

## Modèle de copie : Evaluation en cours de formation

### Développeur Web et Web Mobile

Ceci est un modèle de copie. N'oubliez pas de renseigner vos prénom/nom, ainsi que le nom et le lien vers le projet.

Vous pouvez bien sûr agrandir les cadres pour répondre aux questions sur la description du projet si nécessaire.

**Prénom : MAZOUNI**

**Nom : Mehdi**

**ATTENTION ! PENSEZ À RENSEIGNER VOS NOM ET PRÉNOM DANS LE TITRE DE VOS FICHIERS / PROJETS !**

Nom du projet : Body Top

Lien Github du projet : <https://github.com/Mazouni-Mehdi/Body-Top.git>

Lien Drive du projet (si nécessaire) : .....

**Attention ! Merci de bien classer vos documents dans votre Github ou votre drive.**

URL du site (si vous avez mis votre projet en ligne) : <https://php-body-top.herokuapp.com/>

<https://trello.com/invite/b/UJYtNCZC/ATTI4eea6c4fb3902f0e8e27bd0ba74cea6571D51CC7/ecf-decembre-2022>

Compte Administrateur

[BodyTopEquipeTech@gmail.com](mailto:BodyTopEquipeTech@gmail.com) Admin.123

## Description du projet

1. Liste des compétences du référentiel qui sont couvertes par le projet

### **a. Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité**

- Maquetter une application
- Réaliser une interface utilisateur web statique et adaptable
- Développer une interface utilisateur web dynamique
- Réaliser une interface utilisateur avec une solution de gestion de contenu ou e-commerce

### **b. Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité**

- Créer une base de données
- Développer les composants d'accès aux données
- Développer la partie back-end d'une application web ou web mobile
- Elaborer et mettre en œuvre des composants dans une application de gestion de contenu ou e-commerce

2. Résumé du projet en français d'une longueur d'environ 20 lignes soit 200 à 250 mots, ou environ 1200 caractères espaces non compris

Le projet a pour objectif de développer une application de droits d'accès destinées à l'équipe de développement marketing d'une grande marque de salle de sport : BodyTop.

BodyTop souhaite la création d'une interface simple et intuitive sous forme de modules souscrits par contrat par les différents franchisés : « gérer le planning d'équipe », « faire sa publicité », vendre des boissons »...

Du coté administration, l'application dispose d'un back office uniquement accessible à l'équipe de développement marketing de BodyTop. Celle-ci indiquera la liste des franchisés et leurs structures avec les différents modules souscrits activés et non souscrits désactivés. Un changement sera possible selon l'évolution du contrat. Une barre de recherche permettra à l'utilisateur de trouver rapidement une structure ou un franchisé.

De même, on doit pouvoir trier la liste des franchisés ou des structures afin de ne présenter que les éléments actifs ou désactivés.

Les franchisés et leurs structures se verront attribuer un compte utilisateur créé par l'équipe de développement marketing de BodyTop . Ce compte sera accessible en lecture seule, pour visualiser les modules activés ou non.

Bien que ce projet soit une simulation, les pratiques de sécurité devront être respectées et mises en place pour garantir un site de qualité respectant les mêmes procédures qu'en environnement réel

3. Cahier des charges, expression des besoins, ou spécifications fonctionnelles du projet

Dans le cadre de mon ECF chez studi (contexte de travail°

Explication de l'application les besoins de la marque

Explication des fonctionnalité

4. Spécifications techniques du projet, élaborées par le candidat, y compris pour la sécurité et le web mobile

## Spécifications techniques

### Mes outils de développement

- Système de Contrôle de Version : Git / GitHub
- Éditeur de texte : Visual Studio Code
- Serveur MySQL : 10.4.24-MariaDB
- Terminal de commande : Cmdr
- Création de schéma de données : Excalidraw
- Création de maquettes : Excalidraw
- Organisation : Trello
- Navigateurs : Google Chrome, Mozilla Firefox
- Gestionnaire de Dépendances PHP : Composer
- Commande Symfony : Symfony CLI

## Langages / Frameworks

### Langages

- HTML 5
- Twig
- CSS 3
- SCSS
- PHP 8.1.4

### Frameworks

- Symfony 6
- Bootstrap 5

## Mesures de sécurité

L'utilisation de Symfony et plus particulièrement du « bundle Security » nous apporte une grande panoplie d'outils pour gérer et renforcer la sécurité de l'application.

### L'authentification

- Renforcement de la politique de sécurité lors de la création de mot de passe par l'utilisateur.
- Les mots de passes sont chiffrés en Base de données.

### Les autorisations

- Gestion des droits utilisateurs, et restriction des accès à certaines pages de l'application.

### Protection contre les injections

- Validation obligatoire des données entrées par l'utilisateur, grâce au système de « contraintes » de Symfony
- L'utilisation de l'ORM Doctrine pour gérer les interactions avec la base de données protège l'application des injections SQL, grâce au système de requêtes préparées.
- L'utilisation du moteur de rendu TWIG protège l'application contre le « Cross site scripting » (XSS), grâce à sa syntaxe entre double accolades, qui permet l'échappement des données.

### Les formulaires

- Symfony ajoute une protection contre le « Cross site request forgery » (CSRF), en

## Le protocole Https

- Le service HTTPS activé vient renforcer la sécurité des échanges.

## Frameworks utilisés

### Symfony 6

J'ai choisi symfony car il apporte des outils et des services qui facilitent le développement d'applications PHP, il apporte également une grande flexibilité, une personnalisation, une fiabilité éprouvée et surtout il intègre des mesures de sécurité préventives pour lutter contre les failles et attaques XSS, CSRF et injection SQL .C'est aussi un framework très demandé, qui apparaît souvent dans les offres d'emploi.

### Bootstrap 5

Le choix de Bootstrap s'est imposé à moi car il est beaucoup demandé.

## Création du template

### Structure du template

Le template "base.html.twig" se trouve à la racine du dossier "templates", là où sont rangées toutes mes vues.

Il contient tout le Doctype HTML :

- La balise "head" avec ses balises "meta", "title", "link" et "script"
- La balise "body" qui inclut :
  - Un bloc "nav.html.twig" (templates/\_partials/\_nav.html.twig)
  - Un bloc "content"
  - Un bloc "footer.html.twig" (templates/\_partials/\_footer.html.twig)

J'ai créé un dossier "base" qui contient les éléments de base : menu de navigation et footer.

Les différentes pages du site se retrouveront dans le bloc "body"

Aperçu du fichier "base.html.twig"

```
es > base.html.twig
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>
      {% block title %}Body Top
      {% endblock %}
    </title>
    <link
      rel="icon" href="data:image/svg+xml,<svg xmlns=%22http://www.w3.org/2000/svg%22 viewBox=%220 0 128 128%22>
      <text y=%221.2em%22 font-size=%2296%22>●</text></svg>">
    </link>

    <!-- Feuilles de styles -->
    <link rel="stylesheet" href="{{ asset('asset/css/bootstrap.min.css') }}">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.2.0/css/all.min.css"
      integrity="sha512-xh60/CkQoPOWdDdYTDqerRdPCVd1SpvCA9XXcUnZS2FmJNp1coAFzvtCN9BmamE+4aHK8yyUHUSCcJHgXloTyT2A=="
      crossorigin="anonymous" referrerpolicy="no-referrer"/>
    <link rel="stylesheet" href="{{ asset('asset/css/styles.css') }}">
    {% block stylesheets %}{% endblock %}

  </head>
  <body>
    {% include "_partials/_nav.html.twig" %}

    {% block body %}{% endblock %}

    {% include "_partials/_footer.html.twig" %}

    <!-- Fichiers JS -->
    <script src="{{ asset('asset/js/bootstrap.bundle.min.js') }}" defer></script>
    <script src="{{ asset('asset/js/scripts.js') }}" defer></script>
    {% block scripts %}
    {% endblock %}
  </body>
</html>
```

## Aperçu du fichier “\_nav.html.twig”

```
es > _partials > _nav.html.twig
<nav>
  <a href="/" class="nav-icon" aria-label="visit homepage" aria-current="page">
    
    <span>Body Top</span>
  </a>
  <div class="main-navlinks">
    <button class="hamburger" type="button" aria-label="Toggle navigation" aria-expanded="false">
      <span></span>
      <span></span>
      <span></span>
    </button>
    <div class="navlinks-container">
      {% if is_granted('ROLE_ADMIN') %}
        <a href="/admin/franchise/new">Créer franchise</a>
        <a href="/admin/structure/new">Créer Structure</a>
        <a href="/admin">Voir franchise</a>
        <a href="/admin/structure">Voir Structure</a>
      {% endif %}
    </div>
  </div>
  <div class="nav-authentication">
    <a href="{% if app.user %}/logout{% else %}/login{% endif %}" class="sign-user" aria-label="Sign in page">
      <i class="fa-solid fa-user"></i>
    </a>
    <div class="sign-btns">
      {% if app.user %}
        <a href="/logout" class="btn">Se déconnecter</a>
      {% else %}
        <a href="/login" class="btn">Se connecter</a>
      {% endif %}
    </div>
  </div>
</nav>
```



## Responsive de la barre de navigation

j'ai créé la NavBar, avec la partie responsive également.

