

# DOSSIER PROFESSIONNEL (DP)

*Nom de naissance* ▶ HOUQUET  
*Nom d'usage* ▶ HOUQUET  
*Prénom* ▶ Davina  
*Adresse* ▶ 10 Rue Lucie Berger, 67200 Strasbourg

## Titre professionnel visé

Développeur web et web mobile niveau 5

### MODALITE D'ACCES :

- ☒ Parcours de formation  
☐ Validation des Acquis de l'Expérience (VAE)

## Présentation du dossier

Le dossier professionnel (DP) constitue un élément du système de validation du titre professionnel. **Ce titre est délivré par le Ministère chargé de l'emploi.**

Le DP appartient au candidat. Il le conserve, l'actualise durant son parcours et le présente **obligatoirement à chaque session d'examen.**

Pour rédiger le DP, le candidat peut être aidé par un formateur ou par un accompagnateur VAE.

Il est consulté par le jury au moment de la session d'examen.

### Pour prendre sa décision, le jury dispose :

1. des résultats de la mise en situation professionnelle complétés, éventuellement, du questionnaire professionnel ou de l'entretien professionnel ou de l'entretien technique ou du questionnement à partir de productions.
2. du **Dossier Professionnel** (DP) dans lequel le candidat a consigné les preuves de sa pratique professionnelle
3. des résultats des évaluations passées en cours de formation lorsque le candidat évalué est issu d'un parcours de formation
4. de l'entretien final (dans le cadre de la session titre).

*[Arrêté du 22 décembre 2015, relatif aux conditions de délivrance des titres professionnels du ministère chargé de l'Emploi]*

### Ce dossier comporte :

- ▶ pour chaque activité-type du titre visé, un à trois exemples de pratique professionnelle ;
- ▶ un tableau à renseigner si le candidat souhaite porter à la connaissance du jury la détention d'un titre, d'un diplôme, d'un certificat de qualification professionnelle (CQP) ou des attestations de formation ;
- ▶ une déclaration sur l'honneur à compléter et à signer ;
- ▶ des documents illustrant la pratique professionnelle du candidat (facultatif)
- ▶ des annexes, si nécessaire.

*Pour compléter ce dossier, le candidat dispose d'un site web en accès libre sur le site.*



<http://travail-emploi.gouv.fr/titres-professionnels>

## Sommaire

### Exemples de pratique professionnelle

#### Développer la partie front-end d'une application web ou web mobile sécurisée p.

- ▶ Exercice de formation « **Academia** » ..... p. 5
- ▶ Exercice réalisé dans le cadre du stage « **Bloc Image** » ..... p. 7
- ▶ Projet de fin de formation « **Au Paradis d'Eve** » ..... p. 9

#### Développer la partie back-end d'une application web ou web mobile sécurisée p.

- ▶ Exercice de formation **MongoDB** ..... p. 11
- ▶ Exercice de formation « **Mooviz** » ..... p. 13
- ▶ Projet de fin de formation « **Au Paradis d'Eve** » ..... p. 15

#### Titres, diplômes, CQP, attestations de formation *(facultatif)*

p. \_\_\_\_\_

#### Déclaration sur l'honneur

p. \_\_\_\_\_

#### Documents illustrant la pratique professionnelle *(facultatif)*

p. \_\_\_\_\_

#### Annexes *(Si le RC le prévoit)*

p. 20

# **EXEMPLES DE PRATIQUE PROFESSIONNELLE**

## Activité-type 1

Développer la partie front-end d'une application web ou web mobile sécurisée

Exemple n°1 ► Exercice de formation « Academia »

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

« Academia » réalisée dans le cadre de l'exercice de formation « Sessions », est une **application de gestion de sessions de formation**.

L'une des premières étapes de ce projet a été d'**identifier les différentes tâches** à effectuer en les divisant en sous-tâches au besoin. Les lister dans l'outil de gestion Trello a permis d'avoir une vision globale de l'organisation et de la progression du projet.

Dans le cadre du développement front-end, la **réalisation d'une maquette** avec **Figma** s'est imposée comme une étape nécessaire pour appréhender la partie **UI/UX** de ce projet.

Le développement de la partie front-end a été réalisé avec **HTML** pour la création des différentes balises ainsi que pour l'affichage des éléments de la partie back-end, avec **CSS** pour appliquer du style, des effets au survol sur les différents éléments, ainsi que le responsive design grâce au media queries, et enfin avec **JavaScript** pour permettre l'usage du mode sombre.

### 2. Précisez les moyens utilisés :

Gestion de projet : **Trello**

Maquettage : **Figma**

Versionnage : **Git** (Github)

Environnement de développement : **Laragon**

Editeur de code : **VSCode**

Technologies : **HTML5, CSS, PHP, JavaScript**

Moteur de template : **Twig**

Framework : **Symfony 6**

OS : **Windows 11**

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

« **Academia** » est le résultat d'un travail individuel soutenu et supervisé par les formateurs.

## 4. Contexte

Nom de l'entreprise, organisme ou association		▶ <b>ELAN Formation</b>	
Chantier, atelier, service	▶	<b>ELAN Formation</b>	
Période d'exercice	▶	Du	18/09/2023 au 13/10/2023

## 5. Informations complémentaires (facultatif)

Voir annexe p.20

## Activité-type 1

Développer la partie front-end d'une application web ou web mobile sécurisée

Exemple n°2 ► ► Exercice réalisé dans le cadre du stage « Bloc Image »

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Au cours de mon stage, j'ai développé un composant image en suivant les étapes suivantes : collecte d'informations à partir du ticket **Redmine**, analyse de la maquette **Figma** du client, lecture de la documentation Ibexa sur les **variations d'images**, extraction du lien pour rendre les images cliquables vers un lien interne ou externe au projet, création de la vue et du contrôleur, gestion du **responsive** avec **SCSS**, conversion des images en WebP pour optimisation, et utilisation de **GitLab** pour le versionnage sur le principe GitFlow.

*Accord demandé préalablement au responsable pour présenter cet exercice*

### 2. Précisez les moyens utilisés :

Gestion de projet : **Redmine**

Maquettage : **Figma**

Versionnage : **Git** (GitLab) avec GitFlow

Environnement de développement : **Docker**

Editeur de code : **PhpStorm**

Technologies : **HTML, SCSS, PHP, Ibexa**

Moteur de template : **Twig**

Framework : **Symfony 6**

OS : **Linux**

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

Exercice réalisé dans le cadre du stage, soutenu par le tuteur et d'autres membres de l'équipe.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **ELAN Formation**

Chantier, atelier, service ► **ELAN Formation**

Période d'exercice ► Du **23/10/2023** au **13/11/2023**

## 5. Informations complémentaires (facultatif)

*Voir annexe p.25*



## Activité-type 1

Développer la partie front-end d'une application web ou web mobile sécurisée

Exemple n°3 ► *Projet de fin de formation « Au Paradis d'Eve »*

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Dans le cadre de mon projet de fin de parcours, j'ai conçu "**Au Paradis d'Eve**", un site de réservation de chambres d'hôtes. En commençant par la création de la maquette sur **Figma** et du **wireframe**, établissant ainsi une ligne directrice à suivre.

