

**PARRA Laurie**

**Compte  
rendu  
Projet  
Framework Web  
2019-2020**

# SOMMAIRE

<b>1-Descriptif du projet</b>	
<b>3</b>	
<b>a Présentation du thème</b>	<b>3</b>
<b>b BackLog</b>	<b>3</b>
<b>c Planning</b>	<b>4</b>
<b>2-Outils et méthodologie utilisés</b>	<b>6</b>
<b>a Packages</b>	<b>6</b>
<b>b Images utilisées/Icônes/Polices</b>	<b>6</b>
<b>3-Présentation des résultats obtenus</b>	
<b>7</b>	
<b>4-Difficultés rencontrées</b>	<b>13</b>
<b>5-Conclusion</b>	<b>15</b>

# 1- Descriptif du projet

## a Présentation du thème

Pour ce projet, j'ai décidé de choisir une gestion de différentes salles de sports. Il s'agira alors d'un site de réservation de cours collectifs dans différentes salles de sports. J'aurais ainsi accès aux mêmes bases de données que lors du projet précédent.

## b BackLog

Pour ce projet, j'ai mis en place ce BackLog :

N°	Tâche	Priorité	Notes	Critères d'acceptation
1	En tant qu'utilisateur, je peux avoir accès à la page d'accueil et aux présentation des salles de sports	1	Pour les salles : Lieux, lien vers le planning des cours collectifs, les horaires. d'ouvertures Pour les cours, l'heure, le jours et le cours.	
2	En tant que nouvel utilisateur, je peux créer un compte avec un mot de passe sécurisé	1		
3	En tant qu'utilisateur je peux avoir accès à mon profil et voir mes informations de compte	2		
4	En tant	1	L'utilisateur doit	Les réservations

	qu'utilisateur, je peux réserver un cours dans une salle de sport		avoir un compte valide, et être connecté + Si l'utilisateur choisi 2 (ou plus ) cours, il faut qu'il soit à des horaires différents.	doivent apparaître dans les informations de compte + Si l'utilisateur choisi 2 cours à la même heure et le même jours, un message d'être doit apparaître et la réservations ne doit pas se faire.
5	En tant qu'utilisateur je peux annuler un cours précédemment ajouté.	2		
6	En tant qu'administrateur je peux ajouter une salle de sport, la supprimer, ajouter des cours ou en supprimer.	1	L'administrateur soit être connecté en tant qu'administrateur.	Une page de sécurité doit apparaître pour empêcher toutes utilisateurs non administrateur d'accéder à ces fonctionnalités.

## c Planning

*Le symbole « \* » signifiera des difficultés rencontrées expliquées plus bas dans la section « Difficultés rencontrées »*

Mise en ligne du sujet du projet : 22 janvier 2020

**Semaine 3:** 5-12 février

Création du projet sous Symfony, des premières structures de base de données et début des créations des Entités ne nécessitant pas de relations entre elles. Création des premiers visuels de page d'accueil, et des pages individuelles des salles de sports.

**Semaine 4 :** 12-19 février

Fin de création des entités ( par adéquation, des tables de la base de données ) avec leurs relations.

Entités du projets :

- *Cours* ( id et nom )
- *Clients* ( id, informations sur l'utilisateur ( nom, prénom, email etc..)
- *Salles* ( id, nom, description etc..)
- *Horaires* ( id, idSalle, en relation **ManyToOne** avec l'entité Salles, idCours, en relation **ManyToOne** avec l'entité Cours, jour, heure)
- Reservations(id, EmailClient, idHoraire )

Création des différents planning individuels à l'aide des données présentes dans la base de données.

Création du formulaire de création de compte, et installation de la Security + login.\*

Semaine 5 : 19-26 février

Mise en place de la partie administrateur ( backOffice ) avec le bundle Easy-admin ( /admin en ayant seulement le rôle d'administrateur )

Semaine 6: 26 février- 4 mars

Mise en place d'un formulaire de login/encodage de mot de passe avec encoder ( Security ). Début de mise en place des différents rôles et accès autorisés (ADMIN, USER etc ).\*

Création de la fonction de réservation de cours, impossibilité de réserver deux fois la réservation ou de réserver deux horaires qui aurait le même jour et même heure mais dans une différente salle de sport.

Création de l'API avec API Plateform ( /apip )\*

Semaine 7: 4 – 11mars

Création du style et du CSS du site entier.

Création d'un moyen de sauvegarde des Réservations pour un affichage définitif dans la page Profil.

## **2- Outils et méthodologie utilisés**

### **a Packages**

Pour ce projet, j'ai utilisé les packages suivants: Dev, Prod, Test, Http foundation, pour le formulaire, Security, Easy Admin et Api Platform.

### **b Images utilisées/Icônes/Polices**

- Les images utilisées pour ce projet viennent toutes du site Pixabay. Un site d'images gratuites, libre de droit.

<https://pixabay.com/fr/>

- Pour certains textes, j'ai utilisé une police différente provenant de l'outil de Google : Google font, un outils libre de droit pour projet commercial et non-commercial.

