

DOCUMENTACION CLIENTE



Contenido

HEADER Y FOOTER:.....	1
PAGINA RUTAS Y MIS RUTAS.....	2
INFO RUTAS.....	3
INSERTAR RUTAS.....	4
EDITAR, INCIO SESION Y REGISTRARSE.....	5

HEADER Y FOOTER:

Aquí utilizo el js de forma que hace que si estas registrado aparezca un header y si no aparezca otro distinto, a través de esta función:

```
if (!localStorage.getItem('token')) {  
  header = "http://localhost/dwes/PROYECTO_2TRI/PAGINA1/footer/header.php";  
} else {  
  header = "http://localhost/dwes/PROYECTO_2TRI/PAGINA1/footer/header1.php";  
}
```

Y luego dos funciones una de añadir footer y otra de añadir header

```
function añadirFoot(url) {  
  fetch(url)  
    .then((response) => {  
      return response.text()  
    })  
    .then((data) => {  
      document.querySelector('#footer').innerHTML = data;  
    })  
}  
  
function loadHeader(url) {  
  fetch(url)  
    .then((response) => {  
      return response.text()  
    })  
    .then((data) => {  
      document.querySelector('#encabezado').innerHTML = data;  
      let cerrar = document.getElementById('cerrar')  
      cerrar.addEventListener('click', cerrarSesion)  
      function cerrarSesion(event) {  
        event.preventDefault();  
        localStorage.removeItem('token')  
        localStorage.removeItem('id')  
        window.location.href =  
( 'http://localhost/dwes/PROYECTO_2TRI/PAGINA1/pagian_inicio.php' )  
      }  
    })  
}
```

PAGINA RUTAS Y MIS RUTAS

Aquí llamamos a la api con un fetch de esta forma:

```
fetch(`http://localhost/dwes/PROYECTO_2TRI/APIS/rutas.php?name=${
nombre}&min_dist=${dist}&max_dist=${dist1}`, {
  method: 'GET',
  headers: {
    'Content-Type': 'application/json',
  },
})
.then(response => {
  switch (response.status) {
    case 200:
      console.log("SACASTE LAS RUTAS");
      return response.json();
      break;
    case 401:
      console.log("NO PUEDES INICIAR SESION");
  }
})
.then(data => {
})
```

Luego en el data hago cada ruta y un mapa para cada una de ellas, las llamo por un foreach y los mapas es otro foreach distinto

```
data.forEach(element => {
  let id = element.id
  let nombre = element.route_name;
  let distancia = element.distance
  let max_height = element.max_height;
  let min_height = element.min_height;
  str += `
<div id='routes'>
  <div id='map${id}' class='map'></div>
  <div id='nameRuta'><h4>${nombre}</h4></div>
  <div id='infoRuta'>
    <div id='dist'><a>Distancia: </a><a id='dat'>${distancia}</a></div>
    <div id='max'><a>Altura Maxima: </a><a id='dat'>${max_height}</a></div>
    <div id='min'><a>Altura Minima: </a><a id='dat'>${min_height}</a></div>
    <form
      action='http://localhost/dwes/PROYECTO_2TRI/PAGINA_RUTAS/info.php'>
      <div id='min'><button>Info Ruta</button></div>
      <input id="id" name="id" type="hidden" value="${id}">
    </form>
  </div>
</div>
`;
});
```

```

divRutas.innerHTML = str
data.forEach(element => {
  let id = element.id
  let latitud = element.start_lat;
  let longitud = element.start_lon;
  let map = L.map(`map${id}`, { zoomControl: false }).setView([latitud, longitud],
12);

  L.tileLayer("https://tile.openstreetmap.org/{z}/{x}/{y}.png", {
    maxZoom: 19,
    attribution: '&copy; <a
href="http://www.openstreetmap.org/copyright">OpenStreetMap</a>'
  }).addTo(map);
  var marker = L.marker([latitud, longitud]).addTo(map);
})

```

En las paginas de rutas solo si tienes token sale un botón que te lleva a añadir rutas, además de en la pagina de mis rutas solo te dejara acceder si tienes token, es decir si estas logeado

INFO RUTAS

Aquí genero un mapa grande con un marcador y los puntos para la ruta que recogo de la base de datos, el mapa se añade para todas así:

```

<link rel="stylesheet" href="https://unpkg.com/leaflet@1.9.3/dist/leaflet.css"
integrity="sha256-kLaT2GOSpHechhsozzB+flnD+zUyjE2LlfWPgU04xyl=" crossorigin="" />
<script src="https://unpkg.com/leaflet@1.9.3/dist/leaflet.js" integrity="sha256-
WBkoXOWTeyKclOHuWtc+i2uENFpDZ9YPdf5Hf+D7ewM=" crossorigin=""></script>

```

INSERTAR RUTAS

A esta pagina solo te deja entrar si tienes token, es decir si estas registrado, pasas el archivo gpx a json y reduces los números de puntos a 50 mas o menos de esta forma:

```
let fichergpx = document.querySelector('#fileInput');
document.getElementById('form').addEventListener('submit', function(event) {
  event.preventDefault()
  let fileInput = document.getElementById("fileInput")
  let file = fileInput.files[0];
  let reader = new FileReader();
  reader.onload = function() {
    let gpx = reader.result;
    let parser = new gpxParser();
    parser.parse(gpx)
    let json = parser.tracks[0]
    // console.log(json);

    // Reducimos el número de puntos a aproximadamente 50
    let ratio = Math.round(json.points.length / 50);
    let points = json.points.filter((_, index) => index % ratio == 0);
    let slopes = json.slopes.filter((_, index) => index % ratio == 0);
    let nombreInput = document.getElementById('nameInput')
    let nombre = nombreInput.value;
```

Luego lo paso a json y con un fetch de método post lo paso a la base de datos.

EDITAR, INCIO SESION Y REGISTRARSE

Aquí funcionan todas igual, hacen un fetch cada una y solo cambia el método que es post para inicio sesión y registrarte y put para editar perfil

```
fetch('http://localhost/dwes/PROYECTO_2TRI/APIS/api.php', {
  method: 'POST',
  headers: {
    'Content-Type': 'application/json; charset=utf-8'
  },
  body: usuario
})

.then(response => {
  switch (response.status) {
    case 201:
      console.log("REGISTRADO CON EXITO");
      break;
    case 409:
      console.log("USUARIO EXISTE");
      break;
    case 400:
      console.log("ERROR");
  }
  return response.json();
})

.then(data => {
  console.log(data);
})
```

Además está el método eliminar que está dentro de edit.js y elimina la cuenta pasando el nombre de usuario y contraseña además del token.