



Tecnologias WEB

Algoritmia

Projeto

Projeto realizador por:
Gustavo Fonsêca-40210415
Alfonso Jimenez-40210413

Apresentação do projeto

Nosso objetivo neste projeto é criar um site onde os utilizadores conseguem ver o clima na localização exata ou desejada, desenvolvida com HTML, JS e com utilização de bootstrap.

Chegamos ao nosso objetivo com algumas dificuldades,sendo que já temos experiência com API's. Tivemos uma boa adaptação às API's utilizadas fazendo assim um projeto completo, com opções de mapa e ativação da geolocalização.

Funcionalidades

Home Screen(inicio.html)

- Este código é em HTML e cria uma página web com uma seção de fundo azul escuro e um container. Dentro do container há um jumbotron (um componente grande e destacado) com um título e um texto. Abaixo do jumbotron há dois botões, um para abrir um mapa e outro para ir para a página de Clima e Tempo. Há também um link para o arquivo CSS chamado "styles.css" e um link para Bootstrap, uma biblioteca de estilo para ajudar a formatar a página. O script ao final da página verifica se o navegador tem suporte à Web Speech API e, se sim, usa essa API para fazer o navegador falar "BEM-VINDO AO NOSSO SITE SOBRE CLIMA E TEMPO" usando a voz "Google BR Portuguese".

Clima e tempo(climaetempo.html)

- Este código é em HTML e cria uma página web com um mapa (através de uma div com id "map" e classe "map") e um container. Dentro do container há um título e um parágrafo explicando o propósito da página, um botão para obter informações climáticas da localização exata do utilizador um formulário para que o utilizador informe uma localização específica e um botão para buscar informações climáticas sobre essa localização. Há também uma div para exibir informações climáticas, como umidade e velocidade do vento. Há um link para o arquivo CSS chamado "styles.css" e um link para Bootstrap, uma biblioteca de estilo para ajudar a formatar a página. E tem um script chamado script.js que é responsável por buscar as informações climáticas e mostrar no HTML.

Mapa(mapa.html)

- Este código é em HTML e cria uma página web com um mapa (através de uma div com id "map" e classe "map") e um container. Dentro do container há um botão para obter a localização exata do utilizador e um formulário para que o utilizador informe uma localização específica e um botão para buscar essa localização no mapa. Há também um link para o arquivo CSS chamado "stylee.css" e um link para Bootstrap, uma biblioteca de estilo para ajudar a formatar a página, e uma biblioteca leaflet que é responsável por mostrar o mapa. E tem um script chamado scriptt.js que é responsável por buscar a localização do utilizador e mostrar no HTML, e buscar a localização desejada no mapa.

Utilização da api OpenWeatherMap(script.js)

- Este código é em JavaScript e utiliza a API OpenWeatherMap para obter informações climáticas sobre uma localização específica. Ele usa a geolocalização do navegador para obter a localização atual do utilizador e exibe essas informações na página. Também há um formulário na página para que o utilizador possa buscar informações climáticas sobre uma localização específica. O código também contém algumas verificações para garantir que o navegador suporte geolocalização e tratamento de erros para caso algo dê errado. E tem uma função "goBack()" que volta para a página anterior.

Utilização de geolocation(scriptt.js)

- Este código é um script JavaScript que utiliza a API de geolocalização do navegador para obter a localização do utilizador e mostrar essa localização em um mapa utilizando o Leaflet.js. Quando a página é carregada, a função "getLocation" é chamada e verifica se o navegador suporta a geolocalização. Se sim, a função "navigator.geolocation.getCurrentPosition" é chamada e passa a função "showPosition" como um argumento. A função "showPosition" então pega a latitude e longitude da posição atual do utilizador e as usa para mostrar a localização do utilizador em um mapa.
O utilizador também pode procurar uma localização específica digitando-a no campo de entrada e clicando no botão "Procurar". Isso chama a função "searchLocation" que usa a API OpenCageData para obter a latitude e longitude da localização procurada e mostra essas informações na página. A localização procurada também é marcada no mapa.

CSS

- **styles:** Este é um arquivo CSS define as propriedades de fundo, alinhamento, sombra, cor, borda e posição para elementos como o corpo, o contêiner, o formulário, o botão de geolocalização e o botão de volta. Ele também define a aparência do hover para o botão "btn". Estes estilos ajudam a melhorar a aparência e a usabilidade da página.
- **stylee:** Este código é uma folha de estilo CSS que define estilos para vários elementos, como botões, caixas de texto, etc. Alguns dos estilos incluem:
 - O elemento body tem um fundo gradiente e é alinhado no centro da página.
 - O elemento com a classe "container" tem uma cor de fundo específica, sombra e borda arredondada.
 - O elemento com o id "voltar2" tem uma posição fixa no topo esquerdo da página.
 - O elemento com a classe "centralizar" é alinhado no centro da página.
 - O elemento com o id "procurar" tem uma fonte maior e negrito.
 - O elemento com o id "address" tem uma borda arredondada e sombra.
 - O elemento com a classe "parent" tem uma posição relativa e seus elementos filhos são posicionados de acordo com essa posição.
 - O elemento com o id "searchResult" é posicionado no topo direito da página.

Dificuldades

Inicialmente adicionamos o mapa porque não tínhamos conseguido colocar a localização exata do utilizador (na parte do clima). Tivemos também uma leve dificuldade de aprimorar o mapa, que seria a implementação da visão de satélite, e uma barra de pesquisa onde o utilizador pode pesquisar o endereço desejado, mas conseguimos.

A API do mapa seria uma forma de suprir a falta da API geolocation, uma vez que é necessário no mínimo 2 API'S no projeto, onde buscaria informações climáticas do local exato do utilizador.

Porém, não desistimos da geolocation na página de climas, e finalmente conseguimos criar um botão que quando é clicado mostra as informações climáticas exata de onde o utilizador se encontra.

Conclusão

Neste projeto concluímos que dá para manipular as API'S de diversas formas.

Um exemplo é a API geolocation, onde no mapa a longitude e latitude aparece na página quando pesquisamos o endereço desejado, já na página de clima e tempo pesquisamos a cidade escolhida e no console mostra a latitude e longitude da cidade, e a partir dessas informações aparece na página as informações climáticas. Resumindo, não há apenas uma forma de manipular qualquer API.