<https://developers.google.com/fonts>

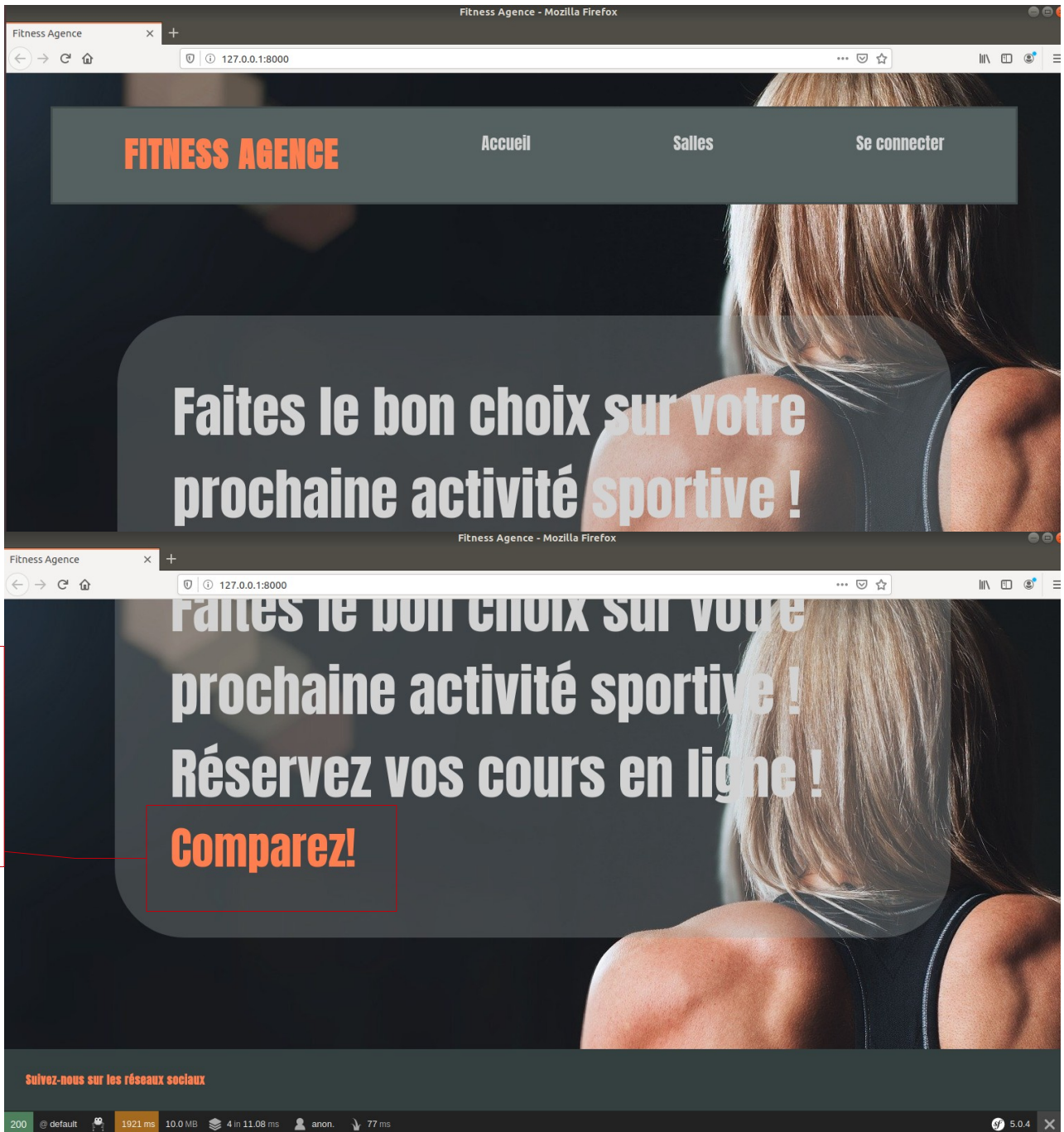
- Pour les icônes présentes sur le site, j'ai utilisé le site gratuit et libre de droit Iconmonstr