Sur le plan du développement **front-end**, j'ai appliqué des **contraintes de validation** précises pour garantir la qualité des informations saisies en précisant par exemple le type de données attendues dans les formulaires.

Le moteur de template **Twig** a été utilisé pour la gestion des contenus. Du côté de **l'expérience utilisateur** (UX), **JavaScript** a permis d'intégrer des améliorations telles que le bouton "**scroll to top**" et le **menu hamburger** pour les appareils mobiles.

D'ailleurs, le projet a été conçu en suivant une approche "**mobile first design**" avec un design **responsive**.

Concernant les normes **RGPD** du côté front-end, la minimisation des informations collectées a été appliquée, en prenant en compte la question de la conservation des données.

En mettant l'accent sur le **référencement naturel**, les bonnes pratiques ont guidé l'utilisation des balises HTML dans ce projet, incluant les **balises sémantiques et des balises strong**, le respect de leur hiérarchie ainsi que de celle des titres. Le site est ainsi **responsive** et **mobile-friendly**, et le contenu est axé sur la **qualité** pour améliorer sa visibilité sur les moteurs de recherche.

### 2. Précisez les moyens utilisés :

Gestion de projet : **Trello**

Maquettage : **Figma**

Versionnage : **Git** (Github)

Environnement de développement : **Laragon**

Editeur de code : **VSCode**

Technologies : **HTML5, CSS, PHP, JavaScript**

Moteur de template : **Twig**

Framework : **Symfony 6**

Bundle : **DomPdf, Calendar-Bundle**

OS : **Windows 11**

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

Ce projet de fin de parcours a été réalisé en autonomie avec le soutien des formateurs du centre de formation Elan.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **ELAN Formation**

Chantier, atelier, service ► **ELAN Formation**

Période d'exercice ► Du **18/12/2023** au **05/02/2024**

## 5. Informations complémentaires (facultatif)

Voir annexe p.29

## Activité-type 2

Développer la partie back-end d'une application web ou web mobile sécurisée

Exemple n°1 ► Exercice NoSQL - MongoDB

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

#### Création de bases de données, l'ajout de documents et l'exécution de requêtes

Dans cet exercice **MongoDB**, j'ai installé **Mongosh** et **MongoDB Compass**. J'ai créé une base de données "dbcommerce" et une collection "clients".

En utilisant **Mongosh** et **Compass**, j'ai inséré, mis à jour, recherché et supprimé des documents dans la collection. L'exportation et l'importation de données ont été démontrées, et la flexibilité de MongoDB en termes de structure de données a été soulignée.

L'utilisation du **shell** a été nécessaire pour certaines opérations, comme le vidage de la collection. En résumé, l'exercice a couvert des opérations de base, offrant une introduction pratique à MongoDB et à ses fonctionnalités **NoSQL**.

### 2. Précisez les moyens utilisés :

Versionnage : **Git**

Environnement de développement : **MongoDB**

Editeur de code : **VSCode**

Technologies : **NoSQL**

OS : **Windows 11**

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

Cet exercice est le résultat d'un travail individuel.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **ELAN Formation**

Chantier, atelier, service ► **ELAN Formation**

Période d'exercice ► Du **01/10/2024** au **10/02/2024**

## 5. Informations complémentaires (facultatif)

*Voir annexe p.38*

## Activité-type 2

Développer la partie back-end d'une application web ou web mobile sécurisée

Exemple n°2 ► «Mooviz », exercice de formation

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

J'ai réalisé Mooviz dans le cadre de l'exercice Cinéma en formation, qui consistait à créer une application listant des films avec leurs détails, le casting qui le compose, les détails des acteurs avec leur rôles, ainsi que les réalisateurs qui sont à l'origine des films.

J'ai utilisé **Trello** pour la gestion de ce projet. Le modèle conceptuel et le modèle logique de la base de données ont été réalisés avec **Looping**.

**Git** a été utilisé pour le versionnage du code. L'environnement de développement a été configuré avec **Laragon**, **HeidiSQL** et **Docker**. L'éditeur de code choisi était **VSCode** (Visual Studio Code). Les technologies principales utilisées dans le développement de Mooviz incluent **HTML5**, **CSS** et **PHP PDO (PHP Data Objects)**, pour interagir avec la base de données, permettant ainsi la manipulation des données relatives aux films, aux acteurs et aux réalisateurs.

En ce qui concerne la sécurité, des contraintes de taille et de type MIME ont été mis en place, en plus de l'identifiant unique et haché, dans le téléchargement de fichier afin de se protéger contre la **faille Upload** qui consiste à soumettre des fichiers malveillants.

### 2. Précisez les moyens utilisés :

Gestion de projet : **Trello**

Modélisation de base de données : **Looping**

Versionnage : **Git**

Environnement de développement : **Laragon**, **HeidiSQL**, **Docker**

Editeur de code : **VSCode**

Technologies : **HTML5**, **CSS**, **PHP PDO**

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

Exercice réalisé dans le cadre du stage, soutenu par le tuteur et d'autres membres de l'équipe.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **ELAN Formation**

Chantier, atelier, service ► **ELAN Formation**

Période d'exercice ► Du **27/11/2023** au **15/12/2023**

## 5. Informations complémentaires (facultatif)

Voir annexe p.35

## Activité-type 2

Développer la partie back-end d'une application web ou web mobile sécurisée

Exemple n°3 ► *Projet de fin de formation « Au Paradis d'Eve »*

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Dans le cadre de mon projet de fin de formation, j'ai eu l'opportunité de développer "**Au Paradis d'Eve**", un **site de réservation de chambres d'hôtes**. Utilisant le framework **Symfony** et ses bundles, le projet est basé sur le paradigme de la **programmation orientée objet** et suit le modèle **MVP** (Modèle-Vue-Présentateur).

La gestion de la base de données repose sur le Système de Gestion de Base de Données **MySQL**, avec la conception du modèle conceptuel de données (**MCD**) et du modèle logique de données (**MLD**) réalisée avec Looping en suivant la méthode Merise.

