

Universidade de Brasília

Departamento de Ciência da Computação



Documento de Projeto / BD 2023.2

**Sistema de Gerenciamento de Materiais para um
Laboratório Didático**

Autores:

LUIZ FREDERICO PINHO TORRES 222011623

LUIS FELIPE HILARIO CARMO 212007153

GABRIEL SANTOS PEREIRA 140140671

RYAN REIS FONTENELE 211036132

Brasília

1 de dezembro de 2023

Conteúdo

1	Introdução	2
1.1	O que é um Banco de dados?	2
1.2	Repositório do Projeto	2
1.3	Vídeo de Apresentação	2
2	Tecnologias Utilizadas	2
3	Modelagem do Banco de Dados	4
4	Como instalar as ferramentas	6
4.1	SQLite	6
5	Node.js	7
6	React	8
7	Tailwind CSS	8
7.1	Inicie um Projeto	8
7.2	Configuração Inicial	8
7.3	Configuração do Tailwind CSS	8
7.4	Configuração do PostCSS	8
7.5	Crie um arquivo CSS	9
7.6	Adicione Estilos HTML	9
7.7	Compilação dos Estilos	9
7.8	Compile os Estilos	9
7.9	Link para o Arquivo CSS	9
7.10	Teste e Personalize	10
8	Referências	10

1 Introdução

1.1 O que é um Banco de dados?

Um banco de dados é um sistema organizado e estruturado que armazena, gerencia e fornece acesso a informações e dados. Ele é projetado para armazenar dados de forma eficiente, permitindo a recuperação, atualização e análise desses dados de maneira rápida e confiável. Os bancos de dados são amplamente utilizados em diversas aplicações e setores, incluindo negócios, ciência, governo e muitos outros, para armazenar informações estruturadas, como registros de clientes, transações financeiras, informações de produtos e muito mais. Eles desempenham um papel fundamental na organização, recuperação e análise de informações para suportar operações diárias e tomada de decisões. Existem vários tipos de sistemas de gerenciamento de banco de dados (SGBDs), como MySQL, Oracle, SQL Server e MongoDB, que oferecem funcionalidades diferentes e são escolhidos com base nas necessidades específicas de cada aplicação.

1.2 Repositório do Projeto

O repositório para o projeto de Sistema de Gerenciamento de Materiais para um Laboratório Didático se encontra na plataforma Github, <https://github.com/reisryan/ProjetoBD2023.2>.

1.3 Vídeo de Apresentação

2 Tecnologias Utilizadas

Neste projeto, desenvolvemos um "Sistema de Gerenciamento de Materiais para um Laboratório Didático" que consiste em uma aplicação web com duas partes principais: o "front-end" e o "back-end", juntamente com um banco de dados.

- **Front-End:** O "front-end" é a parte da aplicação com a qual os usuários interagem diretamente. Ele é responsável pela interface gráfica, layout e experiência do usuário. Inicialmente, optamos por utilizar o framework "React", uma biblioteca de JavaScript de código aberto amplamente usada para construir interfaces de usuário dinâmicas e responsivas. No entanto, após uma reavaliação, decidimos utilizar "HTML" em conjunto com "CSS". Essa decisão foi baseada em vários fatores. Primeiramente, temos uma maior familiaridade com as

linguagens e estruturas do HTML e CSS. Além disso, poderíamos implementar a interface com maior eficiência, pois não precisaríamos utilizar bibliotecas adicionais para escrever o HTML, como seria necessário na implementação do front-end com “React”. Finalmente, optamos pelo HTML e CSS devido à quantidade de materiais de apoio disponíveis. Como são linguagens bem consolidadas, elas apresentam suporte universal e não necessitam de dependências extras. Isso nos permite focar na criação de uma interface de usuário eficaz e agradável, sem a necessidade de gerenciar complexidades adicionais;