J'ai utilisé media queries pour faire apparaître un menu hamburger en dessus de 1000px

Et en faisant évoluer les boutons connexion et déconnexion en Icon

Aperçu d'une partie du fichier "styles.css"

```
asset > css > styles.css > {} @media (max-width: 1000px)
@media (max-width: 1000px) {
  nav {
    padding: 15px 20px;
    position: relative;
  }

  .nav-icon {
    order: 2;
    margin: 0 auto;
  }

  .nav-icon span {
    font-size: 22px;
  }

  /* Hamburger*/
  .hamburger {
    width: 20px;
    height: 20px;
    cursor: pointer;
    border: none;
    display: flex;
    background: #F9CB40;
    align-items: center;
    position: relative;
  }

  .hamburger span {
    display: block;
    width: 100%;
    height: 2px;
    background: #333;
    position: absolute;
    pointer-events: none;
    transition: opacity 0.3s 0.15s ease-out;
  }

  .hamburger span:nth-child(1),
  .hamburger span:nth-child(3) {
    transition: transform 0.3s ease-out;
  }

  .hamburger span:nth-child(1) {
    transform: translateY(7px);
  }

  .hamburger span:nth-child(3) {
    transform: translateY(-7px);
  }

  .hamburger.open span:nth-child(1) {
    transform: translate(0) rotate(135deg);
  }

  .hamburger.open span:nth-child(2) {
    opacity: 0;
    transition: opacity 0s ease-out;
  }

  .hamburger.open span:nth-child(3) {
    transform: translate(0) rotate(-135deg);
  }

  .navlinks-container {
    display: flex;
    flex-direction: column;
    align-items: flex-start;
    position: absolute;
    background: #fafafa;
    top: 100%;
    left: 0;
    height: 100vh;
    padding: 15px 50px 15px 20px;
    border-right: 1px solid #c5c5c6;
    transform: translate(-100%);
  }
}
```

Aperçu du fichier “\_footer.html.twig”

```
tes > _partials > _footer.html.twig
<footer>
  <div class="footer">
    <div class="footer-content">
      <h3>Body Top</h3>
      <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit.</p>
      <p>Perferendis dolor ab illo iure harum </p>
    </div>
    <div class="footer-bottom">
      <p>copyright 2022</p>
    </div>
  </div>
</footer>
```

## Création, récupération et affichage des données

### Création de la base de données

Le fichier “.env” à la racine de mon projet me sert, entre autres, à créer une base de données. Je lui indique que j’utilise du MySQL, l’identifiant, le port où se trouve MySQL et le nom de la base de données, ainsi que la version du serveur

```
# DATABASE_URL="sqlite:///kernel.project_dir%/var/data.db"
DATABASE_URL="mysql://root:@localhost/BodyTop?serverVersion=10.4.24-MariaDB"
# DATABASE_URL="postgresql://app:!ChangeMe!@127.0.0.1:5432/app?serverVersion=14&charset=utf8"
###< doctrine/doctrine-bundle ###
```

symfony console doctrine:database:create

## Création des entités “User, Franchise, Structure, Module”

À l’aide de la commande Symfony CLI, je créer les entités

`symfony console make:entity`

Les fichiers “User.php, Franchise.php, Structure.php, Module.php ” sont créé dans le dossier “Entity” du dossier “src”.

J’ajoute certaines contraintes #[UniqueEntity]

Et j’ai créé certains constructeur pour les relations entre certaines entités

Je fais ensuite une migration, que j’envoie à la base de données

`symfony console make:migration`

`symfony console doctrine:migrations:migrate`

Aperçu du fichier de la migration

```

ions > Version20221030160304.php > ...
/*
 * This file is part of the Symfony package.
 *
 * (c) Fabien Potencier <fabien@symfony.com>
 *
 * For the full copyright and license information, please view the LICENSE
 * file that was distributed with this source code.
 */
final class Version20221030160304 extends AbstractMigration
{
    public function getDescription(): string
    {
        return '';
    }

    public function up(Schema $schema): void
    {
        // this up() migration is auto-generated, please modify it to your needs
        $this->addSql('CREATE TABLE franchise (id INT AUTO_INCREMENT NOT NULL, name VARCHAR(50) NOT NULL, is_active TINYINT(1) NOT NULL, UNIQUE INDEX UNIQ_66F6CE2A5E237E06 (name), PRIMARY KEY (id))');
        $this->addSql('CREATE TABLE module (id INT AUTO_INCREMENT NOT NULL, planning TINYINT(1) NOT NULL, registration TINYINT(1) NOT NULL, negotiation TINYINT(1) NOT NULL, sale TINYINT(1) NOT NULL, PRIMARY KEY (id))');
        $this->addSql('CREATE TABLE structure (id INT AUTO_INCREMENT NOT NULL, franchise_id INT NOT NULL, module_id INT NOT NULL, address VARCHAR(255) NOT NULL, zipcode VARCHAR(10) NOT NULL, PRIMARY KEY (id), INDEX IDX_66F6CE2A5E237E06 (franchise_id), INDEX IDX_66F6CE2A5E237E06 (module_id))');
        $this->addSql('CREATE TABLE user (id INT AUTO_INCREMENT NOT NULL, franchise_id INT DEFAULT NULL, structure_id INT DEFAULT NULL, email VARCHAR(180) NOT NULL, roles LONGTEXT NOT NULL, PRIMARY KEY (id), INDEX IDX_66F6CE2A5E237E06 (franchise_id), INDEX IDX_66F6CE2A5E237E06 (structure_id))');
        $this->addSql('CREATE TABLE messenger_messages (id BIGINT AUTO_INCREMENT NOT NULL, body LONGTEXT NOT NULL, headers LONGTEXT NOT NULL, queue_name VARCHAR(190) NOT NULL, PRIMARY KEY (id))');
        $this->addSql('ALTER TABLE structure ADD CONSTRAINT FK_6F0137EA523CAB89 FOREIGN KEY (franchise_id) REFERENCES franchise (id)');
        $this->addSql('ALTER TABLE structure ADD CONSTRAINT FK_6F0137EA523CAB89 FOREIGN KEY (module_id) REFERENCES module (id)');
        $this->addSql('ALTER TABLE user ADD CONSTRAINT FK_8D93D649523CAB89 FOREIGN KEY (franchise_id) REFERENCES franchise (id)');
        $this->addSql('ALTER TABLE user ADD CONSTRAINT FK_8D93D649523CAB89 FOREIGN KEY (structure_id) REFERENCES structure (id)');
    }

    public function down(Schema $schema): void
    {
        // this down() migration is auto-generated, please modify it to your needs
        $this->addSql('ALTER TABLE structure DROP FOREIGN KEY FK_6F0137EA523CAB89');
        $this->addSql('ALTER TABLE structure DROP FOREIGN KEY FK_6F0137EA523CAB89');
        $this->addSql('ALTER TABLE user DROP FOREIGN KEY FK_8D93D649523CAB89');
        $this->addSql('ALTER TABLE user DROP FOREIGN KEY FK_8D93D649523CAB89');
        $this->addSql('DROP TABLE franchise');
        $this->addSql('DROP TABLE module');
        $this->addSql('DROP TABLE structure');
        $this->addSql('DROP TABLE user');
        $this->addSql('DROP TABLE messenger_messages');
    }
}