La sécurité du projet est assurée par des contraintes de validation (validator), et l'utilisation de **Doctrine ORM** pour la manipulation des données en PHP.

Des **login\_throttling** ont été mis en place afin de se prémunir de **l'attaque par dictionnaire**, permettant de limiter le nombre de tentatives de connexions par minute. Des types de données ont été précisés dans les formulaires afin d'apporter une sécurité supplémentaire contre la faille **XSS** (Cross Site Scripting) qui consiste à injecter du code malveillant dans les formulaires, et contre quoi Symfony nous offre déjà une protection avec l'échappement automatique de **Twig**, son moteur de template. Des requêtes paramétrées ont été effectuées avec Doctrine (**Doctrine Query Language**) permettant de se prémunir contre les **injections SQL**. Aussi, un token est généré dans la session d'un utilisateur, un autre lorsqu'il utilise un formulaire qui permettra de les comparer lors de sa soumission et de se prémunir contre la faille **CRSF** (Cross Site Request Forgery). Enfin, des contraintes de taille et de type MIME ont été mis en place, en plus de l'identifiant unique et haché, dans le téléchargement de fichier afin de se protéger contre la **faille Upload** qui consiste à soumettre des fichiers malveillants.

Un **certificat SSL** (Secure Socker Layer) a également été mis en place localement afin d'offrir une connexion chiffrée à l'utilisateur.

Concernant la conformité au **RGPD**, l'application permet de réserver sans obligation de créer un compte et ne collecte que les informations nécessaires. L'inscription comprend une demande de confirmation d'autorisation d'utilisation des données. Des **autorisations** distinctes pour les **utilisateurs** et les **administrateurs** ont été mises en place, ces derniers disposant d'une interface dédiée à l'administration.

### 2. Précisez les moyens utilisés :

# DOSSIER PROFESSIONNEL (DP)

Gestion de projet : **Trello**  
Maquettage : **Figma**  
Versionnage : **Git** (Github)  
Environnement de développement : **Laragon**  
Base de données : **MySQL (HeidiSQL)**  
Editeur de code : **VSCode**  
Technologies : **HTML, CSS, PHP, JavaScript**  
Moteur de template : **Twig**  
Framework : **Symfony 6**  
Bundle : **DomPdf**  
OS : **Windows 11**

## 3. Avec qui avez-vous travaillé ?

La réalisation de ce projet de fin de parcours a été effectuée en autonomie, avec le soutien des formateurs du centre de formation Elan.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **ELAN Formation**

Chantier, atelier, service ► **ELAN Formation**

Période d'exercice ► Du **16/10/2023** au **31/12/2023**

## 5. Informations complémentaires (facultatif)



## DOSSIER PROFESSIONNEL (DP)

*Voir annexe p.29*

## DOSSIER PROFESSIONNEL (DP)

### Titres, diplômes, CQP, attestations de formation

*(facultatif)*

Intitulé	Autorité ou organisme	Date
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.

## DOSSIER PROFESSIONNEL (DP)

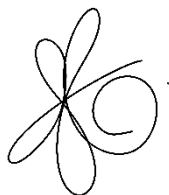
### Déclaration sur l'honneur

Je soussigné ..... *Davina Houquet* .....,  
déclare sur l'honneur que les renseignements fournis dans ce dossier sont exacts et que je  
suis l'auteur des réalisations jointes.

Fait à *Strasbourg* ..... le *10/02/2024* .....

pour faire valoir ce que de droit.

Signature :



## DOSSIER PROFESSIONNEL (DP)

### Documents illustrant la pratique professionnelle

*(facultatif)*

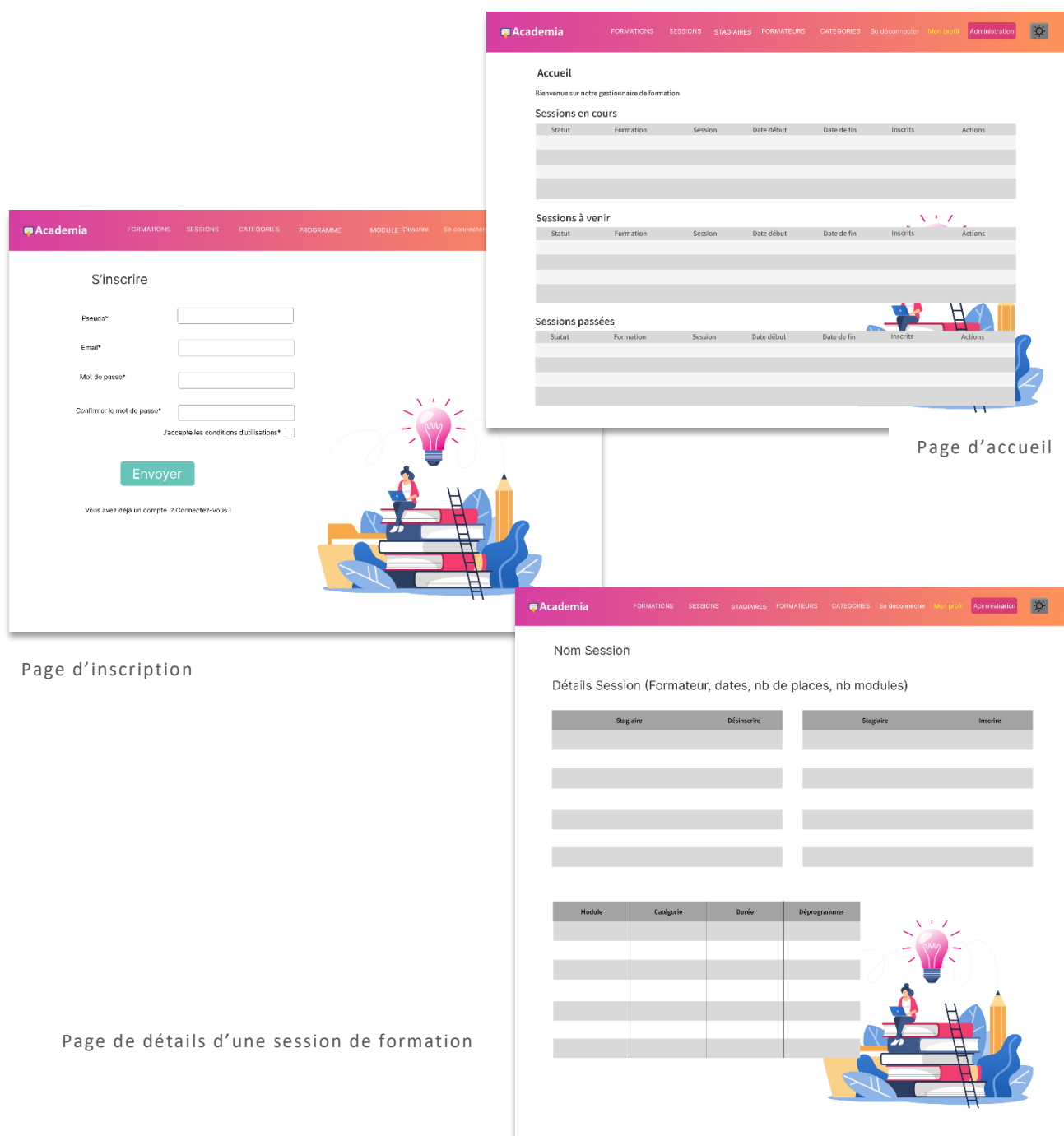
Intitulé
Cliquez ici pour taper du texte.