- **Back-End:** O “back-end” é a parte invisível da aplicação que lida com a lógica de negócios, processamento de dados e interações com o banco de dados. Para a implementação do “back-end” escolhemos o framework “Node.js”, que é um ambiente de tempo de execução de JavaScript no servidor. Ele permite a criação de aplicativos web do lado do servidor usando JavaScript. Usamos nossa própria API para fornecer endpoints para o “front-end” se comunicar com o “back-end”. A API define como os dados são solicitados e como as respostas são formatadas;
- **Banco de Dados (SGBD):** O Sistema de Gerenciamento de Banco de Dados (SGBD) escolhido foi o “SQLite”, que é um sistema de gerenciamento de banco de dados relacional de código aberto. Ele é uma opção leve e amplamente utilizada para armazenar dados. O banco de dados é usado para armazenar e recuperar informações relacionadas aos materiais do laboratório, como inventário, registros de uso e outras informações importantes. No entanto, apesar das vantagens do SQLite, optamos pelo MySQL em nossos projetos. O MySQL é um sistema de gerenciamento de banco de dados relacional robusto e altamente escalável. Ele oferece uma série de recursos avançados, como suporte a transações, alta disponibilidade, segurança aprimorada e uma ampla gama de tipos de dados. Além disso, o MySQL tem uma grande comunidade de usuários e desenvolvedores, o que significa que há uma abundância de documentação, tutoriais e soluções para problemas comuns disponíveis. Em resumo, embora o SQLite seja uma excelente opção para aplicações mais leves e menos complexas, o MySQL é preferido para aplicações que exigem mais robustez, escalabilidade e recursos avançados;
- No contexto de uma aplicação web, a **comunicação entre o “front-end” e o “back-end”** ocorre por meio de solicitações HTTP. O “front-end” faz solicitações para a API do “back-end” para buscar ou enviar dados. O “back-end” processa essas solicitações, realiza operações no

banco de dados conforme necessário e envia respostas de volta ao "front-end". Essa conexão entre "front-end" e "back-end" permite que os usuários interajam com o sistema e acessem informações de maneira segura e eficiente. O uso de tecnologias como "React" e "Node.js" facilita o desenvolvimento ágil e escalável desse tipo de aplicação.

3 Modelagem do Banco de Dados

A criação de uma modelagem ao banco de dados é uma prática útil para visualizar e conceituar relacionamentos e entidades, conduzindo a uma melhor compreensão das funcionalidades do sistema em relação ao banco de dados.