```

Aperçu de la table “Structure” dans phpmyadmin et du fichier “Structure.php”

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
<input type="checkbox"/>	1 id	int(11)			Non	Aucun(e)		AUTO_INCREMENT	Modifier  Supprimer  Plus
<input type="checkbox"/>	2 franchise_id	int(11)			Non	Aucun(e)			Modifier  Supprimer  Plus
<input type="checkbox"/>	3 module_id	int(11)			Non	Aucun(e)			Modifier  Supprimer  Plus
<input type="checkbox"/>	4 address	varchar(255)	utf8mb4_unicode_ci		Non	Aucun(e)			Modifier  Supprimer  Plus
<input type="checkbox"/>	5 zipcode	varchar(5)	utf8mb4_unicode_ci		Non	Aucun(e)			Modifier  Supprimer  Plus
<input type="checkbox"/>	6 city	varchar(150)	utf8mb4_unicode_ci		Non	Aucun(e)			Modifier  Supprimer  Plus
<input type="checkbox"/>	7 is_active	tinyint(1)			Non	Aucun(e)			Modifier  Supprimer  Plus

```
Entity > Structure.php > PHP Intelephense > Structure
<?php

namespace App\Entity;

use App\Repository\StructureRepository;
use Doctrine\ORM\Mapping as ORM;

#[ORM\Entity(repositoryClass: StructureRepository::class)]
class Structure
{
    #[ORM\Id]
    #[ORM\GeneratedValue]
    #[ORM\Column]
    private ?int $id = null;

    #[ORM\Column(length: 255)]
    private ?string $address = null;

    #[ORM\Column(length: 5)]
    private ?string $zipcode = null;

    #[ORM\Column(length: 150)]
    private ?string $city = null;

    #[ORM\Column]
    private ?bool $is_active = null;

    #[ORM\OneToOne(mappedBy: 'structure', cascade: ['persist', 'remove'])]
    private ?User $user = null;

    #[ORM\ManyToOne(inversedBy: 'structure')]
    #[ORM\JoinColumn(nullable: false)]
    private ?Franchise $franchise = null;

    #[ORM\OneToOne(cascade: ['persist', 'remove'])]
    #[ORM\JoinColumn(nullable: false)]
    private ?Module $module = null;

    public function getId(): ?int
    {
        return $this->id;
    }

    public function getAddress(): ?string
    {
        return $this->address;
    }
}
```

## Ajout de données avec les Fixtures

Installation du composant orm-fixtures

**composer require orm-fixtures**

Le fichier “AppFixtures.php” est créé dans le dossier “DataFixtures”

Ce fichier contient une méthode “load”, qui va me permettre d’enregistrer des données dans la base.

- Je crée mon administrateur
- Je définis les différents champs
- Je “persiste” ces données (équivalent du “commit” de Git)
- J’envoie ces données à la base avec un flush (équivalent du “push” de Git)

Aperçu du fichier “AppFixtures.php”

```
<?php

namespace App\DataFixtures;

use App\Entity\User;
use Doctrine\Bundle\FixturesBundle\Fixture;
use Doctrine\Persistence\ObjectManager;
use Symfony\Component\PasswordHasher\Hasher\UserPasswordHasherInterface;

class AppFixtures extends Fixture
{
    public function __construct(UserPasswordHasherInterface $passwordHasher)
    {
        $this->passwordHasher = $passwordHasher;
    }

    public function load(ObjectManager $manager): void
    {
        $admin = new User($this->passwordHasher);
        $admin->setEmail("BodyTopEquipeTech@gmail.com")->setPassword("Admin.123")->setRoles(["ROLE_ADMIN"]);

        $manager->persist($admin);

        $manager->flush();
    }
}
```

## Création de la page d'accueil & des différentes pages

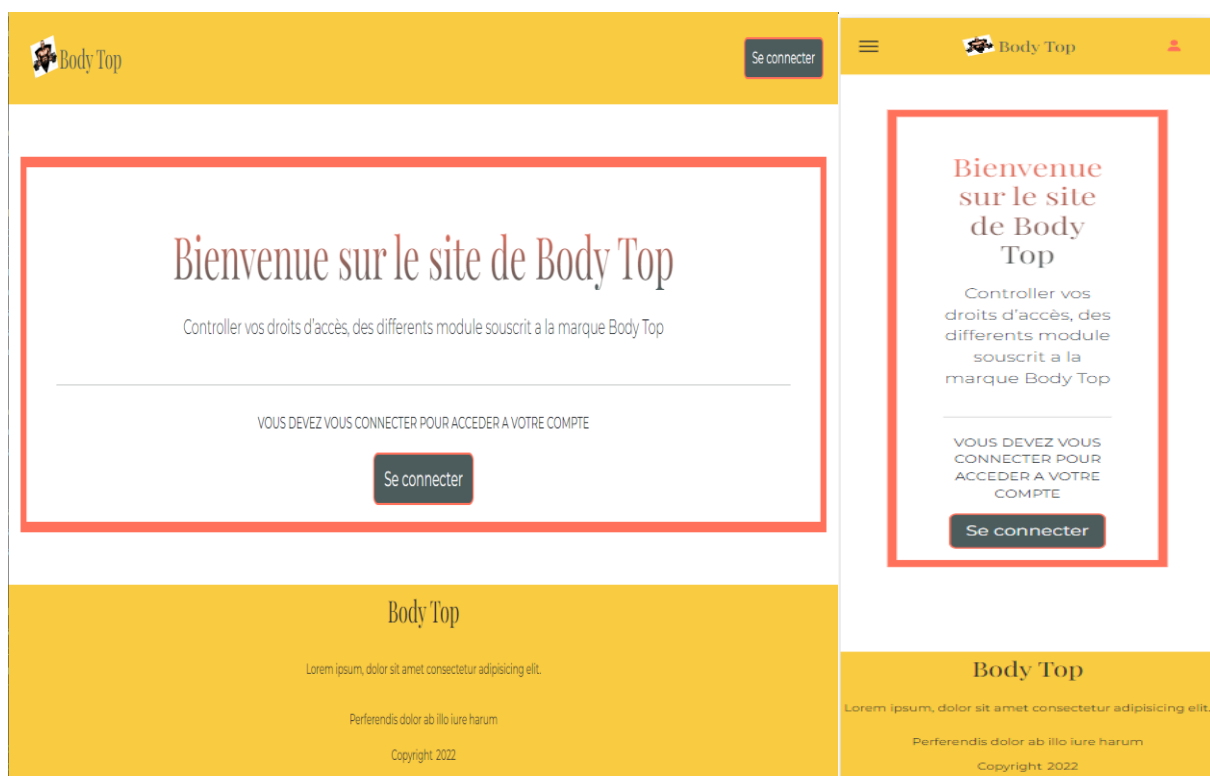
Je créer les différents contrôleur associer aux entités, que je nomme par exemple "HomeController" à l'aide de la commande :

```
symfony console make:controller HomeController
```

Les controllers se trouve dans le dossier "Controller" du dossier "src", je spécifie les routes a chacun dans les annotations de sa fonction "index" ou autre, pour qu'il "s'active" quand la racine du site est appelée

dans l'adresse URL.