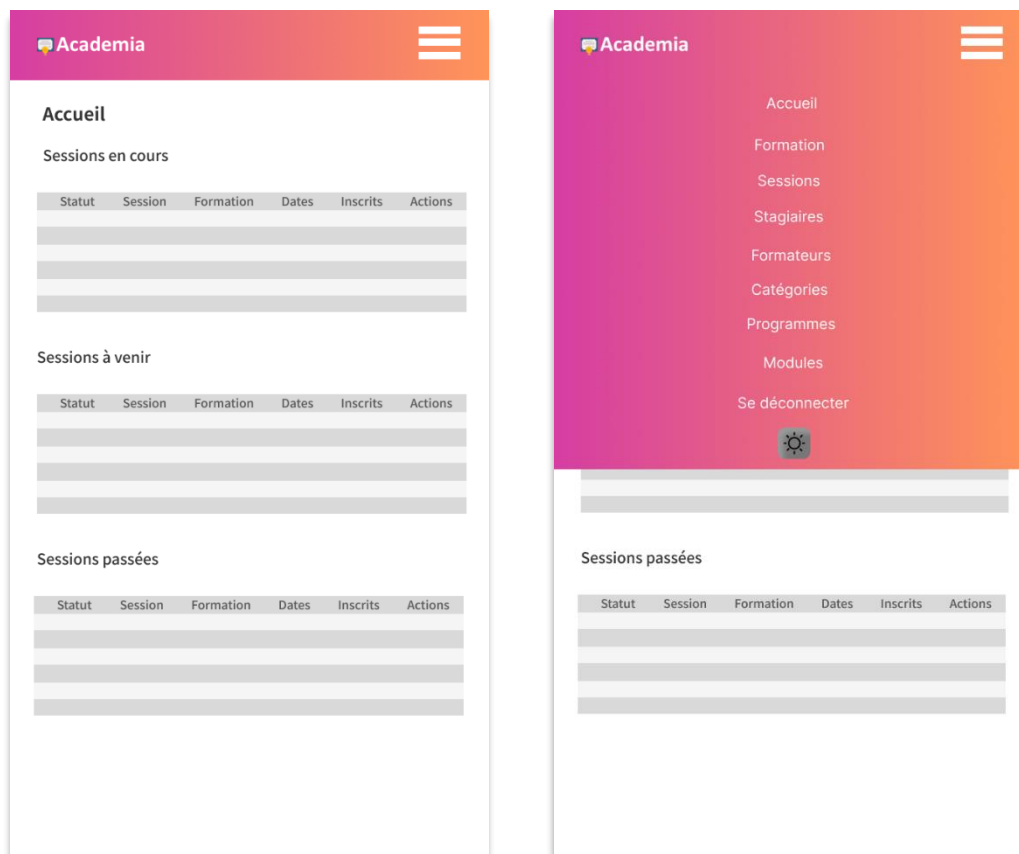
# ANNEXES

(Si le RC le prévoit)

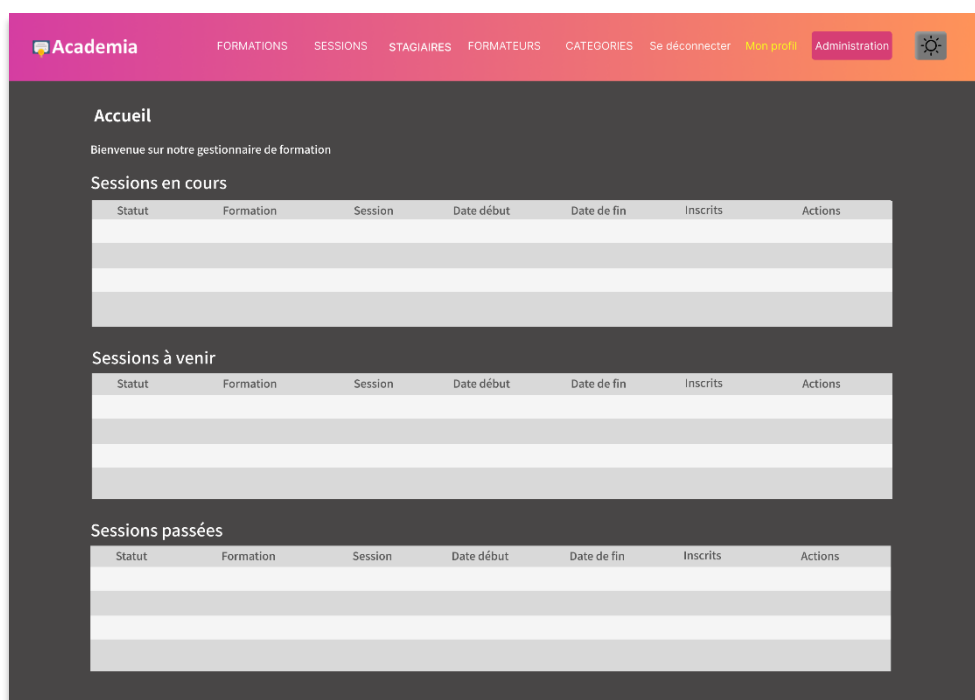
## Exercice de formation « Academia »