<https://iconmonstr.com/about/#q&a>

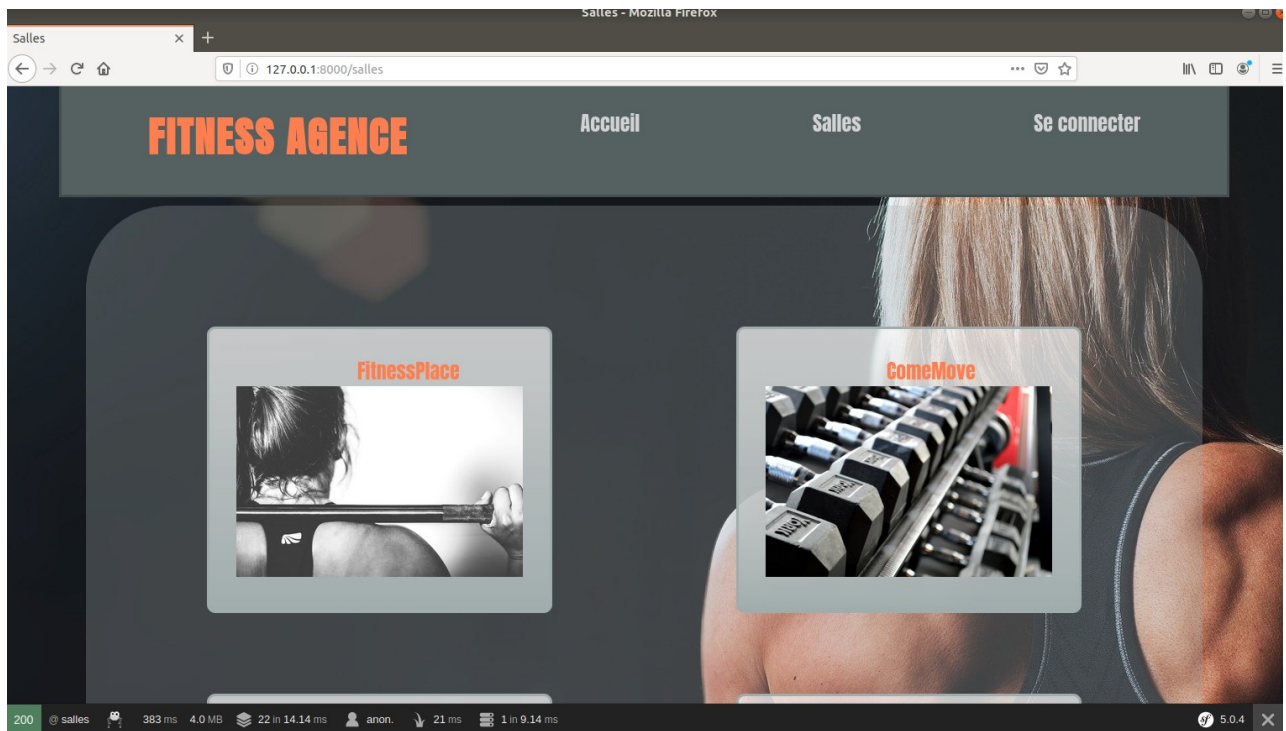
### 3- Présentation des résultats obtenus

Pour ce premier rendu de projet, le rendu final donne ceci en présentation générale :

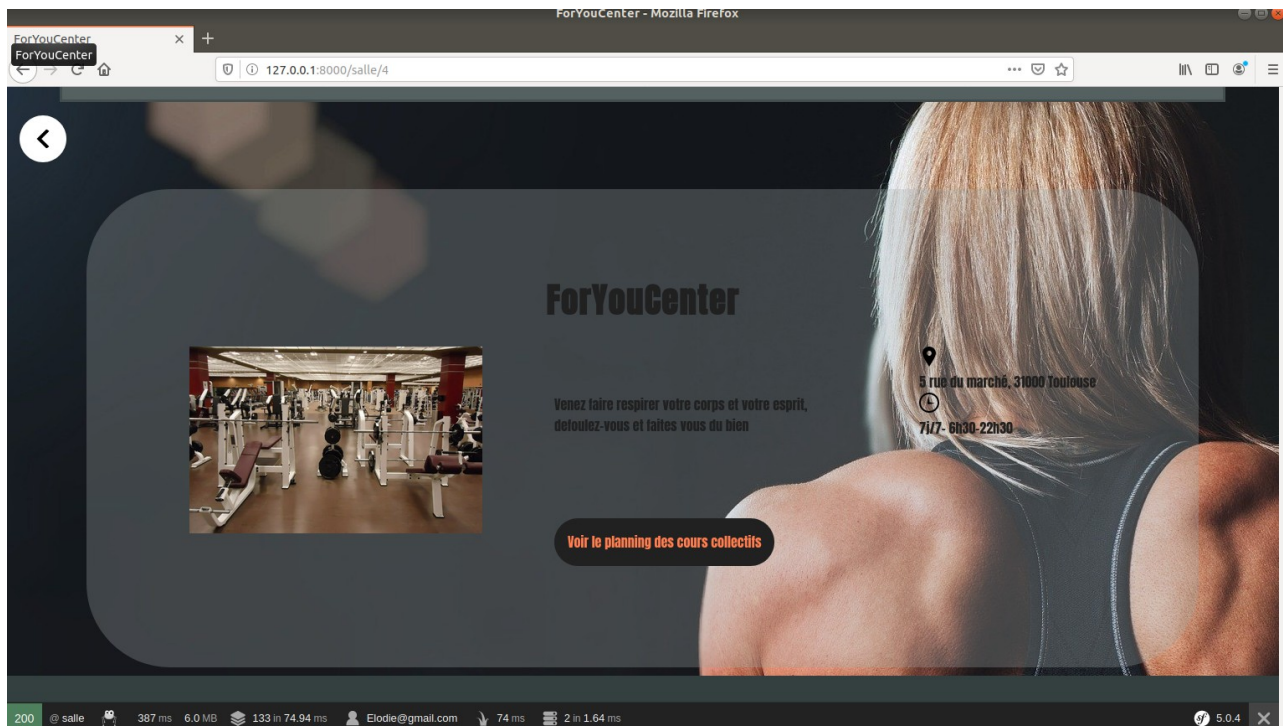
Pour l'accueil :



La page de choix des salles :

