



UniSENAI

Trilha 10

Projeto final da Trilha

Avaliação Prática - Desenvolvimento de Aplicação Completa com Vue.js

Cenário Hipotético

Você foi contratado por uma startup de e-commerce que deseja lançar um painel administrativo para gerenciar seus produtos. O objetivo é desenvolver uma aplicação completa utilizando os conhecimentos adquiridos no curso.

A aplicação deve conter autenticação segura, gerenciamento de produtos, consumo de APIs públicas e layouts responsivos utilizando Vuetify ou Bootstrap.

Requisitos do Sistema

O sistema deverá ser composto por:

1 Autenticação de Usuários com JWT

- Implementar um sistema de login e registro de usuários.
- Apenas usuários autenticados poderão acessar o painel administrativo.
- Proteger rotas utilizando autenticação JWT.
- Permitir logout seguro.

2 Gerenciamento de Produtos

- Criar um painel para cadastro, edição, exclusão e listagem de produtos.
- Os produtos devem ter nome, preço, categoria, imagem e descrição.
- Utilizar uma API RESTful para manipulação dos produtos.
- Exibir uma tabela dinâmica para listar os produtos com opções de filtro e paginação.

3 Consumo de APIs Públicas para Dados Dinâmicos

- Implementar integração com uma API pública para buscar informações dinâmicas.
- Exemplo: Usar a API do OpenWeatherMap para exibir a previsão do tempo no painel administrativo.
- Exibir mensagens de erro amigáveis caso a API esteja indisponível.


4 Layout Responsivo e Moderno

- Criar uma interface amigável utilizando Vuetify e Bootstrap.
 - Garantir que o sistema seja responsivo para diferentes tamanhos de tela.
 - Utilizar componentes do Vuetify para tabelas, formulários, botões e modais.
 - Utilizar Bootstrap Grid System para organização do layout.
-

Regras e Diretrizes






- ◆ O front-end deve ser desenvolvido utilizando Vue.js.
 - ◆ O back-end pode ser implementado em Node.js com Express e MySQL ou PHP com Slim Framework e MySQL.
 - ◆ O Vuex deve ser utilizado para gerenciamento de estado do usuário e produtos.
 - ◆ O Axios deve ser utilizado para chamadas às APIs.
 - ◆ A interface deve seguir os princípios de Clean Code, priorizando código limpo e bem estruturado.
 - ◆ O código deve seguir um padrão de formatação, utilizando ESLint e Prettier.
 - ◆ O sistema deve tratar erros e mensagens de forma amigável ao usuário.
-

Entrega do Projeto


-  O projeto deve ser entregue com:
- ✓ Código-fonte organizado (seguindo a estrutura recomendada no curso).
 - ✓ README.md explicando como rodar a aplicação.
 - ✓ Demonstração funcional do sistema.
 - ✓ Autenticação JWT funcionando corretamente.
 - ✓ Painel de gerenciamento de produtos completo.
 - ✓ Integração com API pública para exibição de dados dinâmicos.
 - ✓ Layout responsivo e profissional utilizando Vuetify e Bootstrap.
-

Avaliação

A avaliação será baseada nos seguintes critérios:

- ✓ Funcionamento correto da aplicação 
 - ✓ Implementação dos requisitos obrigatórios 
 - ✓ Código organizado e seguindo boas práticas 
 - ✓ Uso correto das tecnologias ensinadas no curso 
 - ✓ Interface moderna e responsiva 
-

Desafio Extra (Opcional)

-  Para quem deseja ir além, implemente as seguintes melhorias:
- ◆ Criar upload de imagens para os produtos.
 - ◆ Implementar dark mode no layout utilizando Vuetify.
 - ◆ Criar um dashboard analítico com gráficos dinâmicos.
 - ◆ Permitir exportação da lista de produtos em PDF ou CSV.
-

Agora é com você!

Utilize todo o conhecimento adquirido no curso para desenvolver uma aplicação completa, segura e profissional. 