta por um cabeçalho contendo o nome da aplicação e uma opção de desconexão como sistema. Na parte lateral, é possível encontrar a parte da navegação sistema o qual contém duas rotas: a de perfil onde o usuário pode ver seus dados e livros favoritos e a de acervo onde pode pesquisar os livros, adicionar como favorito ou ver os detalhes do mesmo.

Na tela Home é apresentada uma barra de pesquisa onde o usuário poderá digitar um termo em que queira encontrar em livro. É possível que ele adicione filtros, isto é, que preencha os campos destinado para isto.

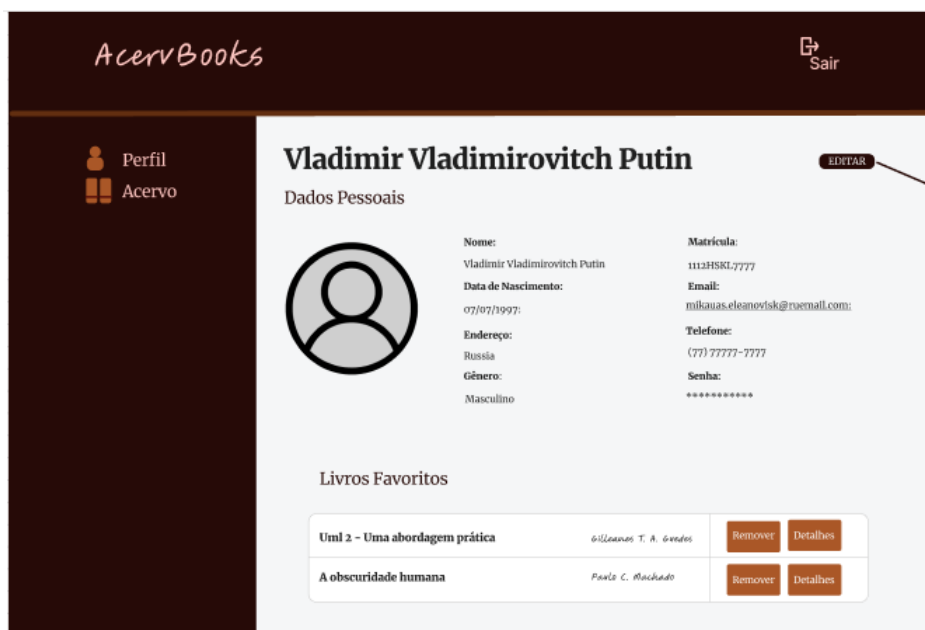
Uma lista de livros é apresentada. Esta lista possui as seguintes características: apresenta a capa do livro para facilitar o reconhecimento do mesmo, o título, o autor, e uma breve descrição. É oferecido também as opções de favoritar o livro e ver os detalhes.



Quando o usuário seleciona a opção "favoritar" o sistema apresenta uma notificação informando que o livro foi salvo nos favoritos.



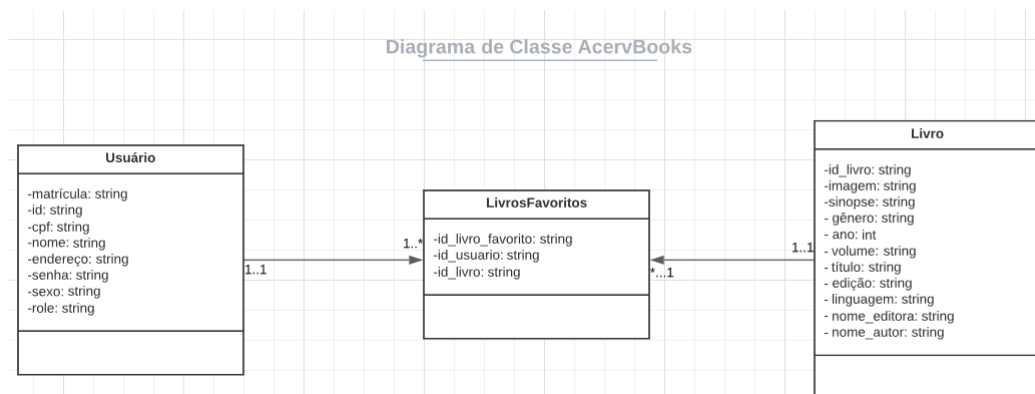
Ao selecionar a opção de "detalhes" aparece para o usuário uma tela que funciona como modal, e nesta é apresentada todas as informações dos livros disponíveis.