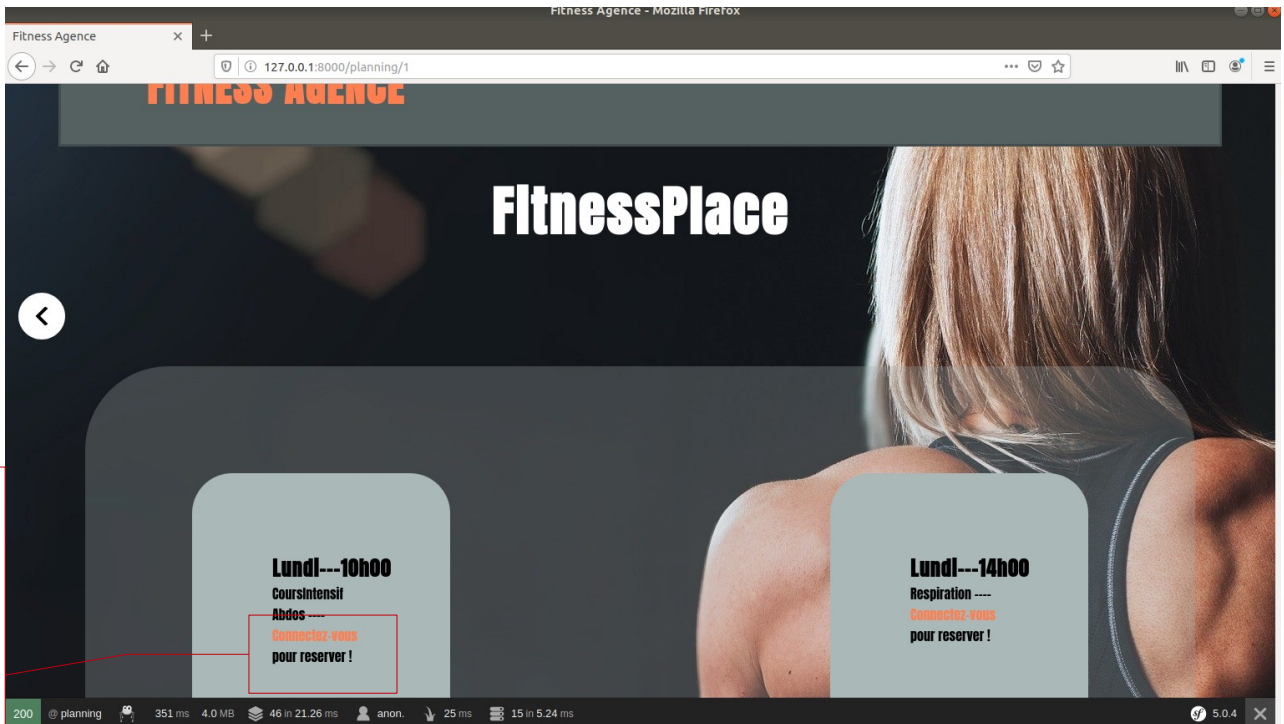


La présentation d'une salle de sport en cliquant sur un des cadres :



