

*Sitio Web con login, modulo para creación y edición de usuarios con acceso al sitio.

Mostrar estadísticas de acceso:

Usuario, fecha, hora y lugar de acceso de cada usuario

Numero de acceso por usuario

Numero de acceso a cada pagina que compone el sitio.

Para resolver este problema se usaran las siguientes tecnologicas:

Html, JS, Bootstrap, Firebase

Primeramente se importara el framework bootstrap en el archivo index.html.

```
index.html > html
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Problema entrevista</title>
8   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
9     integrity="sha384-9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYYxFc+NcPb1dKGj7Sk" crossorigin="anonymous">
10 </head>
11
```

Luego se anadira un componente Navbar para organizar los botones de Sign In, Sign Up, y Log Out.

```
<!-- Nav -->
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  <a class="navbar-brand" href="#">Navbar</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
    aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>

  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav ml-auto">
      <li class="nav-item">
        <a class="nav-link" href="#">Sign In</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#" data-toggle="modal" data-target="#SUModal">Sign Up</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Log Out</a>
      </li>
    </ul>
  </div>
</nav>
```

Para cada boton del navbar, se añade un Modal para introducir los campos de texto necesarios.

Template de los modal utilizados para cada boton.

```
<!-- Modal -->
<div class="modal fade" id="SUModal" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="exampleModalLabel">Modal title</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
        <form id="signup-form">
          <div class="form-group">
            <input type="text" id="signup-email" class="form-control" placeholder="Email" required>
          </div>
          <div class="form-group">
            <input type="password" id="signup-password" class="form-control" placeholder="Password" required>
          </div>
          <button type="submit" class="btn btn-primary">Sign Up</button>
        </form>
      </div>
    </div>
  </div>
</div>
```

Se añaden los scripts para poder utilizar firebase y firestore

```
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
  integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+0GpamoFVy38MVBnE+IbbVYUew+0rCXaRkfj"
  crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.bundle.min.js"
  integrity="sha384-ho+j7jyWK8fNQe+A12Hb8AhRq26LrZ/JpcUGG0n+Y7RsweNrtN/tE3MoK7ZeZdyx"
  crossorigin="anonymous"></script>

<!-- Firebase -->
<!-- The core Firebase JS SDK is always required and must be listed first -->
<script src="https://www.gstatic.com/firebasejs/7.24.0/firebase-app.js"></script>

<!-- TODO: Add SDKs for Firebase products that you want to use
| https://firebase.google.com/docs/web/setup#available-libraries -->
<script src="https://www.gstatic.com/firebasejs/7.24.0/firebase-auth.js"></script>
<script src="https://www.gstatic.com/firebasejs/7.24.0/firebase-firestore.js"></script>

<script src="https://www.gstatic.com/firebasejs/7.24.0/firebase-analytics.js"></script>

<script>
  // Your web app's Firebase configuration
  // For Firebase JS SDK v7.20.0 and later, measurementId is optional
  var firebaseConfig = {
    apiKey: "AIzaSyDLLmajLJO_yIlag4t_HoMWVQeZT_ZWG6Y",
    authDomain: "entrevista-51974.firebaseio.com",
    databaseURL: "https://entrevista-51974.firebaseio.com",
    projectId: "entrevista-51974",
    storageBucket: "entrevista-51974.appspot.com",
    messagingSenderId: "656400975466",
    appId: "1:656400975466:web:523bb204787bdd372f2c13",
    measurementId: "G-DFC554P91E"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
  const auth = firebase.auth();
  const firestore = firebase.firestore();
  firebase.analytics();
</script>
<!-- Main JS -->
<script src="main.js"></script>
```

En esta seccion del codigo se procesan los datos de registro nuevo, que es bastante similar a la seccion de inicio de sesion:

```
//Sign Up part
const signupForm = document.querySelector('#signup-form');

signupForm.addEventListener('submit', async (e) => {
  e.preventDefault();
  console.log('submitting');
  const coords = await getCurrentPosition();

  const signupEmail = document.querySelector('#signup-email').value;

  const signupPassword = document.querySelector('#signup-password').value;

  fstore
    .collection('Fechas')
    .doc(signupEmail)
    .set({
      User: signupEmail,
      Location: coords,
      Date: new Date(Date.now()),
      Times: 1
    })
    .then(function () {
      console.log('Document successfully written!');
    })
    .catch(function (error) {
      console.error('Error writing document: ', error);
    });

  auth
    .createUserWithEmailAndPassword(signupEmail, signupPassword)
    .then((userCredential) => {
      //clear
      signupForm.reset();
      $('#SUModal').modal('hide');

      console.log('sign up');
      console.log(coords);
    });
});
```

Aqui se configuran los estados de la web dependiendo de si el usuario esta o no autenticado

```
//events
//is auth
auth.onAuthStateChanged((user) => {
  if (user) {
    fstore
      .collection('Fechas')
      .get()
      .then((snapshot) => {
        configFechas(snapshot.docs);
        authcheck(user);
      });
  } else {
    configFechas([]);
    authcheck(user);
  }
});
```