

DESENVOLVIMENTO DE UMA LOJA VIRTUAL

Ana Luisa Teixeira Costa	11218963
Cristiano Carvalho Lacerda	8531737
Diany Pressato	11218820
Matheus Araujo	9779026
Murilo Pratavieira	8123082

Disciplina: SCC0219
SCC0219 - Introdução ao Desenvolvimento Web
Profa. Kamila Rios

1. Requisitos do Sistema

As seguintes funcionalidades serão implementadas:

- Usuários devem ser capazes de acessar o sistema, se registrarem e de realizarem compras de produtos universitários, tais como vestuário, acessórios e brindes.
- Os usuários devem ser capazes de selecionarem os produtos, suas quantidades e os adicionarem ao carrinho de compras.
- Os usuários devem ser capazes de buscar produtos por nome ou categorias, além de filtrarem os produtos mostrados por diversos critérios como preço, tamanho, cor.
- A compra deve ser realizada por cartão de crédito e o carrinho esvaziado após o pagamento. Os produtos vendidos devem ser retirados do estoque.
- Administradores devem ser capazes de adicionar e gerenciar os produtos vendidos e de criar novos administradores, além de entrarem no sistema via login.
- O sistema deve ser responsivo, ter boa usabilidade e recurso de acessibilidade.

2. Descrição do Sistema

O frontend da loja foi implementado utilizando os frameworks vueJS e vuetify e uma arquitetura de single page application. O vue foi escolhido em detrimento do React e Angular por ser mais simples com uma curva de aprendizado menor, mas igualmente poderoso e provavelmente mais flexível.

Além disso, o pouco boilerplate necessário assim como os componentes single file o tornam razoavelmente agradável de trabalhar. Toda a aplicação foi construída em arquivos com a extensão .vue que combinam html, js e css em um único arquivo.

Todas as páginas são um único componente renderizados a partir vue-router. As rotas disponíveis e os respectivos componentes são:

```
routes: [  
  {  
    path: "/",  
    component: Home,  
    name: "Home"  
  },  
  {  
    path: "/produtos",  
    component: ListaDeProdutos,  
    name: "ListaDeProdutos"  
  },  
  {  
    path: "/produto",  
    component: Produto,  
    name: "Produto"  
  },  
  {  
    path: "/carrinho",  
    component: Carrinho,  
    name: "Carrinho"  
  },  
  {  
    path: "/registrar",  
    component: Registrar,  
    name: "Registrar"  
  },  
  {  
    path: "/finalizar",  
    component: Finalizar,  
    name: "Finalizar"  
  },  
  {  
    path: "/entrar",  
    component: Login,  
    name: "Entrar"  
  }  
],  
...
```

A biblioteca de interface de usuário Vuetify foi escolhida por ser de fácil usabilidade, estética diferenciada e por imbutir vários recursos de acessibilidade em seus elementos.

O arquivo app.vue contém o cabeçalho e rodapé, implementados utilizando as tags existentes para esse fim do vuetify, além da tag router-view.

Mesmo sem ainda sem existência de um backend todo o conteúdo apresentado é dinâmico - sendo definido como objetos json no código que podem ser facilmente substituído por respostas de requisições http na próxima entrega - e "bindados" no html.

A responsividade das páginas é feita através do uso do sistema de grid e colunas.

A funcionalidade de login foi implementada no frontend, de maneira que o admin é redirecionado para a página de gerenciamento de produto enquanto os outros usuários não.

3. Instruções para Build

```
sudo apt-get install nodejs  
git clone (colocar repositório aqui)  
cd web2020  
npm install  
npm run serve
```

4. Acessibilidade

Idealmente, grupo levou em consideração os seguintes aspectos de acessibilidade:

- Evitar transmitir informações ao usuário apenas através das cores;
- Links sublinhados
- Botões pratonizados
- Aumento e diminuição de letras

Na prática, houve dificuldades técnicas que não permitiram a completude dos requisitos citados acima. Os detalhes sobre estas limitações são descritos no próximo item.

5. Desafios e Limitações

A instalação do Node e Vue não foi simples e intuitiva. Alguns membros do grupo relataram dificuldades de configurar seus ambientes de desenvolvimento em especial no sistema Windows.

Um dos problemas encontrado foi em relação ao recurso de usabilidade que se pretendia implementar de aumentar o tamanho da fonte a partir de um botão na interface. Ao fornecer uma grande quantidade de elementos, a biblioteca Vuetify acaba tornando inflexível boa parte dos elementos que oferece, como por exemplo, a estrutura “v-table” utilizada nas páginas administrativas não tinha suporte para permitir a alteração do tamanho da fonte via o “botão de acessibilidade” implementado pelo grupo.

A dificuldade está que muitos componentes utilizados definem um tamanho de fonte próprio por questões estéticas e de usabilidade do site. Desta forma devido ao modelo de *override* do CSS em que propriedades mais internas são prioritárias em relação as mais externas, não é possível simplesmente mudar o tamanho da fonte no body do html, já que os elementos mais aninhados vão utilizar suas próprias definições.

Para tentar resolver alguns dos problemas relacionados à inflexibilidade do Vuetify, uma possível alternativa necessária foi propagar a mensagem entre os componentes para que cada um atualize sua própria propriedade de tamanho da fonte. A solução para este problema envolve o uso do vuex com a criação de um estado que indica se a fonte deve ser aumentada ou não. Não houve porém, tempo hábil de estudar a documentação desta biblioteca, que é considerada avançada, como mostrado na própria documentação desta.