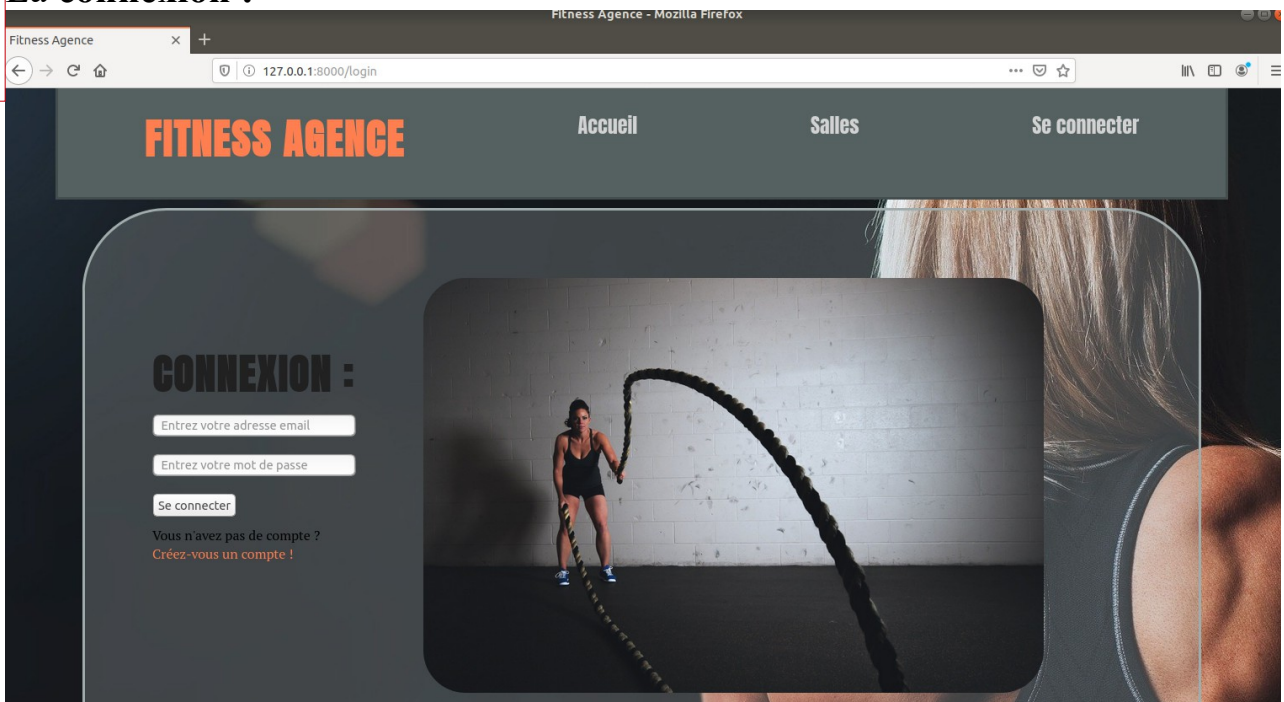


## Le planning d'une salle de sport :

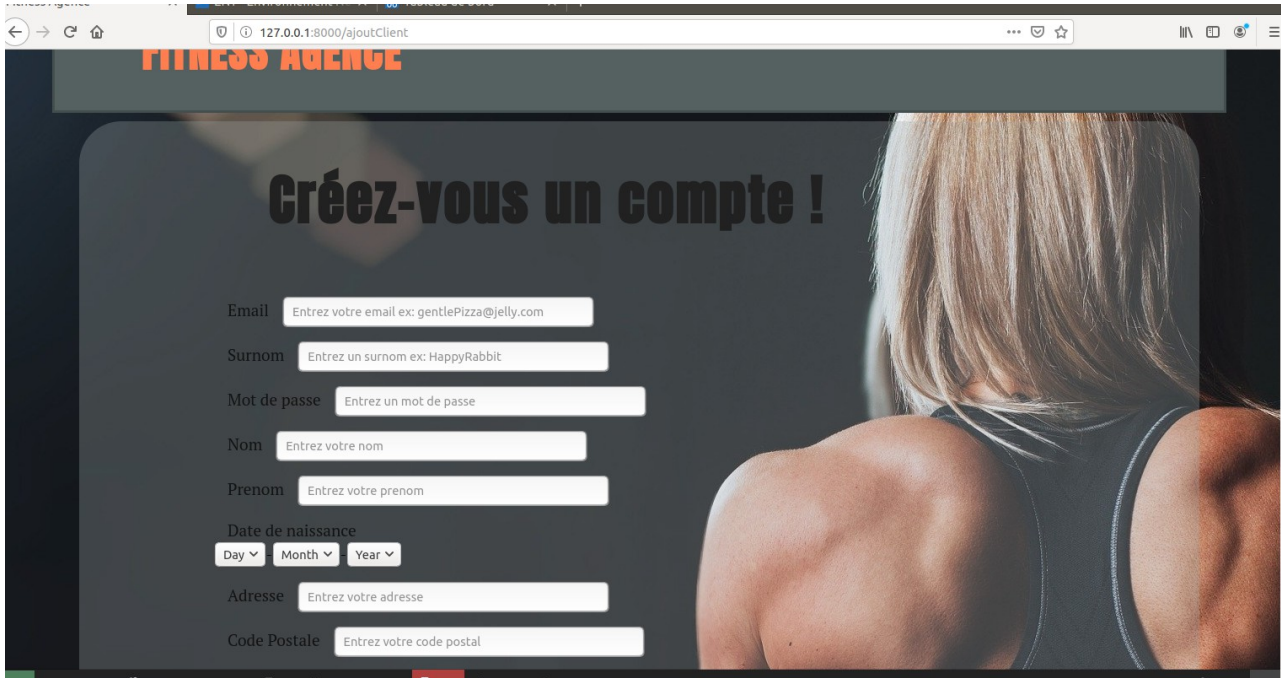


*Le message de demande de connexion apparaît à la place du bouton réserver si la personne n'est pas connectée*  
*Le lien « connectez-vous » pointe vers la page de connexion*