### Maquettage Figma



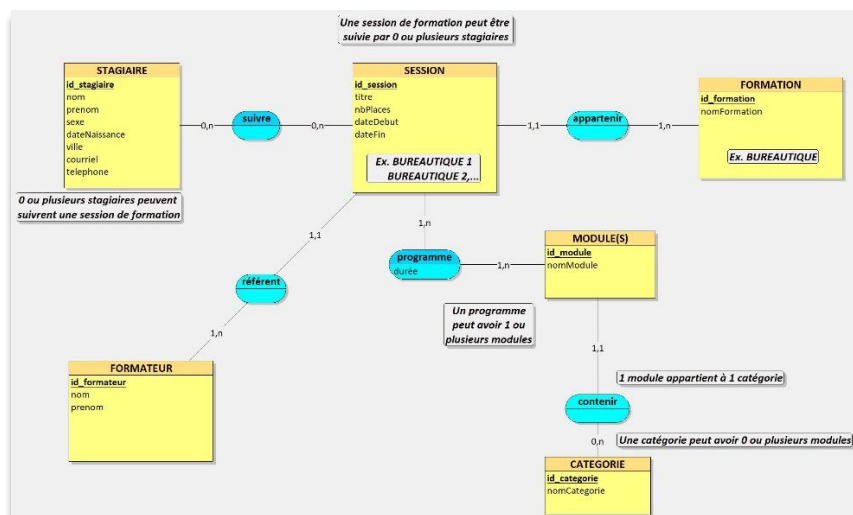


Page d'accueil version mobile

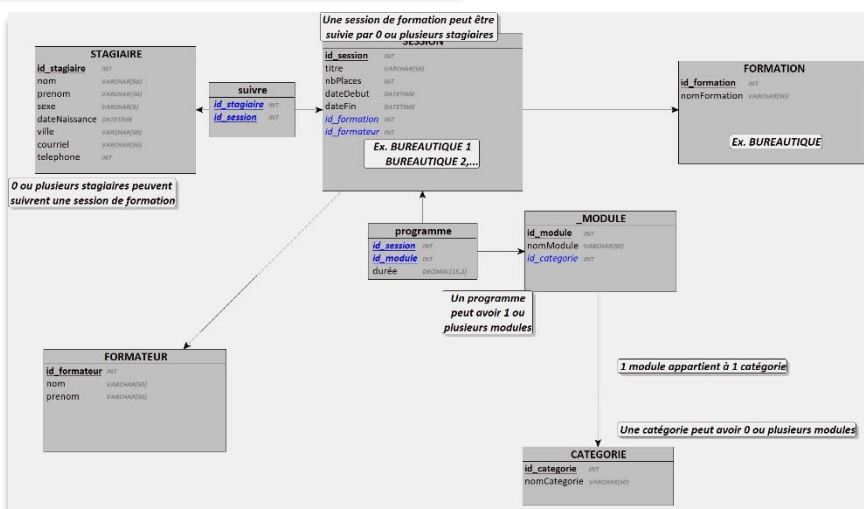


Page d'accueil utilisant le mode sombre

MCD/MLD



Modèle conceptuel de données



Modèle logique de données

## JavaScript

```

function toggleDarkMode() {
    var element = document.body;
    element.classList.toggle("dark-mode");

    // Vérifie si le mode sombre est activé
    var isDarkMode = element.classList.contains("dark-mode");

    // Stocke l'état du mode sombre dans le local storage
    if (isDarkMode) {
        localStorage.setItem("darkMode", "enabled");
    } else {
        localStorage.setItem("darkMode", "disabled");
    }
}

// Vérifie l'état du mode sombre au chargement de la page
window.onload = function() {
    var darkModeState = localStorage.getItem("darkMode");
    var element = document.body;

    // Si le mode sombre était activé lors de la dernière ses
    if (darkModeState === "enabled") {
        element.classList.add("dark-mode");
    }
};
    
```

Fonction JavaScript permettant d'alternier entre le mode clair et le mode sombre

Utilisation dans la vue grâce à un bouton

```

<button onclick="toggleDarkMode()" class="dark-mode"><span>&#9788;</span></button>
    
```

## Des requêtes SQL/DQL

Exemple de requête DQL

Dans la classe SessionRepository, permet de récupérer des objets sessions dont la date est supérieure à la date du jour

```
// /**
//  * @return Session[] Returns an array of Session objects
//  */
public function findSessionsAVenir(): array
{
    $now = new \DateTime();
    return $this->createQueryBuilder('s')
        ->andWhere('s.dateDebut > :now')
        ->setParameter('now', $now)
        ->orderBy('s.dateDebut', 'ASC')
        ->getQuery()
        ->getResult();
}
```

## SEO

templates > session > index.html.twig > div.container > table.table Ajout de block meta description

```
4 {% block meta_description %}Session{% endblock %}
```

Balises titres

templates > stagiaire > index.html.twig > div.container > h1

```
4 {% block meta_description %}Stagiaires{% endblock %}
5
6 {% block body %}
7     <div class="container">
8         <h1>Liste des stagiaires :</h1><br>
```

Balises strong

```
<h1>Bienvenue dans notre <strong>gestionnaire de sessions de formations</strong></h1><br>
```

## Sécurité

config > packages > ! security.yaml

```
1 security:
2     # https://symfony.com/doc/current/security.html#registering-the-user-hashing-passwords
3     password_hashers:
4         Symfony\Component\Security\Core\User\PasswordAuthenticatedUserInterface: 'auto'
5     # https://symfony.com/doc/current/security.html#loading-the-user-the-user-provider
6     providers:
7         # used to reload user from session & other features (e.g. switch_user)
8         app_user_provider:
9             entity:
10                 class: App\Entity\User
11                 property: email
```

Authentification

Autorisations

config > packages > ! security.yaml

```
37 # Easy way to control access for large sections of your site
38 # Note: Only the *first* access control that matches will be used
39 access_control:
40     - { path: ^/admin, roles: ROLE_ADMIN }
41     # - { path: ^/, roles: ROLE_USER }
```

```
{% if is_granted('ROLE_ADMIN') %}
<td><a href="{{path('edit_session', {'id' : session.id})}}"></a></td>
<td><a href="{{path('delete_session', {'id' : session.id})}}"></a></td>
{% endif %}
```



## Exercice réalisé dans le cadre du stage « Bloc Image »

### Maquettage Figma

Extrait de la maquette Figma fournie par le client



### Portions de code pouvant illustrer le design pattern

Sur une structure MVP, voici le contrôleur créé pour répondre à la demande, ainsi que la vue twig qui permet d'afficher l'élément.

```
1 <?php
2
3 namespace App\Controller\View;
4
5 use App\Service\FieldValueHelper;
6 use Ibexa\Bundle\Core\Controller;
7 use Ibexa\Core\Base\Exceptions\InvalidArgumentException;
8 use Ibexa\Core\MVC\Symfony\View\ContentView;
9
10 no usages ± davinahouquet +2
11 class BlockImageLinkController extends Controller
12 {
13     2 usages
14     private FieldValueHelper $fieldValueHelper;
15
16     no usages ± davinahouquet
17     public function __construct(
18         FieldValueHelper $fieldValueHelper
19     )
20     {
21         $this->fieldValueHelper = $fieldValueHelper;
22     }
23
24     /**
25      * @throws InvalidArgumentException
26      */
27     no usages ± davinahouquet +1
28     public function elementView(ContentView $view): ContentView
29     {
30         $linkedContent = $this->fieldValueHelper->getRelationFieldFirstContent($view->getContent(), $field: 'internal_link');
31
32         if ($linkedContent !== null) {
33             $view->addParameters(['linkedContent' => $linkedContent]);
34         }
35
36         return $view;
37     }
38 }
```

