

Teste Para a Escriba Informática V1

Projeto feito como teste para vaga de desenvolvedor Front-End.

Rodando o Projeto

Utilizando o terminal, navegue ate a pasta do mesmo, a seguir digite os seguintes comandos:

`npm i` - Instala as dependências necessárias para o Front-End.

`npm install -g json-server` - Instala o serviço Json-Server para rodar o Back-End

Após a instalação de todas as dependências, abra uma nova aba ou janela de seu terminal e digite:

`npm run dev`

`json-server --watch data.json`

Desenvolvimento da Aplicação

Bibliotecas

A aplicação utiliza as seguintes bibliotecas

- [Vue.js](#) - Framework de frontend utilizado para construir a interface do usuário.
- [Vite](#) - Build tool utilizado para compilar e empacotar o código do projeto.
- [Axios](#) - Biblioteca utilizada para fazer requisições HTTP.
- [Vue Router](#) - Biblioteca utilizada para gerenciar as rotas da aplicação.
- [FontAwesome](#) - Biblioteca de ícones amplamente utilizada na web.
- [Sass](#) - Processador de CSS, implementando novas funcionalidades para o mesmo
- [Vue SweetAlert2](#) - Integração da biblioteca SweetAlert2 com o framework Vue.js.

Regras de SCSS

A nomenclatura das fontes foi criada apartir da metodologia [BEM \(Block Element Modifier\)](#), porém adaptada para gerar dois níveis de navegação. Em suma a metodologia funciona da seguinte forma:

```
.pai{ //Bloco(Block)
  .pai__filho{ //Elemento(Element)
    &.pai__filho--modificado{} //Modificador(Modifier)
  }
}
```

Graças a essa metodologia ao olhar um único componente do código sabemos a quem é seu pai e como onde utiliza-lo.

Layout

O layout da aplicação se baseia em cards. Todavia, como não possui imagens específicas, é mostrado ícones ilustrativos para cada card.

Todas as páginas são responsivas, permitindo a utilização do sistema em diversos dispositivos.

Pastas e Arquivos

- Assets: Assets do aplicativo, em sistemas maiores possuiria os arquivos de fonte, imagens.
- Axios: Arquivo de exportação de regras do Axios, de fora a facilitar a utilização do mesmo na aplicação.
- Componentes: Arquivos dos componentes da aplicação
 - Cards: Cards de cada página, assim facilitando sua utilização em outras páginas e escalonagem
 - Layout: Arquivos mais genéricos do sistema.
 - Ui: Arquivos das Interfaces do Usuário, diferentemente dos cards esses arquivos esperam receber alguma ação.
- Router: Arquivo de rotas da aplicação
- Sass: Arquivos de SCSS da aplicação
- Utils: Arquivo de funções que podem ser utilizadas em diversos pontos da aplicação.
- Views: Arquivo de páginas da aplicação.

- App.vue: Arquivo base da aplicação, não sendo necessário sua edição
- Main.js: Arquivo de importações das bibliotecas e componentes da aplicação.

Ideias para V2

- CRUD de Produtos
- CRUD de Pedidos
- Adicionar Pinia/Vuex para gerenciamento de carrinho de compra