

Rapport Site Web

Le site commence par une page de connexion invitant l'utilisateur à s'authentifier, si le pseudo entré n'est pas connu de la base de donnée, un compte est automatiquement créé avec le mot de passe donné.

Site Weeb

Pseudo :	<input type="text"/>
Mot de passe :	<input type="password"/>
<input type="button" value="Confirmer"/>	

Le header permet d'accéder aux 4 différentes pages du site. On peut voir au-dessus le nom de l'utilisateur ainsi qu'un texte se déconnecter qui permet de retourner sur la page de connexion. Le site dispose d'un contrôleur qui vérifie que l'utilisateur s'est bien connecté et le redirige vers la page de connexion si ce n'est pas le cas.

Bienvenue **lukas** [se déconnecter](#)

Site Weeb			
Accueil	Recherche Anime	Carte Studio	Profil

Bienvenue sur le site Weeb !!!

Le site propose une recherche d'anime en faisant une requête à une API et affiche le résultat se rapprochant le plus de ce qui a été donné. Après avoir affiché le résultat de la recherche, l'utilisateur peut définir l'anime qu'il a recherché comme son anime favori. Enfin, il peut aller sur son profil pour voir l'anime qu'il a défini comme favori.

Recherche d'anime

<input type="text"/>	<input type="button" value="rechercher"/>
----------------------	---

Recherche d'anime

yuru camp

Yuru Camp△



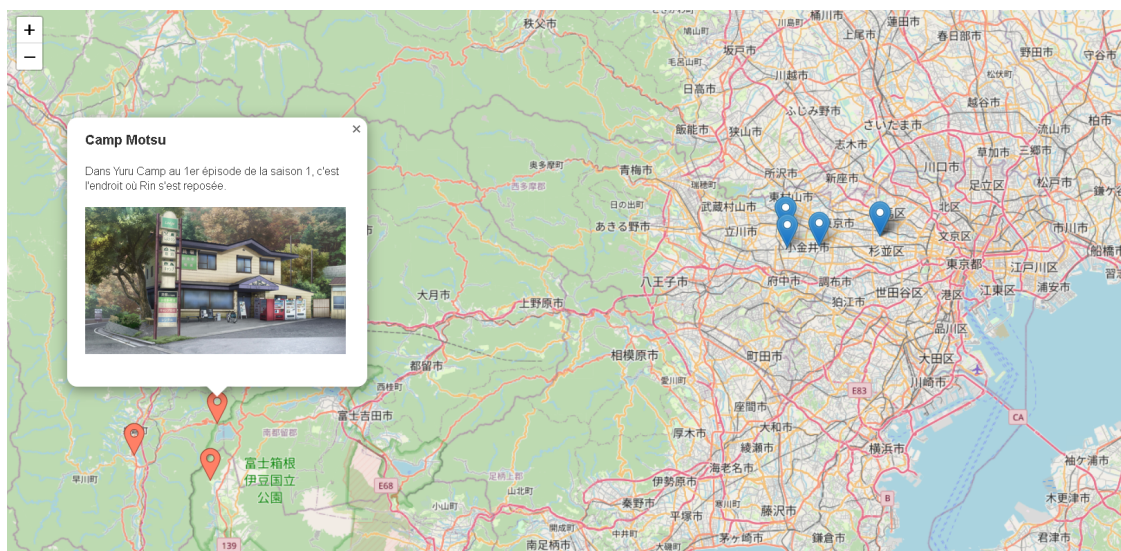
Profil de lukas

Anime favori :

Non Non Biyori



Le site offre aussi une carte possédant de nombreux repères, les marqueurs bleus représentent les studios animations dans la base de données, ceux oranges représentent des lieux utilisés dans l'anime Yuru Camp, tous les marqueurs sont cliquables pour obtenir plus d'information sur la représentation du lieu. Ceci représente un petit échantillon, la carte pourrait supporter bien plus d'informations.



Base de données :

La base de données du site possède 6 tables différentes.

