**DOCUMENTO DE REQUISITOS**

**BIBLIOTECA LUMINARES**

**Aluno:** Victor Gabriel Santos Bento

**Data:** 28/05/2024

**Sumário**

[1. Introdução 2](#_heading=h.gjdgxs)

[1.1. Requisitos 2](#_heading=h.30j0zll)

[1.1.1. O que são requisitos funcionais? 3](#_heading=h.1fob9te)

[1.1.2. O que são requisitos não funcionais? 3](#_heading=h.3znysh7)

[1.2. Descrição do Usuário 3](#_heading=h.2et92p0)

[1.2.1. Atores 4](#_heading=h.tyjcwt)

[2. Requisitos Funcionais 4](#_heading=h.3dy6vkm)

[2.1. Sistema de Login (RF01) 4](#_heading=h.1t3h5sf)

[2.2. Gerenciamento de Livros para Administradores (RF02) 4](#_heading=h.4d34og8)

[2.3. Gerenciamento de Categorias para Administradores (RF03) 5](#_heading=h.2s8eyo1)

[3. Requisitos Não-Funcionais 5](#_heading=h.17dp8vu)

[3.1. Usabilidade (RNF01) 5](#_heading=h.3rdcrjn)

[3.1.1. Responsividade 6](#_heading=h.26in1rg)

[3.1.2. Validação de Campos 6](#_heading=h.lnxbz9)

[3.1.3. Mensagens de Aviso 6](#_heading=h.35nkun2)

[3.2. Banco de Dados MySQL (RNF02) 6](#_heading=h.1ksv4uv)

[3.3. Laravel (RNF03) 7](#_heading=h.44sinio)

[3.4. Frameworks (RNF04) 8](#_heading=h.2jxsxqh)

[3.4.1. Tailwind 8](#_heading=h.z337ya)

[3.5. Requisítos Mínimos Para o Uso (RNF05) 9](#_heading=h.3j2qqm3)

[3.5.1. Tailwind 9](#_heading=h.1y810tw)

[3.5.2. Laravel 9](#_heading=h.4i7ojhp)

[3.5.2.1. Laravel 11.0x 10](#_heading=h.2xcytpi)

[4. Referências Bibliográficas 10](#_heading=h.7yracoy28sd3)

# 1. Introdução

A Biblioteca Luminares é um projeto que funciona como um Catálogo de Livros. O administrador pode criar, observar, atualizar ou deletar livros e categorias. É de desejo que o projeto também tenha diferentes atividades para o usuário comum e o artista. O usuário pode observar os livro e categoria presentes no catálogo. Esta aplicação pode ser utilizada por Bibliotecas.

## 1.1. Requisitos

As necessidades e expectativas dos usuários e outras partes em relação ao sistema em desenvolvimento são denominadas requisitos de software. Estes especificam as funcionalidades, comportamentos e limitações do sistema. Os requisitos podem ser categorizados em funcionais e não funcionais, descrevendo as ações desejadas do software e suas operações em diversos contextos. Esses requisitos influenciam a experiência do usuário, a eficiência operacional e a capacidade de adaptação do software.

### 1.1.1. O que são requisitos funcionais?

As funcionalidades especificam as operações específicas que um sistema ou aplicativo precisa realizar. Elas representam as capacidades tangíveis e as tarefas que o software deve executar para satisfazer as exigências e demandas do usuário.

**Exemplos:**

* Processamento de transações
* Gerenciamento de usuários
* Busca e filtragem de dados
* Geração de relatórios

### 1.1.2. O que são requisitos não funcionais?

Os requisitos de desempenho, segurança, confiabilidade e usabilidade ditam os métodos pelos quais o software deve cumprir suas funções. Esses aspectos são essenciais para garantir a qualidade e eficácia do software, abordando áreas como velocidade, proteção, estabilidade e facilidade de uso. Ao contrário dos requisitos funcionais, esses requisitos não estão diretamente ligados às tarefas específicas do software, mas sim à sua operação e contexto de uso.

**Exemplos:**

* Desempenho
* Segurança
* Confiabilidade
* Usabilidade

## 1.2. Descrição do Usuário

No âmbito da tecnologia da informação, um usuário é definido como qualquer indivíduo que interaja com um sistema computacional, seja este um software, um aplicativo, um site ou um dispositivo eletrônico.

### 1.2.1. Atores

**Administrador/Funcionário:** O administrador ou funcionário têm permissão para editar a tabela de livros e suas categorias, podendo criar, excluir ou modificar o conteúdo dessas tabelas.