## La connexion :



## La création de compte :



**Créez-vous un compte !**

Email

Surnom

Mot de passe

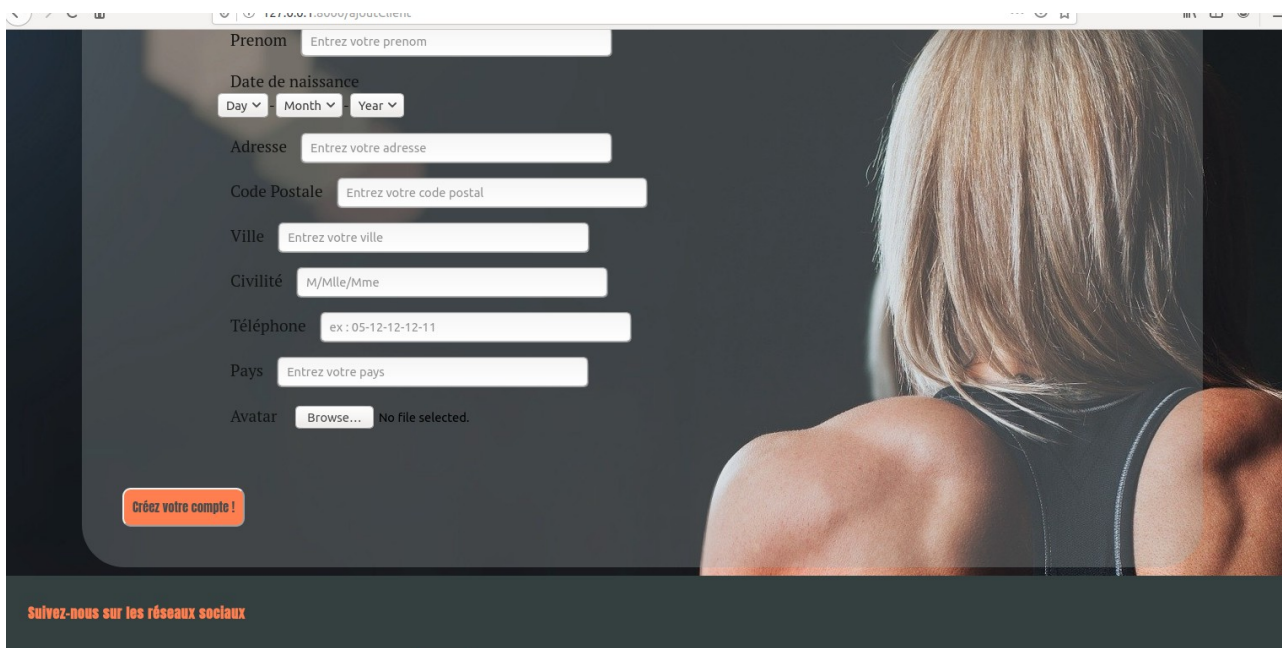
Nom

Prenom

Date de naissance  
Day  Month  Year

Adresse

Code Postale



Prenom

Date de naissance  
Day  Month  Year

Adresse

Code Postale

Ville

Civilité

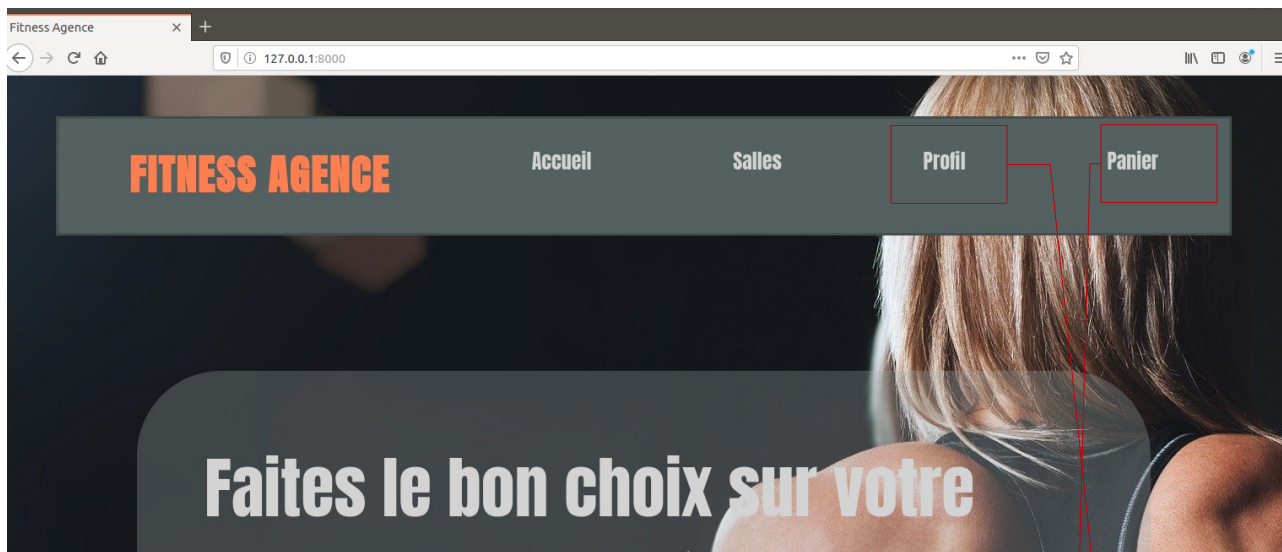
Téléphone

Pays

Avatar  No file selected.