Contrôleur du bloc image

```

1  {# @var LinkedContent \Ibexa\Contracts\Core\Repository\Values\Content\Content #}
2  {% if LinkedContent is defined %}
3      {% set link = ibexa_path(LinkedContent) %}
4  {% elseif not ibexa_field_is_empty(content, 'external_link') %}
5      {% set link = ibexa_field_value(content, 'external_link') %}
6  {% endif %}
7
8  {% if ibexa_field_value(content, 'format').selection[0] == 1 %}
9      {% set class = 'full' %}
10     {% set alias = 'image_element_full1725_webp' %}
11 {% endif %}
12
13 <div class="image-container">
14     {% if link is defined and link is not empty %}
15         <a href="{{ link }}">
16             {{ ibexa_render_field(content, 'image', {
17                 'template': '@ibexadesign/fields/image.html.twig',
18                 'attr': {
19                     'class': class|default('demi'),
20                 },
21                 'parameters': {
22                     'alias': alias|default('image_element_demi863_webp'),
23                 }
24             }) }}
25         </a>
26     <div class="label-image">
27         <a href="{{ link }}" class="image-label">{{ ibexa_field_value(content, 'label') }}</a>
28     </div>
29 {% else %}
30     {{ ibexa_render_field(content, 'image', {
31         'template': '@ibexadesign/fields/image.html.twig',
32         'attr': {
33             'class': class|default('demi'),
34         },
35         'parameters': {
36             'alias': alias|default('image_element_demi863_webp'),
37         }
38     }) }}
39     <div class="label-image">
40         <a href="{{ link }}" class="image-label">{{ ibexa_field_value(content, 'label') }}</a>
41     </div>
42 {% endif %}

```

Vue twig du bloc image

## Responsive Design

```

37
38 .full {
39     width: 100%;
40     max-height: 500px;
41     object-fit: cover;
42
43     img {
44         width: 100%;
45     }
46 }
47
48 .demi {
49     width: 50%;
50
51     img {
52         max-width: 100%;
53         height: 400px;
54         object-fit: cover;
55     }
56 }
57
58 @media (device-width >= 769px) {
59     .demi {
60         width: 50%;
61         height: auto;
62
63         img {
64             width: 100%;
65             max-height: 400px;
66             object-fit: cover;
67         }
68     }
69
70     .full {
71         min-width: 100%;
72         max-height: 500px;
73         object-fit: cover;
74     }
75 }
76
77 @media (device-width >= 100px) and (device-width <= 768px) {
78     .demi {

```

SCSS du bloc image

Utilisation de media queries pour un design responsive qui permet d'afficher les images sur des écrans de différentes tailles

## Maquettage Figma



Maquette du bloc flux rss

## Design pattern



Contrôleur du bloc flux rss

```

1 <div class="block-flux-rss-container">
2   <h2>{{ ibexa_content_name(content) }}</h2>
3   {% if items is defined %}
4     {% for item in items %}
5       <div class="title-date">
6         <a href="{{ item.link }}" target="_blank"><h3>{{ item.title }}</h3></a>
7         <p>Publié le {{ item.pubDate|date("d/m/Y") }}</p>
8       </div>
9       <div class="description-rss-link">
10        {{ item.description }}
11      </div>
12    {% endfor %}
13  {% else %}
14    <p>Flux RSS Indisponible</p>
15  {% endif %}
16 </div>
17

```

Vue twig du bloc flux rss

## Bonnes pratiques

Utilisation de service pour récupérer les valeurs des entrées du back office selon leur type

```

1 usage  ± davina.houquet
157 public function getUrlFieldValue(Content $content, string $field, string $default = null): ?string
158 {
159     if ($this->isFieldEmpty($content, $field)) {
160         return $default;
161     }
162     /** @var UrlFieldValue $fieldValue */
163     $fieldValue = $content->getFieldValue($field);
164     return $fieldValue->link;
165 }
166
167
168 1 usage  ± davina.houquet
168 public function getIntegerFieldValue(Content $content, string $field, int $default = null): ?int
169 {
170     if ($this->isFieldEmpty($content, $field)) {
171         return $default;
172     }
173     /** @var IntegerFieldValue $fieldValue */
174     $fieldValue = $content->getFieldValue($field);
175     return $fieldValue->value;
176 }
177 }
178

```

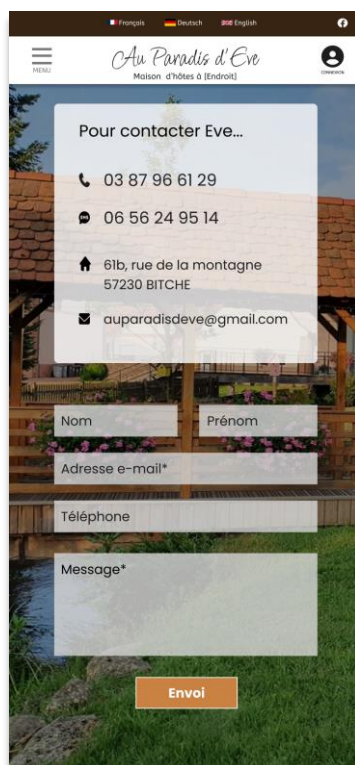
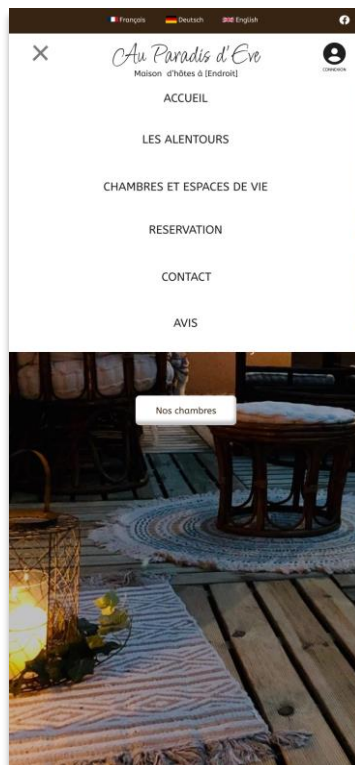
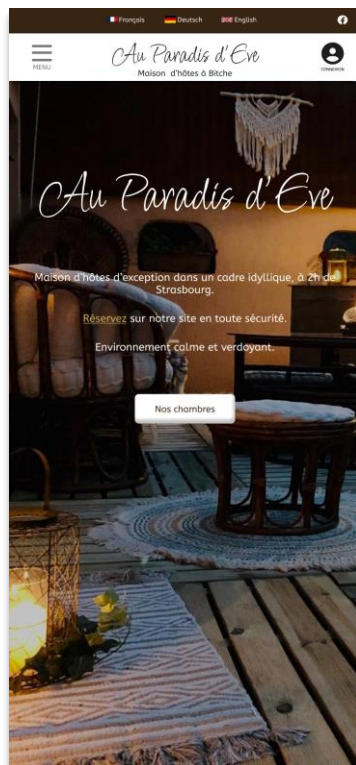
Service/FieldValueHelper.php