**Usuário Comum:** O usuário comum pode visualizar os livros criado. O usuário não pode adicionar livros.

# 2. Requisitos Funcionais

## 2.1. Sistema de Login (RF01)

O sistema deve permitir que o usuário crie sua conta e faça o login, caso o email e a senha sejam válidos.

**Entradas:**email e senha.

**Saídas:** Mensagens de sucesso ou erro.

**Pré-condições:** Nenhuma.

* O usuário deve ser redirecionado para a página inicial caso o login seja bem sucedido.
* O sistema deve exibir uma mensagem de erro caso o login ou a senha estejam incorretos.
* O sistema deve exibir uma mensagem caso os inputs estejam vazios.

## 2.2. Gerenciamento de Livros para Administradores (RF02)

O sistema deve permitir que o administrador crie novos livros no catálogo, atualize informações de livros existentes, delete livros do catálogo e visualize a lista completa de livros.

**Pré-condições:** O usuário deve ter permissões de administrador.

**Entradas:**  Informações dos livros (idlivro, nome, isbn,categoria, autor, descrição).

**Saídas**: Mensagem de sucesso ou erro.

* O sistema deve exibir uma mensagem caso o processo seja executado corretamente.
* O sistema deve exibir uma mensagem caso ocorra um erro.
* O sistema deve permitir que o administrador escolha a categoria do livro.

## 2.3. Gerenciamento de Categorias para Administradores (RF03)

O sistema deve permitir que o administrador crie novas categorias, atualize informações de categorias existentes, delete e visualize a lista completa de categorias.

**Pré-condições:** O usuário deve ter permissões de administrador.

**Entradas:**  Informações da categorias (idcategoria, nome).

**Saídas**: Mensagem de sucesso ou erro.

* O sistema deve exibir uma mensagem caso o processo seja executado corretamente.
* O sistema deve exibir uma mensagem caso ocorra um erro.

**Entradas:**  Informações da categoria (iddcategoria, nome, etc).

**Saídas**: Mensagem de sucesso ou erro.

* O sistema deve exibir uma mensagem caso o processo seja executado corretamente.
* O sistema deve exibir uma mensagem caso ocorra um erro.

# 3. Requisitos Não-Funcionais

## 3.1. Usabilidade (RNF01)

A usabilidade em aplicativos é essencial para o design de UX, visando garantir que os usuários possam interagir de maneira eficaz e satisfatória.

### 3.1.1. Responsividade

A capacidade do software de se adaptar a uma variedade de dispositivos e tamanhos de tela, mantendo tanto a funcionalidade quanto a estética do aplicativo.

### 3.1.2. Validação de Campos

As notificações de advertência são alertas exibidos após o usuário executar uma ação, indicando se a ação foi concluída com êxito e mantendo o usuário atualizado à medida que ele interage com a interface do aplicativo. Essa validação no front-end foi realizada usando o Parsley.js.

### 3.1.3. Mensagens de Aviso

Notificações de alerta são mensagens que surgem quando o usuário executa uma ação, comunicando se a ação foi realizada com sucesso e mantendo o usuário informado sempre que ele interage com a interface do aplicativo.

## 3.2. Banco de Dados MySQL (RNF02)

Um sistema de armazenamento de dados é uma compilação de informações organizadas de forma estruturada. Esse sistema serve como um repositório onde os dados são mantidos e administrados. O termo "relacional" indica que os dados são organizados em tabelas e que essas tabelas possuem relações entre si. Quando um software não suporta esse modelo, costumamos nos referir a ele como um Sistema Gerenciador de Banco de Dados (DBMS).

MySQL é um exemplo de DBMS, sendo um sistema de gerenciamento de banco de dados relacional de código aberto amplamente utilizado em diversas aplicações gratuitas para gerenciar suas bases de dados. O MySQL opera utilizando a linguagem SQL (Structured Query Language - Linguagem de Consulta Estruturada), que é a linguagem predominante para inserir, acessar e gerenciar o conteúdo armazenado em um banco de dados.

O MySQL é um RDBMS (Sistema de Gerenciamento de Banco de Dados Relacional) com um modelo de cliente-servidor. RDBMS refere-se a um tipo de software ou serviço utilizado na criação e administração de bancos de dados que seguem o modelo relacional.

**Vantagens:**

Uma das principais vantagens do MySQL é sua natureza de código aberto. Isso significa que qualquer pessoa pode baixar, usar e modificar o MySQL de acordo com suas necessidades.

* Código aberto.
* Facilidade de uso.
* Segurança.