Aperçu de la page d'accueil en vue ordinateur et mobile



Après la connexion selon le rôle de utilisateur vous êtes redigé vers des endroit différent du site

Pour l'administrateur vous êtes redigé vers la partie Admin de l'application  
l'authentification grâce au fichier : "UserAuthenticator.php"

Aperçu du fichier "UserAuthenticator.php"

```
public function onAuthenticationSuccess(Request $request, TokenInterface $token, string $firewallName): ?Response
{
    $user = $token->getUser();

    if (in_array('ROLE_ADMIN', $user->getRoles())) {
        return new RedirectResponse($this->urlGenerator->generate('app_admin_'));
    }

    if (in_array('ROLE_USER', $user->getRoles())) {
        return new RedirectResponse($this->urlGenerator->generate('app_profile_'));
    }

    if ($targetPath = $this->getTargetPath($request->getSession(), $firewallName)) {
        return new RedirectResponse($targetPath);
    }

    return new RedirectResponse($this->urlGenerator->generate('app_home'));
}
```

Dans la partie Admin, **Administrateur va pouvoir**, voir, créer, modifier & supprimer les franchisés et là ou leurs structures.

Quand les entités ont été créé tout à l'heure, Symfony à générer automatiquement les fichier Repository, qui vont me permettre de récupérer les données dans les différentes table.

Aperçu du fichier "FranchiseRepository.php"

```
Repository > FranchiseRepository.php > PHP Intelephense > FranchiseRepository
<?php

namespace App\Repository;

use App\Entity\Franchise;
use Doctrine\Bundle\DoctrineBundle\Repository\ServiceEntityRepository;
use Doctrine\Persistence\ManagerRegistry;

/**
 * @extends ServiceEntityRepository<Franchise>
 *
 * @method Franchise|null find($id, $lockMode = null, $lockVersion = null)
 * @method Franchise|null findOneBy(array $criteria, array $orderBy = null)
 * @method Franchise[] findAll()
 * @method Franchise[] findBy(array $criteria, array $orderBy = null, $limit = null, $offset = null)
 */
class FranchiseRepository extends ServiceEntityRepository
{
    public function __construct(ManagerRegistry $registry)
    {
        parent::__construct($registry, Franchise::class);
    }
}
```

## Aperçu du fichier “AdminController.php”

```
<?php

namespace App\Controller;

use App\Entity\Franchise;
use Doctrine\Persistence\ManagerRegistry;
use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
use Symfony\Component\HttpFoundation\Response;
use Symfony\Component\Routing\Annotation\Route;

#[Route('/admin', name: "app_admin_")]
class AdminController extends AbstractController
{
    #[Route('/', name: '')]
    public function index(ManagerRegistry $doctrine): Response
    {
        $repository = $doctrine->getRepository(Franchise::class);
        $franchises = $repository->findAll();
        return $this->render('admin/index.html.twig', [
            "franchises" => $franchises,
        ]);
    }
}
```

## Aperçu du fichiers “index.html.twig”

```
s > admin > index.html.twig
{% extends 'base.html.twig' %}

{% block title %}Liste des Franchisés{% endblock %}
{% block body %}
    <h1>Liste des Franchisés</h1>

    <div class="input-control-search d-flex justify-content-center">
        <input type="text" id="searchbar" class="form-control" placeholder="Rechercher" aria-label="Search">
        <button class="btn" type="submit">
            <i class="fa-solid fa-magnifying-glass"></i>
        </button>
    </div>

    <div class="container">
        {% for franchise in franchises %}
            <div class="card">
                {% if franchise.name %}
                    <h5 class="card-title">{{ franchise.name }}</h5>
                {% endif %}
                <a href="{{ path('app_admin_list_structure', {'id': franchise.id}) }}">Voir la/les structure(s)</a>
                <div class="pra">
                    <a href="{{ path('app_admin_edit_franchise', {'id': franchise.id}) }}" id="btn" class="button">Modifier</a>
                    <a href="{{ path('app_admin_delete_franchise', {'id': franchise.id}) }}" id="btn" class="button">Supprimer</a>
                </div>
            </div>
        {% endfor %}
    </div>
{% endblock %}
```



Aperçu de la page d'accueil de la partie Administration.

The screenshot shows the 'Liste des Franchisés' page. At the top, there is a navigation bar with the 'Body Top' logo and links: 'Créer franchise', 'Créer Structure', 'Voir franchise', and 'Voir Structure'. A 'Se déconnecter' button is on the right. Below the navigation bar, the title 'Liste des Franchisés' is centered. A search bar with the placeholder 'Rechercher' and a magnifying glass icon is on the right. The main content area displays a grid of nine franchise cards. Each card represents a city: Lille, Douai, Seclin, Paris, Lens, Calais, Reims, Nancy, and Carvin. Each card contains a link 'Voir la/les structure(s)' and two buttons: 'Modifier' and 'Supprimer'. At the bottom of the page, there is a footer with the 'Body Top' logo and a placeholder text 'Lorem ipsum, dolor sit amet consectetur adipisicing elit.'

On peut apercevoir la liste de tous les franchisés, avec la possibilité d'accéder au(x) structure(s) lié à cette franchise.

Exemple en cliquant sur voir la/les structure(s) de la franchise de Lille nous arrivons sur cette page

The screenshot shows the 'Liste de la/des structure(s) de la franchise de Lille' page. The navigation bar is identical to the previous page. The title 'Liste de la/des structure(s) de la franchise de Lille' is centered. Below the title, there is a grid of four cards, each representing a specific structure in Lille. Each card displays the address, postal code, and city (Lille), followed by a link 'Voir les modules' and two buttons: 'Modifier' and 'Supprimer'. At the bottom of the page, there is a footer with the 'Body Top' logo, a placeholder text 'Lorem ipsum, dolor sit amet consectetur adipisicing elit.', and a copyright notice 'Copyright 2022'.

on peut également modifier et supprimer chaque franchises et structures, de même une confirmation de sécurité nous sera demander afin de valider notre choix.

Aperçu du fichier “FanchiseController.php”

```
StructureController.php X
rc > Controller > StructureController.php > ...
> for

1  <?php
2
3  namespace App\Controller;
4
5  use App\Entity\Franchise;
6  use App\Entity\User;
7  use App\Entity\Structure;
8  use App\Form\UserStructureType;
9  use App\Form\StructureType;
10 use Doctrine\Persistence\ManagerRegistry;
11 use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
12 use Symfony\Component\HttpFoundation\Request;
13 use Symfony\Component\HttpFoundation\Response;
14 use Symfony\Component\PasswordHasher\Hasher\UserPasswordHasherInterface;
15 use Symfony\Component\Routing\Annotation\Route;
16
17 #[Route('/admin', name: "app_admin_")]
18 class StructureController extends AbstractController
19 {
20     #[Route('/structure/{id<d+>}', name: 'list_structure')]
21     public function list(Franchise $franchise): Response
22     {
23         $structure = $franchise->getStructure();
24
25         return $this->render('admin/structure/list.html.twig', [
26             'franchise' => $franchise,
27             'structure' => $structure
28         ]);
29     }
30
31     #[Route('/structure', name: 'structure')]
32     public function listglobal(ManagerRegistry $doctrine): Response
33     {
34
```

```
#[Route('/structure/new', name: 'structure_new')]
public function add(Request $request, UserPasswordHasherInterface $userPasswordHasher, ManagerRegistry $doctrine): Response
{
    $this->denyAccessUnlessGranted('IS_AUTHENTICATED_FULLY');
    $user = new User($userPasswordHasher);
    $form1 = $this->createForm(UserStructureType::class, $user);
    $form1->handleRequest($request);

    if ($form1->isSubmitted() && $form1->isValid()) {
        $em = $doctrine->getManager();
        $em->persist($user->getStructure());
        $em->persist($user);
        $em->flush();
        return $this->redirectToRoute('app_admin_');
    }
    return $this->render('admin/structure/create.html.twig', [
        'form1' => $form1->createView(),
    ]);
}

#[Route('/structure/edit/{id<id>}', name: "edit_structure")]
public function update(Request $request, Structure $structure, ManagerRegistry $doctrine): Response
{
    $this->denyAccessUnlessGranted('IS_AUTHENTICATED_FULLY');
    $form1 = $this->createForm(StructureType::class, $structure);
    $form1->handleRequest($request);
    if ($form1->isSubmitted() && $form1->isValid()) {
        $em = $doctrine->getManager();
        $em->flush();
        return $this->redirectToRoute('app_admin_');
    }
    return $this->render('admin/structure/create.html.twig', [
        "form1" => $form1->createView()
    ]);
}
```