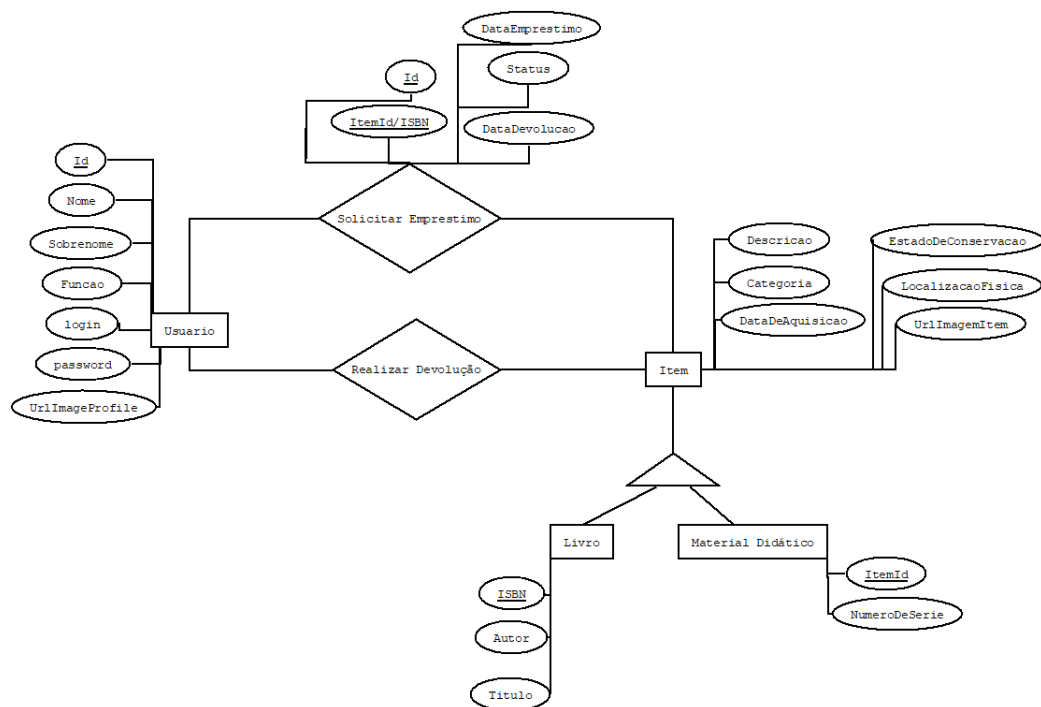


Figura 1: Modelo Entidade-Relacionamento

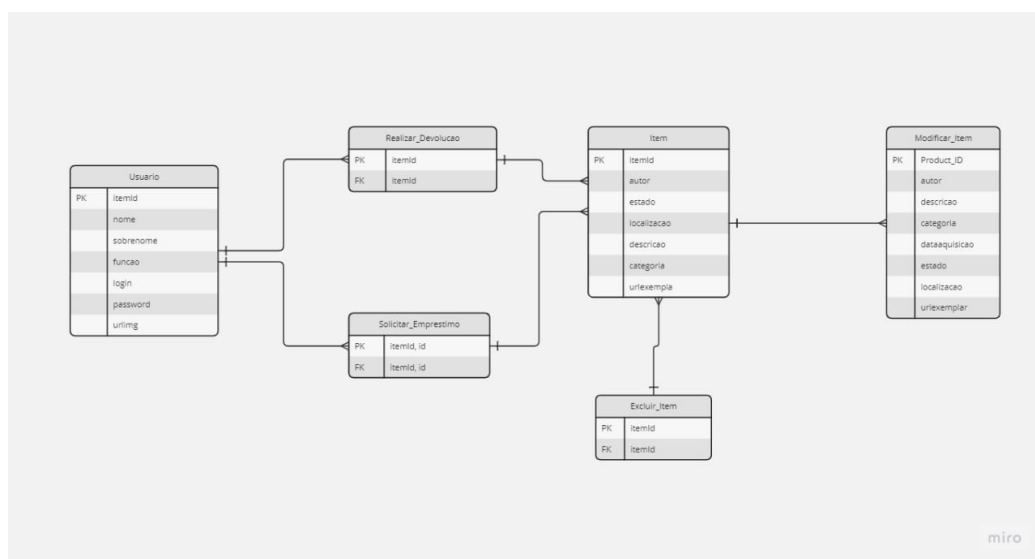


Figura 2: Modelo Relacional

```

Cadastro(PRIMARY key: itemId:int; autor: varchar, descricao:
    varchar, categoria: varchar,
dataaquisicao: date, estado: varchar, titulo: varchar,
    localizacao: varchar, urlexemplar: varchar) - User Autorizado

ModificarCadastro(PRIMARY key: itemId:int; autor: varchar,
    descricao: varchar, categoria: varchar, dataaquisicao: date,
    estado: varchar, localizacao: varchar, titulo: varchar,
    urlexemplar: varchar, FOREIGN key: itemId) - User Autorizado

ExcluirCadastro(PRIMARY key: itemId:int, FOREIGN key: itemId) -
    User Autorizado

SolicitarEmprestimo(dataEmprestimo: date, dataDevolucao: date,
    status: varchar, PRIMARY keys: id: int, itemId: int;
    Foreign keys: id, itemId)

RealizarDevolucao(PRIMARY keys: itemId, Foreign keys: itemId)
Usuario(PRIMARY key id: int, nome: varchar, sobrenome: varchar,
    funcao: int,
    login:varchar, password:varchar, urlimg: varchar)

```

4 Como instalar as ferramentas

4.1 SQLite

Para instalar o SQLite no Windows, primeiro, devemos acessar a página de downloads do SQLite. Em seguida, devemos procurar os arquivos pré compilados para Windows e baixar o “sqlite-tools”;

Logo após baixá-lo, teremos um arquivo compactado com um nome semelhante a este: “sqlite-tools-win32-x86-3390200.zip”. Vamos copiar este arquivo para a Unidade C;

Descompacta-se o Arquivo da Unidade C. Clique com o botão direito em cima do arquivo e selecione a opção “Extrair aqui”, Obs: “Para este caso, foi utilizado o programa WinRAR para descompactar os arquivos”;

Após selecionar a opção “Extrair aqui”, os novos arquivos estarão em uma pasta com um nome semelhante a este: “sqlite-tools-win32-x86-3390200”.

Podemos renomear esta pasta para ficar com um nome mais curto e legível para nós, vamos renomeá-la para “sqlite3”. Ao abrir a pasta “sqlite3” veremos 3 arquivos; Feito isso, precisamos que o Windows reconheça o SQLite

através do Prompt de Comando. Para isso, devemos adicionar uma variável de ambiente.

Para adicionar uma variável de ambiente, pesquise na barra de busca do Windows: “Editar as variáveis de ambiente do sistema”, Em “Propriedades do Sistema”, clique no botão com a opção “Variáveis de Ambiente. Em “Variáveis do sistema”, vamos encontrar a variável “Path” e clicar no botão com a opção “Editar”. Na sequência, vamos clicar em novo, e colar o caminho da pasta dos arquivos do SQLite que foram baixados:

C : /sqlite3.

Por fim, clique em OK.

Para verificar se foi instalado, vamos utilizar o Prompt de Comando do Windows. Para abri-lo, você pode utilizar o atalho do seu teclado através do botão windows juntamente com a tecla r (win+r). Ao utilizar este atalho, essa janelinha irá aparecer no canto inferior esquerdo da sua tela. Em seguida, basta digitar cmd e clicar em OK; Com o Prompt de Comando aberto, digite:

sqlite3

Ao executar o comando, você verá na tela a versão do programa, um texto informativo e a linha de comando do SQLite iniciada.