## 3.3. Laravel (RNF03)

Laravel é um framework de desenvolvimento web em PHP que é conhecido por sua sintaxe fácil de entender e usar. Ele é projetado para simplificar o processo de criação de aplicativos web, fornecendo uma estrutura sólida e diversos recursos úteis. Com o Laravel, os desenvolvedores podem se concentrar na criação de suas aplicações sem se preocupar com os detalhes técnicos complicados. O framework oferece recursos como injeção de dependência, uma maneira elegante de lidar com bancos de dados, filas de execução de tarefas, testes automatizados e muito mais. Seja você um iniciante ou um desenvolvedor experiente, o Laravel é uma excelente escolha, pois pode ajudá-lo a começar no mundo do desenvolvimento web ou a aprimorar suas habilidades existentes.

**Vantagens:**

A melhor coisa em usar o Laravel é que ele é extremamente simples para um iniciante, mas oferece muitos recursos avançados para um programador profissional.

* É executado em várias plataformas (Windows, Linux, Unix, Mac OS X, etc.).
* É compatível com quase todos os servidores usados ​​atualmente (Apache, IIS, etc.).
* Suporta uma ampla variedade de bancos de dados.
* É gratuito.
* É fácil de aprender e funciona com eficiência no lado do servidor.

## 3.4. Frameworks (RNF04)

Frameworks são estruturas compostas por um conjunto de códigos genéricos que permite o desenvolvimento de sistemas e aplicações. Um framework funciona como uma espécie de template ou modelo que, quando utilizado, oferece certos artifícios e elementos estruturais básicos para a criação de alguma aplicação ou software.

O framework utilizados no projeto foi:

* Tailwind.

### 3.4.1. Tailwind

Tailwind CSS funciona verificando todos os seus arquivos HTML, componentes JavaScript e quaisquer outros modelos em busca de nomes de classes, gerando os estilos correspondentes e, em seguida, gravando-os em um arquivo CSS estático.

**Vantagens:**

* Utiliza um fluxo de trabalho que prioriza o utilitário para construir componentes complexos a partir de um conjunto restrito de utilitários primitivos.
* Estiliza elementos em estado interativos, como passar o mouse, focar e muito mais, usando modificadores condicionais.
* Gerencia a duplicação e mantém os projetos sustentáveis criando abstrações reutilizáveis.
* Cria interfaces de usuário totalmente responsivas que se adaptam a qualquer tamanho de tela usando modificadores responsivos.

## 3.5. Requisítos Mínimos Para o Uso (RNF05)

São os requisitos de versão mínima do navegador para rodar o site, compatibilidade com os principais navegadores (Chrome, Firefox, Safari, Edge), requisitos de hardware e software para o servidor e os dispositivos clientes.

### 3.5.1. Tailwind

* Navegadores alternativos que usam a versão mais recente do WebKit, Blink ou Gecko (tanto diretamente, quanto através da API web view da plataforma) não suportam explicitamente o Bootstrap.
* Suporte a navegadores modernos como Google Chrome, Mozilla Firefox, Safari, Microsoft Edge.
* Suporte básico ao Internet Explorer 11 (embora o Bootstrap 5 tenha removido o suporte oficial, projetos legados podem exigir).
* Organização do projeto que permita a inclusão dos arquivos principais do Tailwind (CSS) e suas dependências, como PurgeCSS (caso necessário).

### 3.5.2. Laravel

* Configuração adequada do servidor para processar arquivos PHP (geralmente via módulo ou FastCGI).
* Apache, Nginx, IIS ou qualquer outro servidor web compatível com PHP.
* Composer instalado
* O Laravel é compatível com diversos sistemas operacionais, incluindo Linux, Windows, macOS e outros sistemas baseados em Unix.

#### 3.5.2.1. Laravel 11.0x

* Servidor Web: Apache 2.4 ou Nginx 1.18.
* Sistema Operacional: Ubuntu 20.04, Windows 10, macOS Catalina.
* Sistema Operacional: Ubuntu 20.04, Windows 10, macOS Catalina.
* Extensões: ctype, fileinfo, mbstring, openssl, PDO, tokenizer, xml.
* Banco de Dados: MySQL 5.7+, MariaDB 10.2+, PostgreSQL 10+, SQLite 3.31+.
* Ferramentas: Composer, Git.

## 4. Referências Bibliográficas

https://laravel.com/

https://pt.linkedin.com/pulse/tailwind-css-pontos-positivos-e-negativos-marco-antonio  
https://medium.com/lfdev-blog/como-escrever-requisitos-de-software-de-forma-simples-e-garantir-o-m%C3%ADnimo-de-erros-no-sistema-app-74df2ee241cc

https://querobolsa.com.br/revista/requisitos-funcionais-e-nao-funcionais