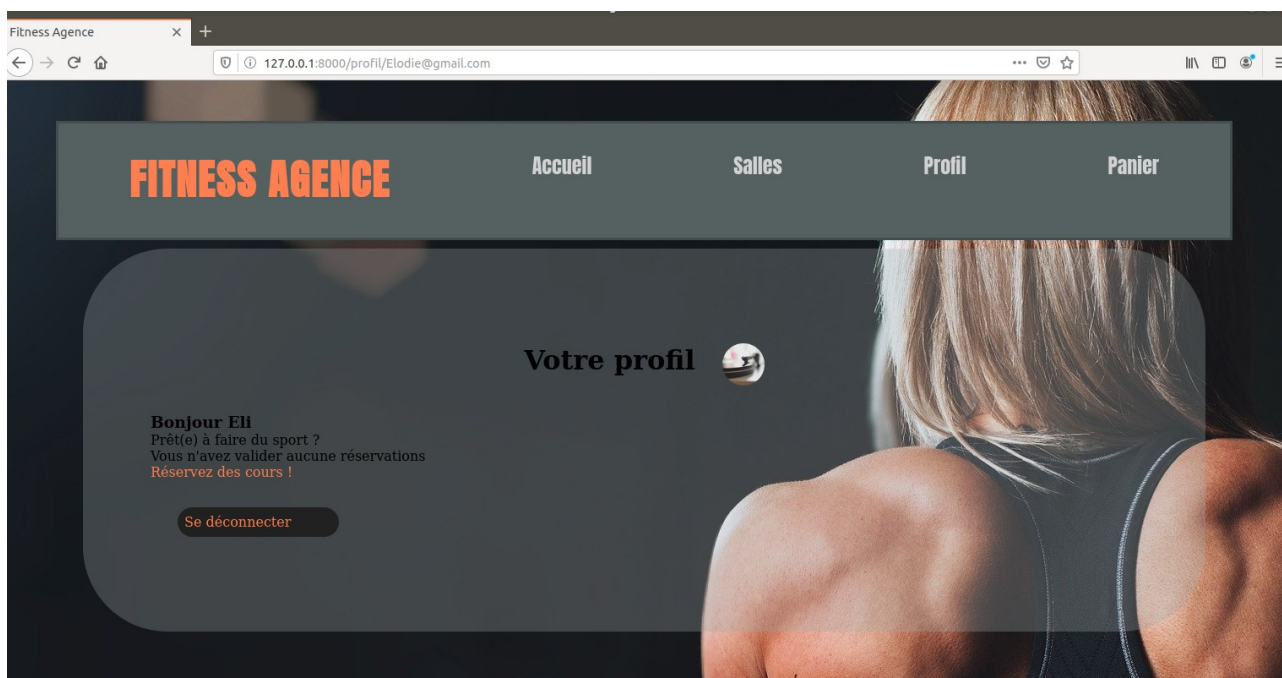
Suivez-nous sur les réseaux sociaux

L'apparence du site quand la personne est connectée : (ici pour l'exemple je me suis connectée en temps que : email : [Elodie@gmail.com](mailto:Elodie@gmail.com) mdp:jesuisadminaussi )