## Aperçu du fichiers “script.js”

```
// confirmation de sécurité A chaque modification ou suppression //
const btn = document.querySelector('#btn');

function evenement() {
    alert('Veuillez confirmer votre choix')
}

btn.addEventListener("click", evenement);
```

## Formulaire de modification de la structure de Lille au 10 rue de paris - 59000


[Créer franchise](#)
[Créer Structure](#)
[Voir franchise](#)
[Voir Structure](#)
[Se déconnecter](#)

Créer ou modifier une structure

Adresse : 10 rue paris

Code postal : 59000

Ville : Lille

Activer la structure : ☐ Oui ☒ Non

Franchise : Lille

Module

Gérer planning équipe : ☐ Oui ☒ Non

Service d'inscription sur internet : ☐ Oui ☒ Non

Négociation prix fournisseur : ☐ Oui ☒ Non

Vendre des boissons : ☐ Oui ☒ Non

Promotion de la salle : ☐ Oui ☒ Non


Faire son mailing : ☐ Oui ☒ Non

Suivre les formations : ☐ Oui ☒ Non

Valider

Body Top

## Détails des permissions actifs ou non de la structure de Lille au 10 rue de paris - 59000


[Créer franchise](#)
[Créer Structure](#)
[Voir franchise](#)
[Voir Structure](#)
[Se déconnecter](#)

Page de la structure de Lille - 10 rue paris

Lille

10 rue paris - 59000

Gérer planning équipe

Service d'inscription sur internet

Négociation prix fournisseur

Vendre des boissons

Promotion de la salle

Faire son mailing

Suivre les formations

Modifier

Body Top

Lorem ipsum, dolor sit amet consectetur adipisicing elit.

Perferendis dolor ab illo iure harum

Copyright 2022

J'ai imbriqué 2 formulaires ensemble StructureType.php avec ModuleType.php

Grace a cette ligne

```
->add('Module', ModuleType::class)
```

J'ai mis en place plusieurs contrainte sur les différents champs du formulaire

Afin de lier la structure du 10 rue de paris – 59000 Lille avec la franchise de Lille

J'ai ajouté un menu déroulant ou apparait tous les noms des franchises en ordre alphabétique

Grace a se champs

```
->add('franchise', EntityType::class, [
    'expanded' => false,
    'class' => Franchise::class,
    'multiple' => false,
    'required' => false,
    'query_builder' => function (EntityManager $er) {
        return $er->createQueryBuilder('h')
            ->orderBy('h.name', 'ASC');
    },
    'choice_label' => 'name',
    'attr' => [
        'class' => 'select2'
    ]
])
```

```
StructureType.php > PHP Intelephense > StructureType > buildForm
        'minMessage' => "L'adresse doit faire plus de 5 caractères",
        'maxMessage' => "L'adresse ne doit pas faire plus de 255 caractères"]],
    ]
})
->add('zipcode', TextType::class, [
    'label' => "Code postal : ",
    'required' => true,
    'constraints' => [
        new NotBlank(["message" => "Le code postal ne doit pas être vide !"]),
        new Length([ 5, "Message" => "Le code postal doit faire 5 caractères"]),
    ]
])
->add('city', TextType::class, [
    'label' => "Ville : ",
    'required' => true,
    'constraints' => [
        new NotBlank(["message" => "Le nom de ville ne doit pas être vide !"]),
        new Length(["min" => 2, "max" => 150,
            "minMessage" => "Le nom de ville doit faire plus de 2 caractères",
            "maxMessage" => "Le nom de ville ne doit pas faire plus de 150 caractères"]),
    ]
])
->add('is_active', ChoiceType::class, [
    'expanded' => true,
    'multiple' => false,
    'label' => 'Activer la structure : ',
    'choices' => [
        'Oui' => 'true',
        'Non' => 'false',
    ]
])
->add('franchise', EntityType::class, [
    'expanded' => false,
    'class' => Franchise::class,
    'multiple' => false,
    'required' => false,
    'query_builder' => function (EntityRepository $er) {
        return $er->createQueryBuilder('h')
            ->orderBy('h.name', 'ASC');
    },
    'choice_label' => 'name',
    'attr' => [
        'class' => 'select2'
    ]
])
->add('Module', ModuleType::class)
;
}
```

On peut également y apercevoir une barre de recherche dynamique, recherche sur les champs de la card



Grace a cette fonction Javascript

Aperçu du fichiers “script.js”

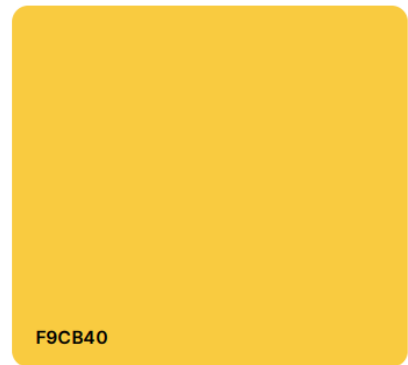
```
// Barre de recherche dynamique //
const searchBar = document.querySelector("#searchbar");

searchBar.addEventListener("keyup", (e) => {
  const searchedLetters = e.target.value;
  const cards = document.querySelectorAll(".card");
  filterElements(searchedLetters, cards);
});

function filterElements(letters, elements) {
  for (let i = 0; i < elements.length; i++) {
    if (elements[i].textContent.toLowerCase().includes(letters)) {
      elements[i].style.display = "block";
    } else {
      elements[i].style.display = "none";
    }
  }
}
```

## Charte graphique

### Couleurs utilisées

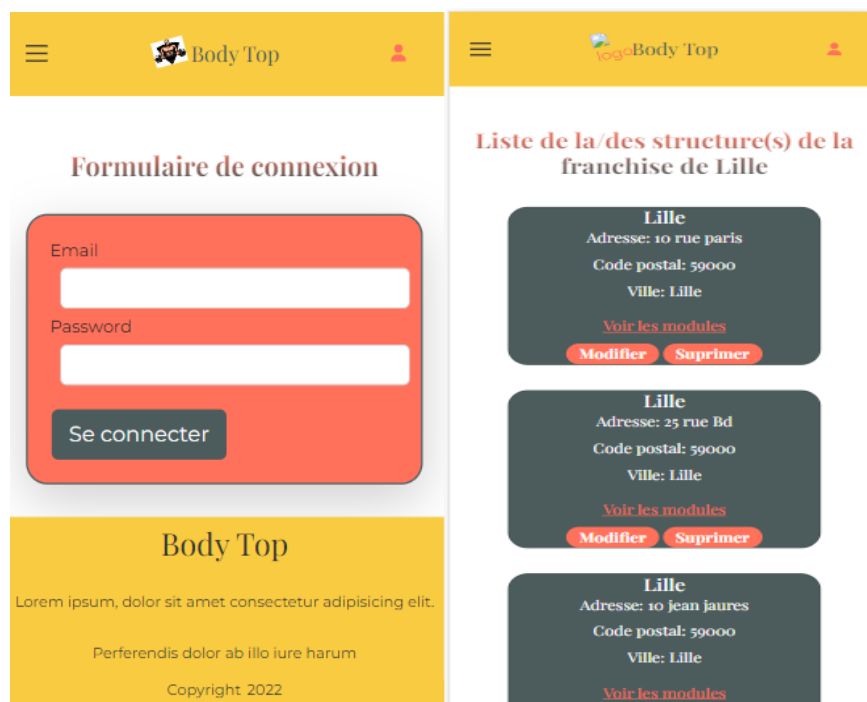


### Polices utilisées

montserrat, monospace;