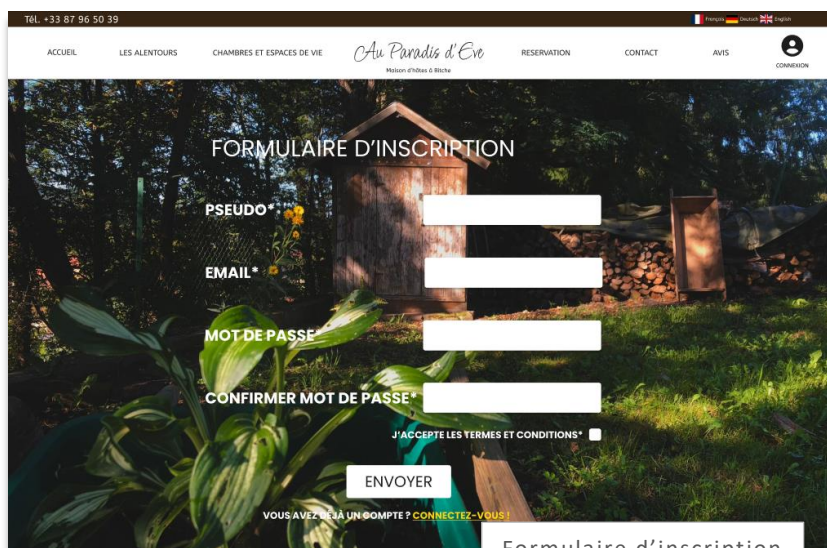
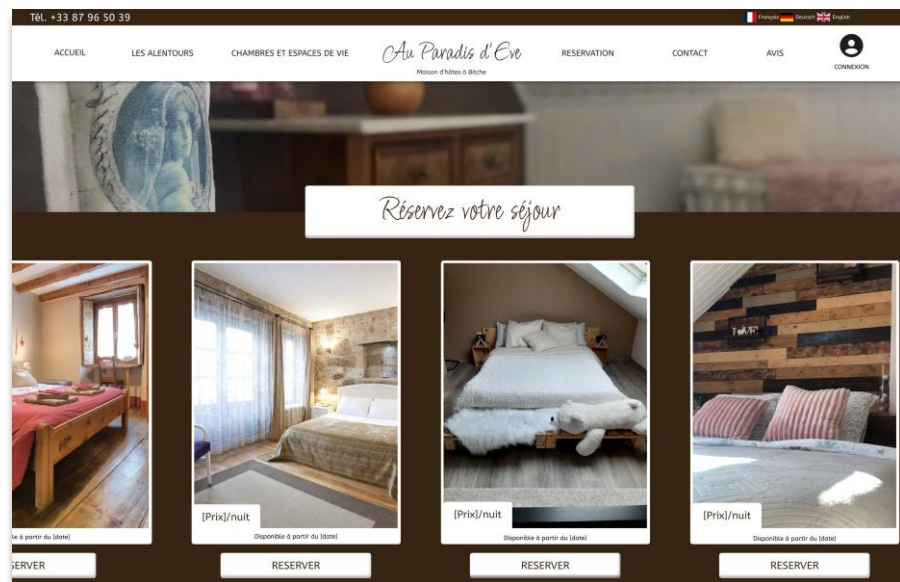
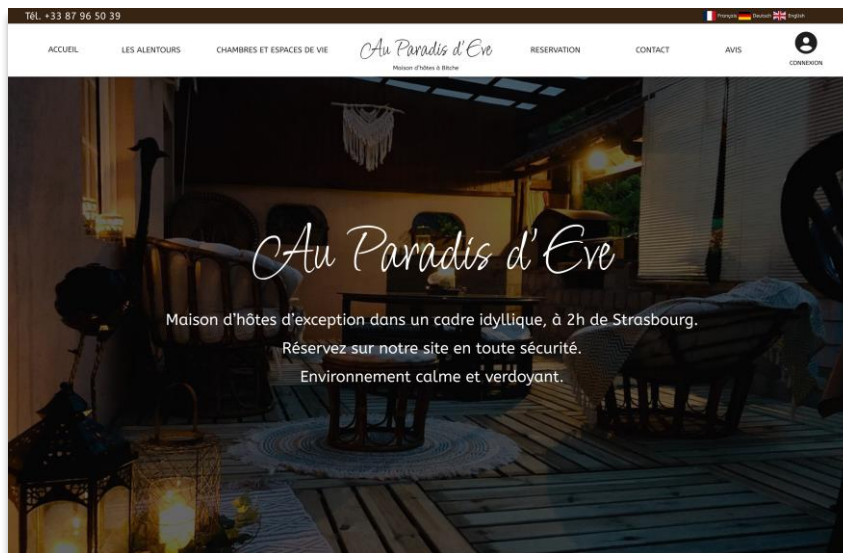
## Projet de fin de formation « Au Paradis d'Eve »