5 Node.js

Aqui estão os passos resumidos para instalar o Node.js no Windows:

1. Baixe o instalador do Node.js do site oficial.
2. Execute o instalador clicando duas vezes no arquivo baixado.
3. Clique em "Next" na janela do instalador.
4. Aceite os termos de uso marcando a caixa de seleção e clique em "Next".
5. Escolha o local de instalação padrão e clique em "Next".
6. Selecione as funcionalidades que deseja instalar (recomendado manter as opções padrão) e clique em "Next".
7. Escolha a opção para instalar as ferramentas para módulos nativos e o Chocolatey (recomendado) e clique em "Next".
8. Clique em "Install" para iniciar a instalação.
9. Aguarde a conclusão da instalação.
10. Verifique a instalação abrindo o Prompt de Comando e digitando 'node -version' e 'npm -version'. Certifique-se de reiniciar o computador após a instalação.

Agora você tem o Node.js e o npm instalados no seu sistema Windows.

6 React

Verifique o Node.js e o npm: Antes de instalar o React, verifique se você tem o Node.js e o npm instalados no seu sistema. Você pode fazer isso executando os seguintes comandos no seu terminal: `"$node -version"` e `"$npm -version"`. Certifique-se de que esses comandos retornem as versões corretas do Node.js e do npm. Se não tiver o Node.js instalado, você pode seguir o tutorial anterior para instalá-lo. Com o Node instalado, o React já vem automaticamente.

7 Tailwind CSS

7.1 Inicie um Projeto

Primeiro, você precisa ter um projeto web em que deseja usar o Tailwind CSS. Você pode criar um novo projeto ou adicionar o Tailwind CSS a um projeto existente.

7.2 Configuração Inicial

Para começar, abra um terminal na raiz do seu projeto e execute o seguinte comando para criar um arquivo de configuração do Tailwind CSS:

```
npm install tailwindcss postcss autoprefixer
```

7.3 Configuração do Tailwind CSS

Em seguida, crie um arquivo de configuração do Tailwind CSS executando o seguinte comando:

```
npx tailwindcss init -p
```

Isso criará um arquivo chamado `'tailwind.config.js'` na raiz do seu projeto.

7.4 Configuração do PostCSS

Crie um arquivo de configuração do PostCSS chamado `'postcss.config.js'` na raiz do seu projeto e adicione o seguinte código:

```
javascript module.exports = { plugins: [tailwindcss, autoprefixer], }
```

7.5 Crie um arquivo CSS

Crie um arquivo CSS (por exemplo, 'styles.css') na pasta do seu projeto e adicione o seguinte código para importar o Tailwind CSS:

```
css@import'tailwindcss/base';@import'tailwindcss/components';@import'tailwindcss/utilities';
```

7.6 Adicione Estilos HTML

Agora você pode começar a usar as classes do Tailwind CSS no seu HTML. Por exemplo:

```
html <divclass="bg-blue-500p-4"><pclass="text-white">Olá,TailwindCSS!<p></div>
```

7.7 Compilação dos Estilos

Para compilar seus estilos CSS, adicione um script no arquivo 'package.json' para executar o processo de construção. Abra o 'package.json' e adicione o seguinte script:

```
"scripts": {"build-css": "postcssstyles.css -ooutput.css"}
```

7.8 Compile os Estilos

Agora, execute o seguinte comando para compilar seus estilos CSS:

```
npmrunbuild -css
```

Isso criará um arquivo 'output.css' com seus estilos finais.

7.9 Link para o Arquivo CSS

Por fim, inclua o arquivo CSS compilado no seu projeto HTML:

```
html <linkrel="stylesheet"href="output.css">
```

7.10 Teste e Personalize

Agora você pode testar seu projeto e começar a personalizar os estilos usando as classes do Tailwind CSS. Consulte a documentação oficial do Tailwind CSS em <https://tailwindcss.com/docs> para aprender mais sobre as classes disponíveis e como personalizar o Tailwind CSS para atender às necessidades do seu projeto.

Isso conclui o tutorial básico para começar a usar o Tailwind CSS no seu projeto web.

8 Referências

1. I. Araújo e R. Silvério. “Como instalar o Node.js no Windows, Linux e macOS — Alura”. Alura. Consult. 2023-09-26. [Em linha]. Disponível: <https://l1nq.com/F1v5L>
2. V. Louzada. “SQLite: da instalação até sua primeira tabela — Alura”. Alura. Consult. 2023-09-26. [Em linha]. Disponível: <https://www.alura.com.br/artigos/sqlite-da-instalacao-ate-primeira-tabela>
3. “SQLite Home Page”. SQLite Home Page. Consult. 2023-09-26. [Em linha]. Disponível: <https://www.sqlite.org/index.html>
4. “Installation – React”. React. Consult. 2023-09-26. [Em linha]. Disponível: <https://react.dev/learn/installation/try-react>