

Relatório do Projeto: Website de Meteorologia

Relatório do Projeto Web: Aplicação de Meteorologia

Disciplina: Tecnologias Web

Aluno: Tiago Silva [40220319]

Data: 24/01/2025

Relatório do Projeto: Website de Meteorologia

Relatório do Projeto: Website de Meteorologia

Resumo do Projeto:

Este relatório descreve o desenvolvimento de um site de meteorologia que utiliza uma API para obter dados climáticos em tempo real e gerenciar o histórico de previsões do usuário. Inclui detalhes sobre o contexto, objetivos, recursos utilizados e desafios enfrentados durante o desenvolvimento.

1. Contexto do Projeto

O projeto consiste no desenvolvimento de um website de meteorologia que permite aos usuários obterem informações climáticas, como temperatura, umidade, descrição do clima e o seu ícone correspondente, para qualquer cidade pesquisada. Além disso, o site conta com funcionalidades para gerenciar o histórico de buscas dos usuários e confirmar previsões, oferecendo uma experiência personalizada. Esse projeto utiliza a API OpenWeatherMap para obter dados climáticos em tempo real e a MockAPI para gerenciar usuários e o histórico de pesquisas.

2. Explicação Geral dos Objetivos do Projeto

Os principais objetivos do projeto foram:

1. Criar um site de meteorologia interativo:

- O usuário pode realizar login, pesquisar cidades e visualizar informações climáticas atualizadas.

2. Gerenciar histórico personalizado:

- O site armazena até duas cidades pesquisadas recentemente no histórico do usuário.

3. Integrar APIs externas e locais:

- Utilizar a API pública OpenWeatherMap para obter dados climáticos.
- Utilizar a MockAPI para simular um banco de dados de usuários e histórico.

4. Permitir a atualização de previsões:

- As previsões armazenadas podem ser confirmadas ou atualizadas com dados mais recentes.

Relatório do Projeto: Website de Meteorologia

3. Documentação dos Recursos Utilizados

3.1. API Pública: OpenWeatherMap

- Descrição: Fornece dados climáticos em tempo real, incluindo temperatura, umidade, descrição textual do clima, e ícones ilustrativos.

- Recursos utilizados no projeto:

- Endpoint:

https://api.openweathermap.org/data/2.5/weather?q={{cidade}}&appid={{chaveAPI}}&units=metric&lang=pt_br

- Dados obtidos:

- Temperatura em graus Celsius.
- Umidade (%).
- Descrição do clima (ex.: 'nublado').
- Ícone representando o clima.

- Método HTTP: GET

3.2. MockAPI

- Descrição: Ferramenta que simula um banco de dados para armazenar e manipular informações de usuários e históricos.

- Recursos criados:

- Endpoint de Usuários: <https://6787ca27c4a42c916108448a.mockapi.io/api/v1/users>

- Dados armazenados: id, email.

- Métodos: GET.

- Endpoint de Histórico: <https://6787ca27c4a42c916108448a.mockapi.io/api/v1/historico>

- Dados armazenados: userId, cidade, temperatura, descricaoClima, umidade, status, timestamp.

- Métodos: GET, POST, PUT, DELETE.

Relatório do Projeto: Website de Meteorologia

4. Desenvolvimento do Projeto

4.1. Problemas e Situações Encontradas

Durante o desenvolvimento, algumas dificuldades surgiram, sendo as principais:

1. Integração com a OpenWeatherMap API:

- Foi necessário compreender como enviar requisições HTTP utilizando o método fetch, passando parâmetros no URL para especificar a cidade e a chave da API.

2. Como implementar o método DELETE:

- Um desafio foi implementar, ou seja, adicionar alguma funcionalidade útil e interessante usando esse método. Para isso pensei em limitar o histórico aos últimos dois elementos.

3. Atualização de previsões no histórico.

4. Utilização do método PUT:

- Exigiu o uso do método PUT, onde as informações existentes são atualizadas conforme a vontade do utilizador.

5. Tratamento de erros:

- Foi necessário tratar erros, como cidade não encontrada, conexão falha com a API, e entrada de dados inválidos pelo usuário.

Problema: Limitação do histórico a dois itens mais recentes

O sistema foi projetado para limitar o histórico de previsões a apenas dois itens mais recentes por utilizador. No entanto, havia um problema em que, ao atingir o limite, a exclusão do item mais antigo nem sempre ocorria corretamente, o que resultava em inconsistências no número de itens exibidos.

Solução:

Foi implementada uma verificação adicional antes de adicionar novos itens ao histórico. A lógica foi ajustada para garantir que o item mais antigo fosse sempre identificado e removido antes da inserção de um novo.

Relatório do Projeto: Website de Meteorologia

Essa verificação foi testada com diferentes cenários para garantir a consistência do histórico.

4.2. Soluções Implementadas

- Troca dinâmica de fundo: Um array com URLs de imagens foi utilizado para alterar o plano de fundo do site a cada 7 segundos.
- Validação de inputs: Adicionados alertas e verificações para garantir que o campo de entrada de cidade não estivesse vazio antes de enviar uma requisição.
- Armazenamento local: O localStorage foi usado para armazenar o ID do usuário logado, permitindo personalizar o histórico.

5. Conclusão

Este projeto proporcionou uma experiência prática completa, abordando desde a construção da interface de usuário até a integração com APIs e o uso de métodos HTTP. O resultado foi um site funcional que cumpre os objetivos definidos e permite ao usuário buscar informações climáticas e gerenciar seu histórico de forma intuitiva.

6. Trabalho Futuro

Algumas melhorias e funcionalidades adicionais podem ser implementadas no futuro:

1. Autenticação real de usuários:

- Substituir a MockAPI por uma API que suporte login seguro com autenticação.

2. Design Responsivo:

- Melhorar o layout para dispositivos móveis, garantindo uma boa experiência em telas menores.

3. Gráficos e visualizações:

- Exibir gráficos históricos de temperatura ou umidade das cidades pesquisadas.

4. Mais informações climáticas:

- Expandir os dados exibidos, incluindo previsões de vários dias e horários.

Relatório do Projeto: Website de Meteorologia

5. Tradução de termos climáticos:

- Implementar um sistema de tradução mais completo para outros idiomas.