

Documentation

Index

Le fichier **index.html** correspond à la page principale du site. Il s'agit du point d'entrée permettant d'accéder aux différentes sections telles que l'espace administrateur, l'espace entreprise et le menu de connexion.

Admin

La section administrateur permet la gestion générale du site.

Le fichier **admin.html** propose un panneau d'administration donnant accès à des statistiques sur le nombre de comptes et le nombre d'offres disponibles.

Le fichier **admin_config.html** permet de consulter toutes les offres présentes sur le site ainsi que leurs contenus détaillés.

Le fichier **admin_navbar.html** contient la barre de navigation utilisée dans l'espace administrateur.

Le dossier **css** regroupe les feuilles de style nécessaires à l'affichage des éléments de base sur l'index et les autres pages du site.

Le fichier **profil.html** présente un modèle de page de profil utilisateur avec la possibilité de modifier ou de supprimer son profil.

Entreprise

La section entreprise est dédiée aux structures souhaitant publier et gérer leurs annonces.

Le fichier **entreprise.html** affiche une interface permettant la création d'une entreprise ainsi que la publication d'une annonce.

Le fichier **entreprise_config.html** offre une interface interne pour visualiser les informations liées à l'entreprise, notamment les candidatures reçues et les annonces postées.

Profile

La section profile est dédiée aux utilisateurs souhaitant gérer leurs coordonnées, modifier ou supprimer leur compte utilisateur.

Le fichier **profile.html** affiche une interface permettant une solution pour gérer tout ce qui est son profil enregistré (Adresse email, nom, prénom, mot de passe...).

Communication des données

Pour la communication des données entre le Front-end et le Back-end, le langage **JavaScript** sera utilisé.

Les échanges avec les fichiers PHP se feront à l'aide de fonctions asynchrones, permettant d'effectuer des requêtes vers le serveur sans recharger la page.

Ces fonctions permettront de récupérer les données depuis la base de données et de les afficher dynamiquement sur le site, assurant ainsi une interaction fluide entre l'interface utilisateur et les informations stockées côté serveur.

Voici quelques exemples:



```
1  async function enterpriseRecrutments() {
2      fetch('../php/application/ReceiveApplication.php')
3          .then(response => response.json())
4          .then(json => {
5              if (json.success) {
6                  console.log(json.data)
7                  displayRecrutments(json.data)
8              } else {
9                  console.error("Erreur dans la réponse:", json);
10             }
11         })
12     }
13 }
```

Ici nous utilisons cette fonction afin de récupérer les CV envoyés par les utilisateurs en liaison avec l'offre demandée.

```
1 async function lunchLoad() {
2   fetch('../php/offer/OfferDisplay.php')
3     .then(response => response.json())
4     .then(json => {
5       if (json.success) {
6         displayAnnounce(json.data);
7       } else {
8         console.error("Erreur dans la réponse:", json);
9       }
10    })
11  }
12 }
13 }
```

Cette fonction prends en compte le fait qu'on puisse afficher les annonces avec les détails récupérés via le fichier OfferDisplay.php en liaison avec la base de donnée. Cela va afficher donc les annonces de toutes les entreprises sur le index.html

```
1 function displayAnnounce(announce) {
2   const jobgrid = document.getElementById('jobgrid');
3   announce.forEach(div => {
4     const div2 = document.createElement('div');
5     console.log(div);
6     div2.innerHTML = `
7       <div class="">
8         <h3>${div.Titre}</h3>
9         <h4>${div.entreprise_nom || "Non précisé"}</h4>
10        <p>${div.Type}</p>
11      </div>
12      <div class="favAndCheck">
13        <div>
14          
15        </div>
16        <div class="d-flex justify-content-center align-items-center">
17          
18          <article>Vu</article>
19        </div>
20        <button class="btn btn-primary more-btn" id="more" onclick="">En savoir plus</button>
21      </div>
22    `;
23    div2.classList.add("announce-Card");
24    div2.classList.add("p-3");
25    div2.style.cursor = "pointer";
26    jobgrid.appendChild(div2);
27    const morebutton = div2.querySelector('.more-btn');
28    morebutton.onclick = () => displayDetails(div.Titre, div.entreprise_id, div.Type, div.Description, div.
    entreprise_nom, div.entreprise_id);
29  });
30  Profil_Loaded();
31  AdminMenu();
32 }
```

La fonction asynchrone `lunchLoad()` permet d'établir la communication avec les fichiers PHP et de récupérer les données issues de la base de données.

Une fois ces données obtenues, la fonction `lunchLoad()` appelle une seconde fonction, non asynchrone, nommée `displayAnnounce()`.

Cette dernière reçoit en paramètre les données transmises par la fonction asynchrone.

Grâce à ce mécanisme, il est possible d'analyser les informations envoyées par le PHP et de les afficher dynamiquement en HTML sur le site.

A code editor window with a dark background and light-colored text. It contains a JavaScript function named `AdminVerification` which is asynchronous. The code uses `fetch` to get data from a PHP file, checks if the user is an admin, and redirects to an index page if not. The code is numbered from 1 to 9.

```
1 async function AdminVerification (){
2   const request = await fetch(
3     "../../php/admin/AdminVerify.php");
4   const Admin = await request.json();
5   console.log(Admin)
6   if (!Admin["admin"]){
7     window.location.href = "../../index/"+Admin["Link"];
8   }
9   adminConfig();
}
```

Dans le fichier `admin.js`, une fonction asynchrone est utilisée afin de vérifier les droits de l'utilisateur interagissant avec la page d'administration.

Cette fonction a pour but de contrôler si l'utilisateur possède le rôle d'administrateur.

Si ce n'est pas le cas, l'utilisateur est automatiquement redirigé vers la page principale du site, à savoir `index.html`.

Ce système permet de sécuriser l'accès à l'espace administrateur et d'éviter qu'un utilisateur non autorisé puisse consulter ou modifier les données réservées à la gestion du site.