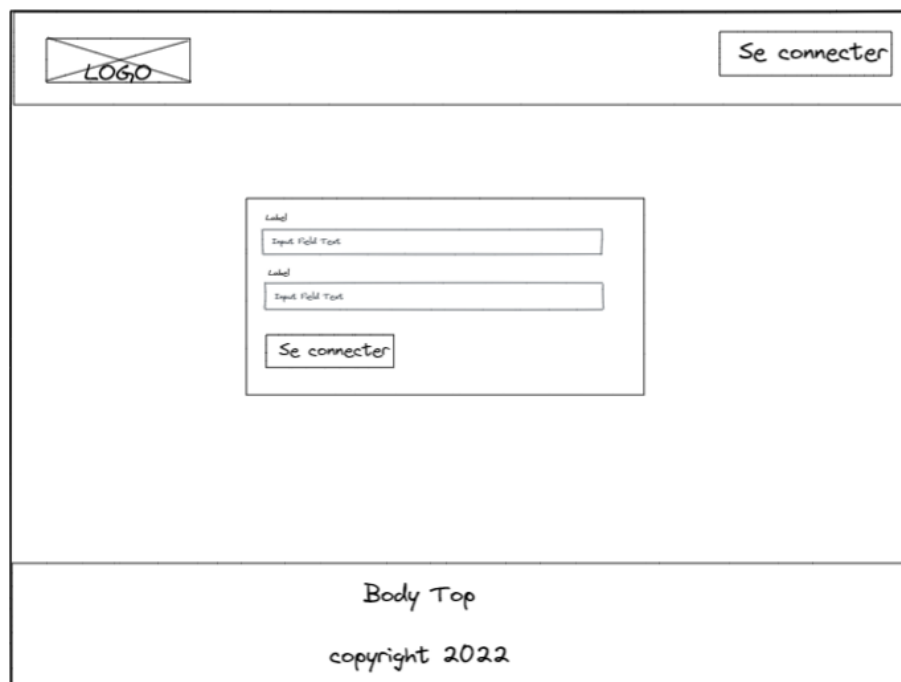
playfair display

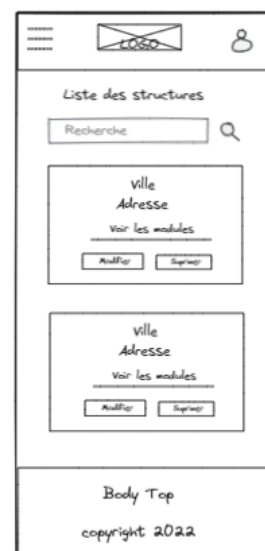
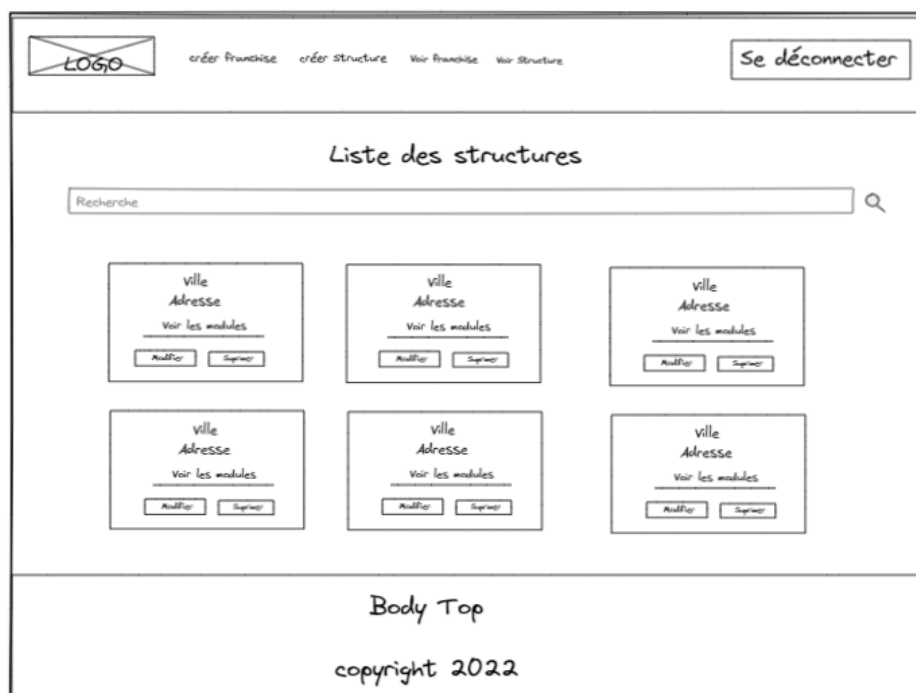
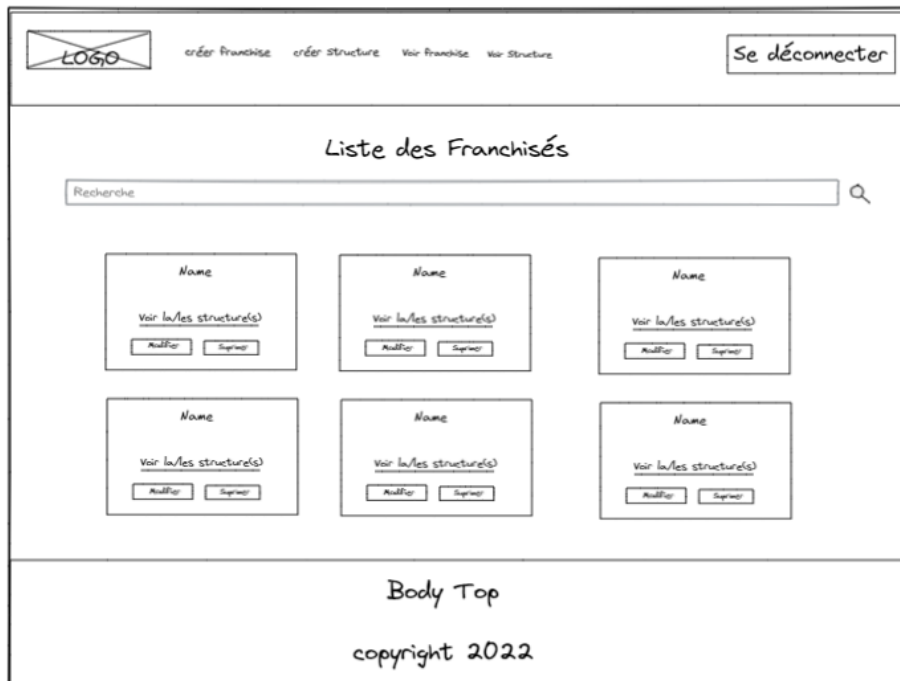
### Elements






## Wireframes






[créer franchise](#)
[créer Structure](#)
[Voir franchise](#)
[Voir Structure](#)

Se déconnecter

## Créer ou modifier une structure

Label

Label

Label

Checkboxes

☐ ☒

Dropdown

dropdown

Checkboxes

☐ ☒

Checkboxes

☐ ☒

Checkboxes

☐ ☒

Checkboxes

☐ ☒

Checkboxes

☐ ☒



Checkboxes

☐ ☒

Valider

Body Top

copyright 2022

## Créer ou modifier une structure

Label

Label

Label

Checkboxes

☐ ☒

Dropdown

dropdown

Checkboxes

☐ ☒

Checkboxes

☐ ☒

Checkboxes

☐ ☒


Checkboxes

☐ ☒

Valider

Body Top

copyright 2022


[créer franchise](#)
[créer Structure](#)
[Voir franchise](#)
[Voir Structure](#)