Les tables studio et endroit correspondent aux différentes locations apparaissant dans la carte et contiennent pour chaque entrée le nom ainsi que les coordonnées GPS du point avec une description accompagné du chemin vers une image. La table utilisateur contient le pseudo avec le mot de passe hashé, de plus, pour la page profil, il contient une colonne pour stocker l'anime favori. Les tables anime, personnage, et appartenir serviront pour de future amélioration.

Table	Action	Lignes	Type	Interclassement	Taille	Perte
<input type="checkbox"/> anime	★ Afficher Structure Rechercher Insérer Vider Supprimer	1	InnoDB	armSCII8_general_ci	32 Kio	-
<input type="checkbox"/> appartenir	★ Afficher Structure Rechercher Insérer Vider Supprimer	1	InnoDB	armSCII8_general_ci	48 Kio	-
<input type="checkbox"/> endroit	★ Afficher Structure Rechercher Insérer Vider Supprimer	5	InnoDB	armSCII8_general_ci	16 Kio	-
<input type="checkbox"/> personnage	★ Afficher Structure Rechercher Insérer Vider Supprimer	1	InnoDB	armSCII8_general_ci	16 Kio	-
<input type="checkbox"/> studio	★ Afficher Structure Rechercher Insérer Vider Supprimer	5	InnoDB	armSCII8_general_ci	16 Kio	-
<input type="checkbox"/> utilisateur	★ Afficher Structure Rechercher Insérer Vider Supprimer	2	InnoDB	armSCII8_general_ci	16 Kio	-
6 tables	Somme	15	MyISAM	utf8mb4_general_ci	144 Kio	0 o

Requête Anilist API (GraphQL) :

Documentation : <https://anilist.gitbook.io/anilist-apiv2-docs/>

Cette API nous permet de récolter de nombreuses informations sur les studios d'animation, anime ainsi que personnages d'anime. Cependant la localisation des lieux ne se trouve pas dans l'API nous empêchant ainsi de l'utiliser pour y ajouter des points sur notre carte.

L'API nous sert pour la page de recherche, le nom qui rentré dans l'input sera recherché une fois l'utilisateur ayant validé son choix. La requête se trouve ci-dessous.

```
var query = `
  query($search : String){
    Media (search: $search, type : ANIME){
      id
      coverImage{
        large
      }
      title{
        romaji
      }
    }
  }
`;
```

Le bouton d'ajout en favori ajoute alors l'id de cette requête dans la BD dans la table de l'utilisateur. Ce bouton active la requête ajax suivante.

```
function favori(){
    let idAnime = $("#id").val();
    let data = {id : idAnime}
    $.ajax({
        //L'URL de la requête
        url: "putFavAnime.php",

        data: data,

        //La méthode d'envoi (type de requête)
        method: "POST"
    })
}
```

```
<?php
session_start();
$pseudo = isset($_SESSION['pseudo'])?($_SESSION['pseudo']):'';
$id = isset($_POST['id'])?($_POST['id']):'';

require_once('connectSQL.php');
$sql="UPDATE utilisateur SET IdAnimeFav = :id WHERE Identifiant = :pseudo";
$commande = $pdo->prepare($sql);
$commande->bindParam(':id', $id);
$commande->bindParam(':pseudo', $pseudo);
$commande->execute();
?>
```

Une dernière requête sur l'API se trouve dans le profil permettant alors de récupérer l'anime favori à l'aide de l'id qui est stocké. grâce à la requête graphql suivante.

```
var query = `
    query($id : Int){
        Media (id: $id, type : ANIME){
            id
            coverImage{
                large
            }
            title{
                romaji
            }
        }
    }
`;
```

Futures Améliorations :

Comme potentiel amélioration, il a été prévu d'implémenter de nouvelles fonctionnalités comme la recherche d'anime par studio et la recherche de personnage par anime grâce aux tables actuellement inutilisées. Il devrait aussi être possible dans le futur de pouvoir filtrer les points sur la carte Leaflet afin d'avoir uniquement les marqueurs qui intéressent les utilisateurs.