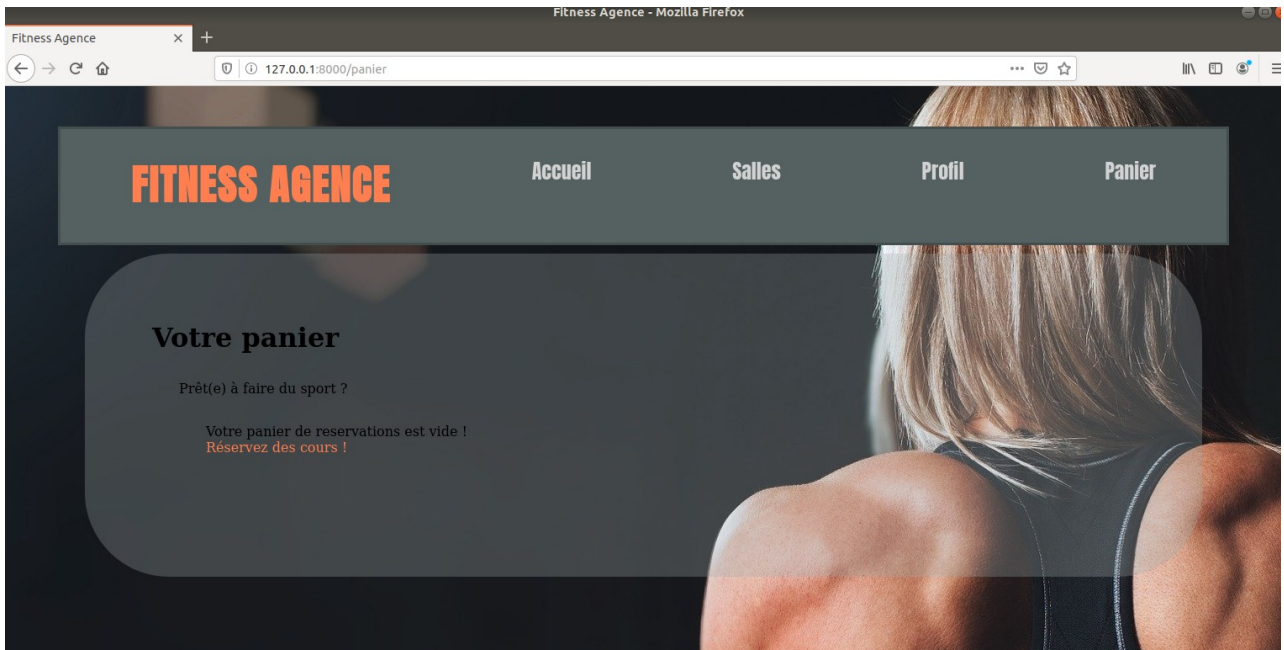
*Ces deux nouveaux menus apparaissent*

**Le profil (vide ici ):**





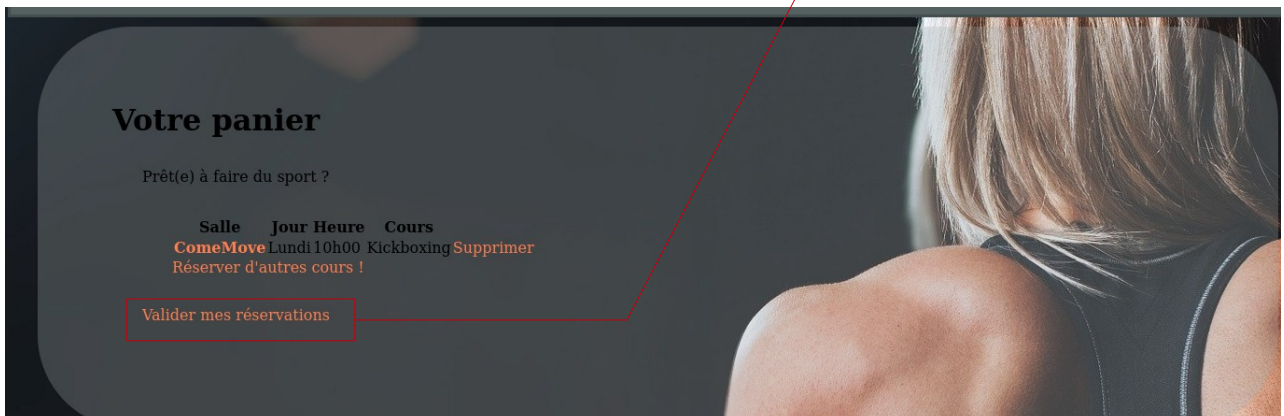
## Le panier (vide ici) :



Le panier est un endroit où, temporairement, les réservations se stockent. Il se vide à chaque déconnexion et quand la personne « Valide ses réservations » :

Quand le client connecté choisit différents horaires dans les planning et clique sur « réserver » les informations de l'horaire viennent se stocker ici ( nom de la salle en un lien renvoyant vers la description de la salle en question, le jour, l'heure et le nom du cours ). Cependant, le client arrivera sur une page d'erreur lui indiquant son problème s'il essaye de réserver deux fois la même heure ou une heure avec le même jour et la même heure qu'une autre réservation dans une différente salle de sport, pour éviter les réservations où la personne ne pourrait pas s'y rendre.

Lorsque la personne est satisfaite de ses réservations, elle peut les « valider », à l'aide du bouton adéquate qui apparaît lorsqu'il y a au minimum 1 réservation. En cliquant, les réservations vont s'enregistrer dans la base de données. C'est de ces données dont la page Profil se sert pour afficher définitivement les réservations.



### **Le BackOffice :**

L'espace administrateur n'est disponible que pour les personnes ayant « ROLE\_ADMIN » dans leurs attributs Rôles il est disponible avec l'adresse « /admin ».  
( Voir onglet Difficultés Rencontrées )

### **L'Api :**

L'api est disponible avec l'adresse « /apip ». ( Voir onglet Difficultés Rencontrées )

## **4- Difficultés rencontrées**

### **Le packages Security : ( Difficulté résolue )**

Je personnellement eu du mal à prendre en main le très complet bundle Security même en suivant scrupuleusement les indications de la documentation officielle de Symfony.

J'ai donc mis beaucoup de temps là dessus, cependant il ne peut pas être considéré comme du temps « perdu » car cela m'a permis de savoir m'en servir comme je le désirais.

### **L'API Platform :**

L'api que j'ai mis en place avec Api Platform fonctionne très bien pour afficher les données de ma base de données, pour ajouter une entité, elle fonctionne également, mais un problème se pose lors de la création d'une entité de la table Clients.

### **Administration :**

*Encodage du mot de passe lors de la création via le backOffice*

J'ai essayé de gérer l'encodage du mot de passe lors d'une modification ou de la création d'un client dans le backOffice, mais, impossibilité de la faire fonctionner normalement.

*Le Path\_avatar*

C'est une variable essentielle au fonctionnement dont je me suis servi pour enregistrer dans le dossier des images les images choisies dans le formulaire de création de compte comme avatar. Elle vaut à la fin le surnom de la personne suivis de « .jpg » pour pouvoir la retrouver dans le dossier en question. Elle se sert de l'attribut « avatar » qui lui retourne un type file

et qui permet de choisir une image dans le formulaire. Cependant, si on essaye de créer un client directement via le backOffice, il faudrait que l'image se trouve déjà dans le dossier de destination car la manipulation pour « l'upload » de l'image ne s'active que lors de l'enregistrement du client par le formulaire ( La manipulation: quand l'entité créée avec le formulaire est « Persist », les fonctions pour « l'upload » de l'image dans le bon dossier s'active et l'image est déplacée puis renommée, ces fonctions se trouvent dans l'entité Clients ).

### **Le panier :**

Le panier que j'avais créé ne se sauvegardait pas lors de la déconnexion, or, ici, il s'agissait plutôt de conserver les réservations enregistrées.

Solution trouvée :

Créer une nouvelle table Réservations qui permettrait que lorsque la personne valide ses horaires choisies, ces dernières se stockent. Ainsi, elles seraient en mémoire et facilement retrouvables pour un affichage définitif dans l'onglet Profil.

## 5- Conclusion

### Premier rendu du projet de FrameWork WEB

Avec ce premier rendu, j'ai rencontré plusieurs difficultés et je n'ai pas réussi à toutes les résoudre. Cependant, cela m'a permis de me rendre compte de la rapidité avec laquelle il est facile de produire avec Symfony et combien d'outils facilitant le codage il offre.