Por fim, se tem a tela de perfil do usuário. Nela é possível encontrar as informações do usuário, bem como uma foto, se o usuário preferir utilizá-la. Há também a opção de edição dos dados, o qual o usuário poderá atualizar os dados, se necessário.

Além disso, é apresentada uma lista de livros favoritos. A lista apresenta as seguintes características: o nome do livro, o autor e as opções de remover e detalhes. A primeira opção exclui o livro dos favoritos e a segunda apresenta os mesmos detalhes do livro.

## 5.3 Modelagem dos dados



As principais classe são Usuário e Livro. A primeira representa o usuário cadastrado no sistema cujas as informações são importantes para controle. A segunda mostra todas as informações dos livros, estas informações são relevantes pois facilitam na busca de livros mais próximos das informações passadas pelo usuário. Por fim, há um classe de relacionamento entre as classe usuário e livro, esta classe é responsável por guardar as informações dos livros como favoritos, relacionando o usuário e o livro.

## 6 Acompanhamento e Avaliação do Projeto Durante a Execução

Para o acompanhamento e avaliação do projeto ao longo de seu desenvolvimento, a equipe utilizará o Trello como ferramenta de gerenciamento, e organização, de tasks e sprints. A cada nova etapa definida nos quadros e executada pela equipe, também se disponibilizará no repositório remoto todas as pastas e documentos resultantes de cada procedimento.

Além da ferramenta de gerenciamento, faz-se necessário também a ocorrência de reuniões e feedbacks sempre que houver necessidade ao longo de todo processo de execução do projeto, visando o alinhamento contínuo da equipe até a entrega do produto.

## 7 Resultados Esperados e Disseminação dos Resultados

### 7.1 Resultados Esperados

O estudo realizado teve como objetivo a criação de um software que facilitasse a conexão entre os estudantes do Instituto Federal de Educação, Ciência e Tecnologia do Piauí

(IFPI) e os livros disponibilizados na biblioteca, ou seja, um software que atuasse como um buscador apresentando livros que se enquadrassem aos dados informados pelo usuário, exibindo o livro mais próximo da descrição feita pelo usuário cuja interface se mostrasse intuitiva e simples de ser utilizada.

## **7.2 Disseminação dos resultados**

Os resultados desse estudo foram divulgados através de um artigo científico que detalha a metodologia utilizada na criação do software. O artigo foi disponibilizado em um servidor remoto que estará disponível para a comunidade em geral. Além disso, foram realizadas apresentações-testes para os estudantes da turma de Análise e Desenvolvimento de Sistemas, juntamente com o professor-orientador da disciplina de Projeto Integrador, apresentando o software em funcionamento em tempo real para que todos pudessem avaliar a eficiência ou o fracasso da solução tecnológica apresentada.

## **8 Trabalhos Futuros**

Como trabalho futuro, pretende-se melhorar o algoritmo de busca de livros para que seja mais eficiente. Ou seja, permitir com menos informações, respostas mais aproximadas às informações fornecidas pelo usuário. Para esse aperfeiçoamento do algoritmo de correspondência de palavras-chave, em um futuro, será feita a integração com técnicas avançadas de processamento de linguagem natural para melhorar a precisão da pesquisa; além do uso de técnicas de aprendizado de máquina, para, por exemplo, classificar os resultados com base na relevância de consultas já feitas pelos usuários.

Por fim, expandir o sistema buscador para outras plataformas como o Android, primeiramente, e depois para IOS. Para que isto aproxime mais o usuário do sistema, permitindo outras opções fora o uso exclusivo na plataforma web.

## **9 Referências Bibliográficas**

1. Tim O'Reilly. "Web 2.0: Compact Definition?".

Disponível em: <http://radar.oreilly.com/archives/2005/10/web-20-compact-definition.html>. Acesso em: 10 de maio de 2023.

2. MANESS, Jack M. "Teoria da biblioteca 2.0: Web 2.0 e suas implicações para as bibliotecas". Disponível em: <https://brapci.inf.br/index.php/res/v/92256> Acesso em: 10 maio de 2023



3. Wikipedia contributors. "Discovery system (bibliographic search)".

In: Wikipedia: The Free Encyclopedia. Retrieved September 15, 2022. Disponível em: [https://en.wikipedia.org/wiki/Discovery\\_system\\_\(bibliographic\\_search\)](https://en.wikipedia.org/wiki/Discovery_system_(bibliographic_search)).

4. MONGODB. Full-Text Search. Disponível em: <https://www.mongodb.com/basics/full-text-search>. Acesso em: 15 maio 2023.