Se déconnecter

## Page de la structure de Name - adresse

ville

adresse

☐

 Place your copy here
 

☐

 Place your copy here
 

☐

 Place your copy here
 

☐

 Place your copy here
 

☐

 Place your copy here
 

☐

 Place your copy here
 



☐

 Place your copy here

Modifier

Body Top

copyright 2022

## Page de la structure de Name - adresse

ville

adresse

☐

 Place your copy here
 

☐

 Place your copy here
 

☐

 Place your copy here
 

☐

 Place your copy here
 

☐

 Place your copy here
 

☐

 Place your copy here
 

☐

 Place your copy here

Modifier

Body Top

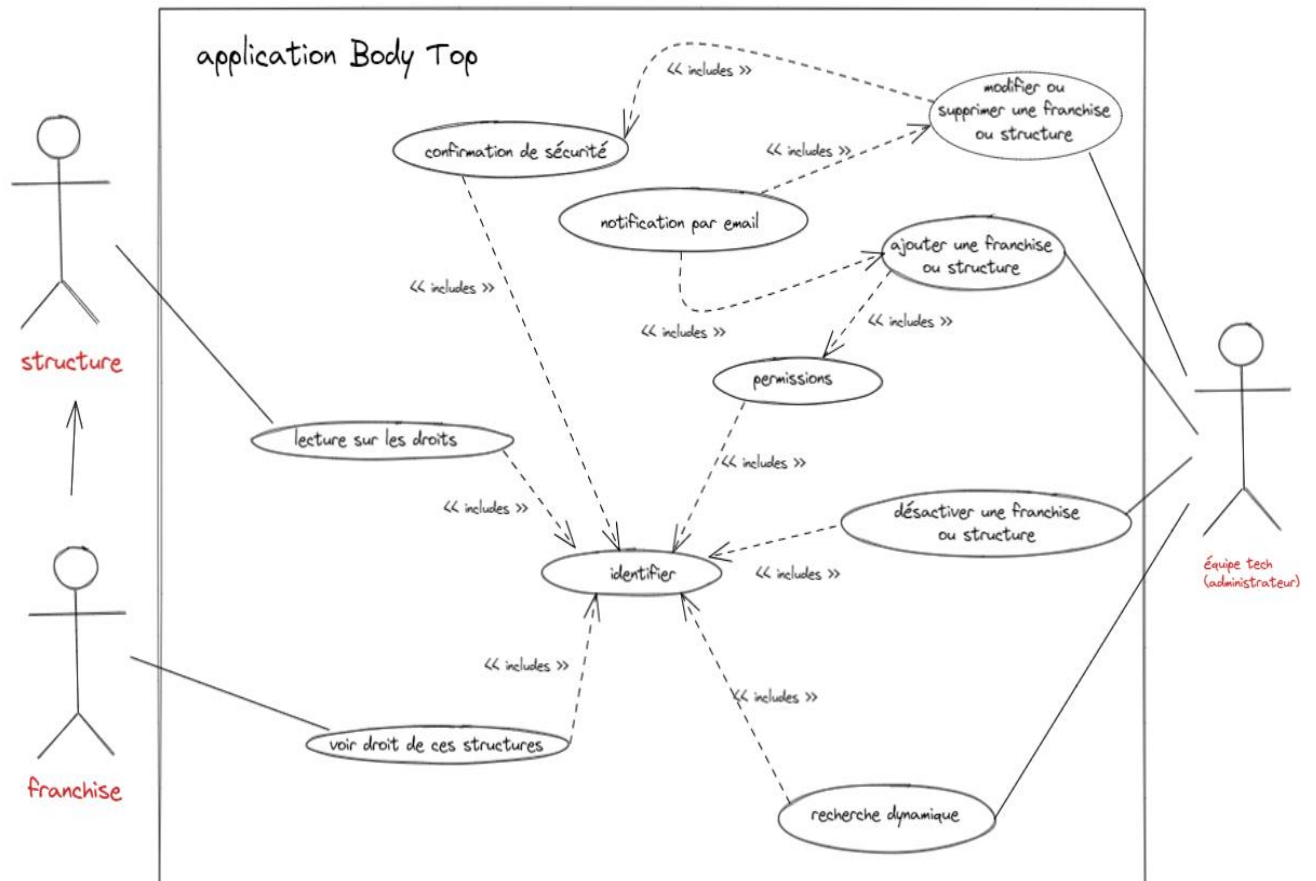
copyright 2022

©Studi - Reproduction interdite

ECF - Développeur Web et Web Mobile

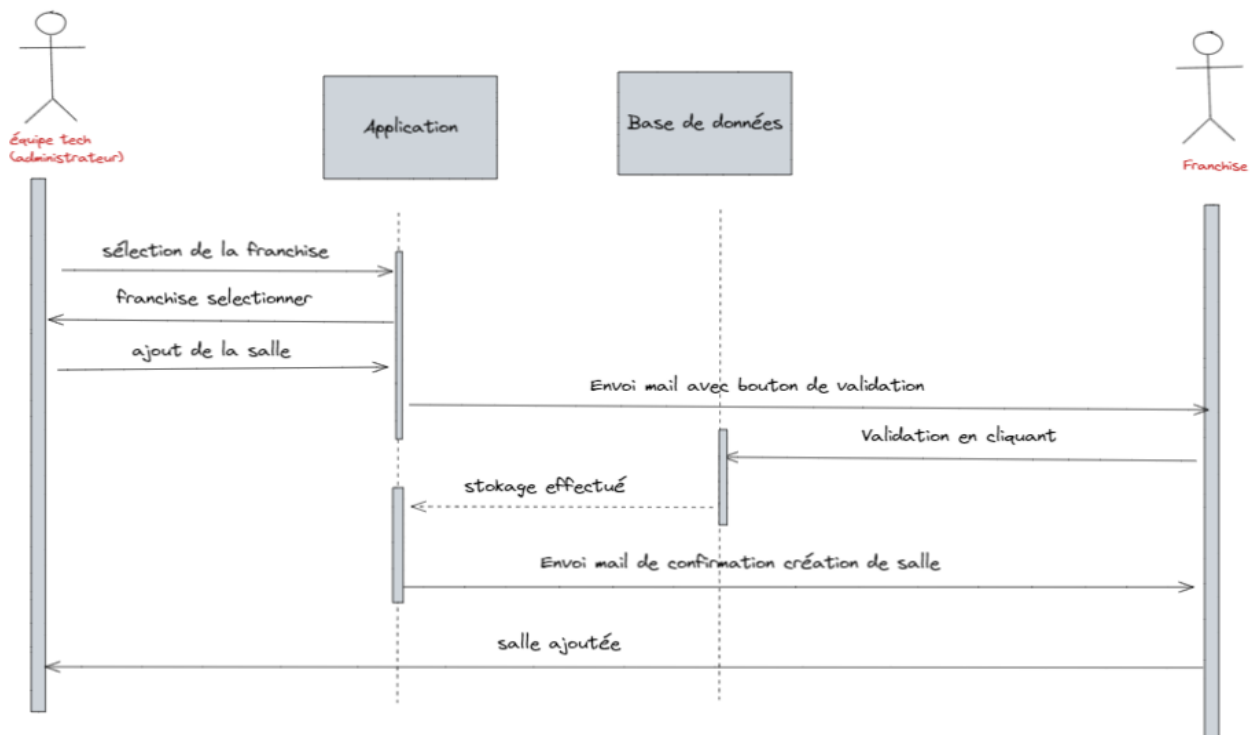
Page 24 sur 34

## Diagramme de cas d'utilisation

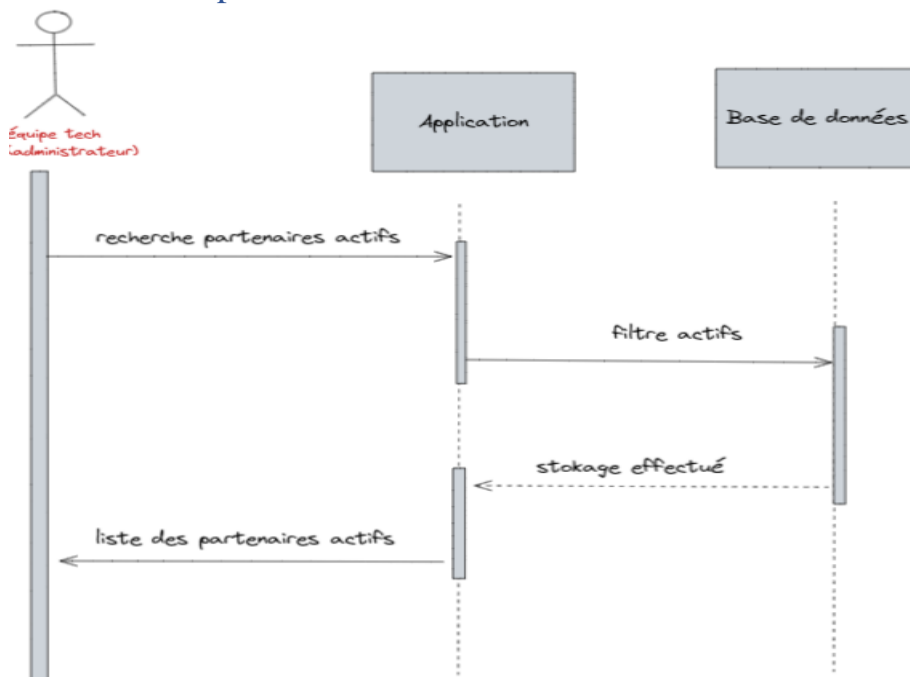


## Diagramme de Séquence

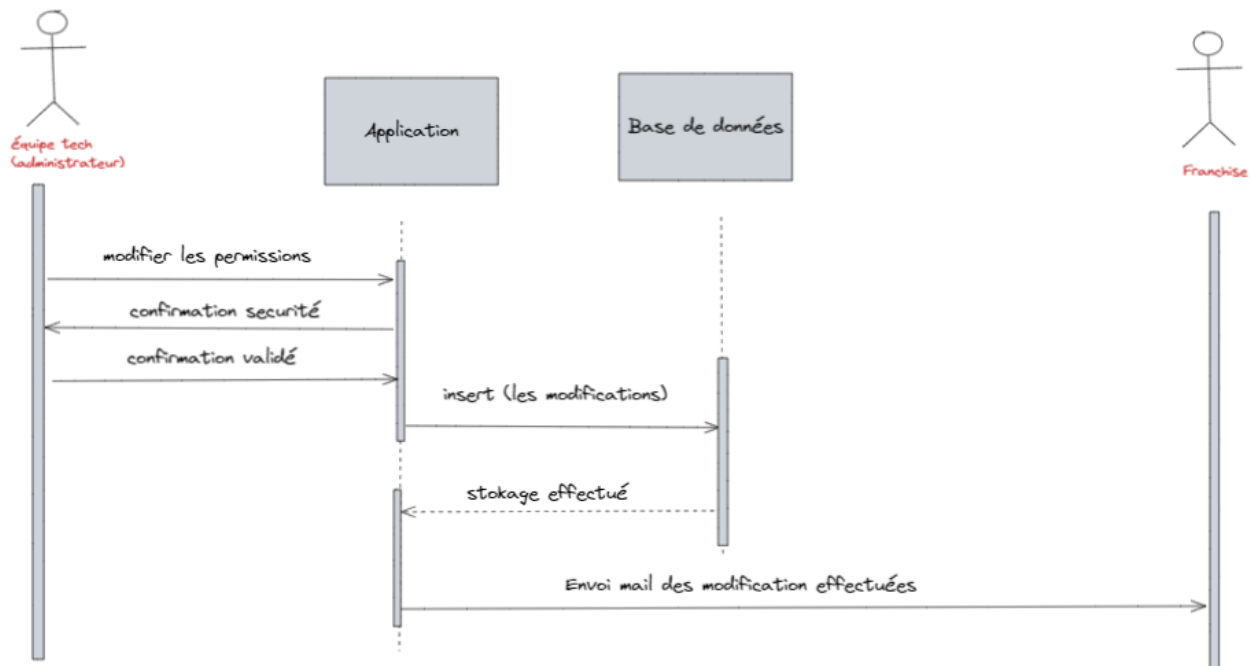
### Ajout de salle



### Sélection des partenaires actifs



## Modification des droits



## Conclusion

Concernant le projet, les fonctionnalités ne sont pas toutes développées. Mon rendu EFC étant avant le 31/10/2022, il me reste encore 4 mois avant examen passage courant mars pour continuer l'application et rendre un dossier plus complet afin de proposer ma version final.

En effet, le développement est un défi de tous les jours, car il me faut apprendre à

l'aide de la documentation officielle, à l'aide des forums dédiés, de tutoriels vidéo, et reproduire ce que je viens d'apprendre sur mon projet. Je pense que c'est la

meilleure méthode de travail afin de progresser rapidement, et que c'est une

bonne façon d'entrevoir ce qu'est le métier de développeur.

## À l'avenir...

Je voudrais approfondir Symfony plus en détail afin d'être plus à l'aise. Et pouvoir aller plus loin dans les fonctionnalités. Et apprendre d'autres langages ce métier me passionne beaucoup c'est le début d'une grande aventure.

Je m'investis beaucoup même si les débuts sont un peu compliqués mais je me sens progresser de jour en jour.

Concernant mes projets personnels, actuellement je développe une application pour une artiste peintre de mon village, avec une partie administrateur pour qu'elle puisse mettre en ligne ces nouveaux tableaux. Je la développe en Symfony et je voudrais la présenter cela dans mon dossier professionnel

## Ce que cette expérience m'a apporté

Cette expérience m'a permis d'avoir une vision d'ensemble de la création d'un site web, du code en passant par la maquette et la documentation.

J'ai beaucoup apprécié travailler sur le Framework Symfony, et souhaiterais vraiment continuer dans ce domaine.

Je sais maintenant que le plus important dans un projet n'est pas forcément le code, et si je devais le refaire, je travaillerais plus sur la maquette, et le modèle conceptuel de données

5. Description de la veille, effectuée par le candidat durant le projet, sur les vulnérabilités de sécurité

Recherche anglophone:

J'ai effectué une recherche sur les Les jetons utilisés pour la protection CSRF mots-clés on était : CSRF TOKEN. Ce qui m'a retourné sur les différents sites suivants.

<https://portswigger.net/web-security/csrf/tokens>

<https://fr.barracuda.com/glossary/csrf>

<https://symfony.com/doc/current/security/csrf.html>

6. Description d'une situation de travail ayant nécessité une recherche, effectuée par le candidat durant le projet, à partir de site anglophone

## How to Implement CSRF Protection

CSRF is a method by which a malicious user attempts to make your legitimate users unknowingly submit data that they don't intend to submit.

CSRF protection works by adding a hidden field to your form that contains a value that only you and your user know. This ensures that the user - not some other entity - is submitting the given data.

The tokens used for CSRF protection are meant to be different for every user and they are stored in the session. That's why a session is started automatically as soon as you render a form with CSRF protection.

Moreover, this means that you cannot fully cache pages that include CSRF protected forms. As an alternative, you can:

- Embed the form inside an uncached [ESI fragment](#) and cache the rest of the page contents;
- Cache the entire page and load the form via an uncached AJAX request;
- Cache the entire page and use [hinclude.js](#) to load the CSRF token with an uncached AJAX request and replace the form field value with it.



7. Extrait du site anglophone, utilisé dans le cadre de la recherche décrite précédemment, accompagné de la traduction en français effectuée par le candidat sans traducteur automatique (environ 750 signes).

CSRF est une méthode par laquelle un utilisateur malveillant tente de faire en sorte que vos utilisateurs légitimes soumettent sans le savoir des données qu'ils n'ont pas l'intention de soumettre.

La protection CSRF fonctionne en ajoutant un champ masqué à votre formulaire qui contient une valeur que seuls vous et votre utilisateur connaissez. Cela garantit que l'utilisateur - et non une autre entité - soumet les données fournies.

Les jetons utilisés pour la protection CSRF sont censés être différents pour chaque utilisateur et ils sont stockés dans la session. C'est pourquoi une session est démarrée automatiquement dès que vous affichez un formulaire avec la protection CSRF.

De plus, cela signifie que vous ne pouvez pas entièrement mettre en cache les pages contenant des formulaires protégés par CSRF. En alternative, vous pouvez :

- Intégrez le formulaire dans un [fragment ESI](#) non mis en cache et mettez en cache le reste du contenu de la page ;
- Mettez en cache la page entière et chargez le formulaire via une requête AJAX non mise en cache ;
- Mettez en cache la page entière et utilisez [hinclude.js](#) pour charger le jeton CSRF avec une requête AJAX non mise en cache et remplacez la valeur du champ de formulaire par celle-ci.

8. Autres ressources

9. Informations complémentaires