### Maquettage Figma

Page d'accueil sur mobile

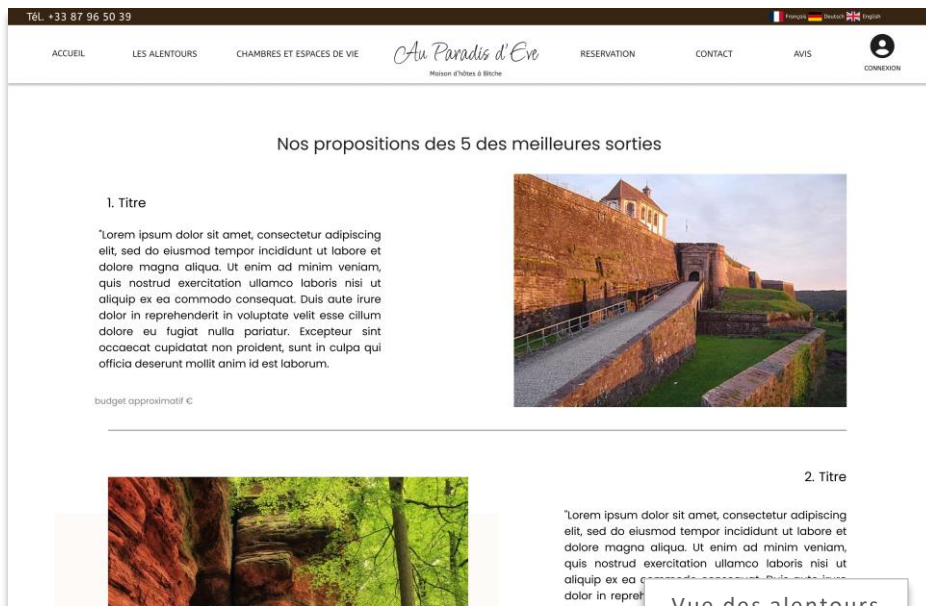




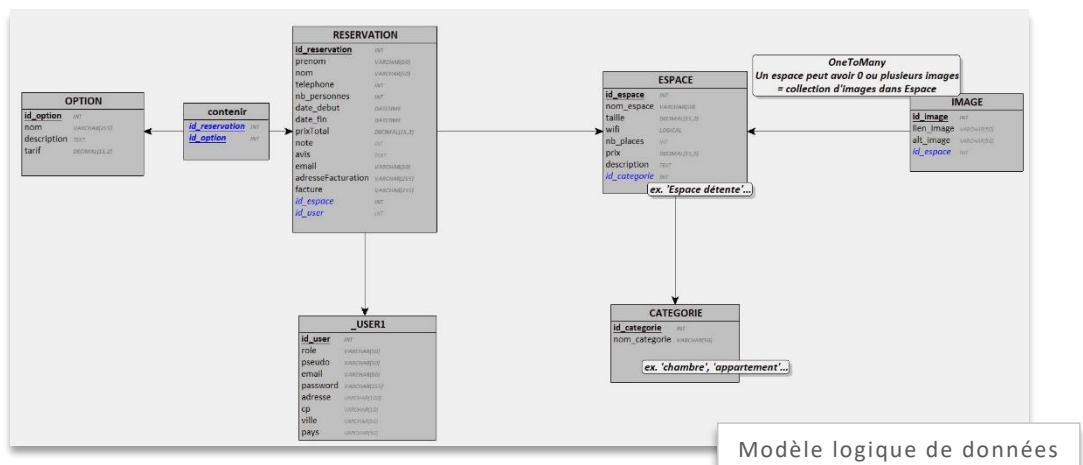
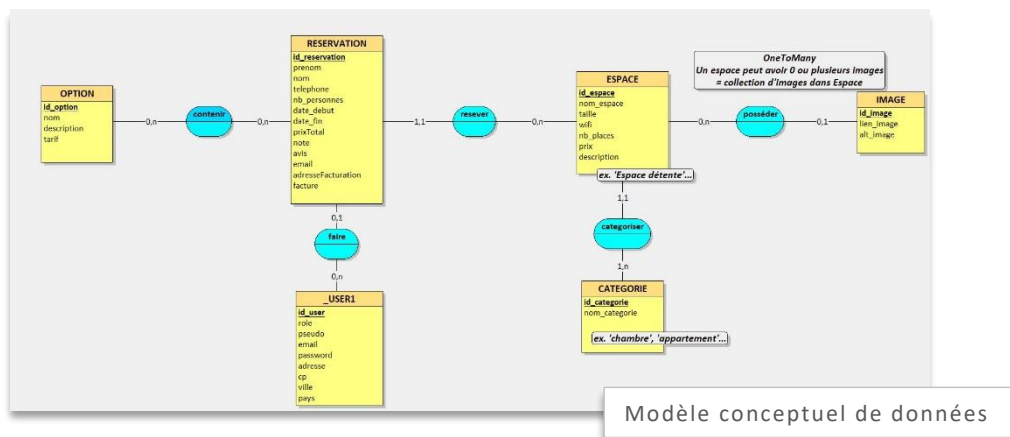


Formulaire d'inscription





## MCD/MLD



## Des éléments JavaScript

```
// Menu burger
function toggleBurgerMenu() {
  var menu = document.querySelector('nav ul');
  var icon = document.getElementById('burger-icon');

  menu.classList.toggle('active');

  // Change l'icône en fonction de l'état du menu
  if (menu.classList.contains('active')) {
    icon.innerHTML = '<span class="material-symbols-outlined">close</span>';
  } else {
    icon.innerHTML = '<span class="material-symbols-outlined">menu</span>';
  }
}
```

Fonction JavaScript permettant de faire apparaître le menu burger

Icône présente dans la barre de navigation, qui appelle la fonction

```
<div class="burger-icon" onclick="toggleBurgerMenu()">
  <span class="material-symbols-outlined">
    menu
  </span>
</div>
```

## Des requêtes SQL/DQL

Exemple de requête DQL :



```
public function findByCategorie($categorieId): array
{
    return $this->createQueryBuilder('e')
        ->andWhere('e.categorie = :categorieId')
        ->setParameter('categorieId', $categorieId)
        ->orderBy('e.id', 'ASC')
        ->setMaxResults(10)
        ->getQuery()
        ->getResult()
;
}
```

Exemple de requête DQL

Dans la classe EspaceRepository, permet de récupérer des objets espace selon leur catégorie

Exemple d'utilisation dans le détail d'un espace selon sa catégorie

```
public function show_espace(Espace $espace, EspaceRepository $espaceRepository, EntityManagerInterface $entityManager): Response
{
    $repository = $entityManager->getRepository(Espace::class);
    $chambre = $repository->findByCategorie(1);
}
```

## Search Engine Optimization

```
<h1 class="licorice-title"><strong>Au Paradis d'Eve</strong></h1>
```

```
<h2>Nos chambres</h2>
```

```
<h3>Les réservations à rembourser partiellement</h3>
```

```
<meta name="description" content="{% block meta_description %}Accueil - Au Paradis d'Eve{% endblock %}">
```

```
{% block meta_description %} Remplissez vos coordonnées - Coordonnées de réservation {% endblock %}
```

```
{% block meta_description %} Nouvelle réservation - formulaire de réservation {% endblock %}
```

*Hiérarchie des titres, balises strong*

[Facebook](#) [Instagram](#) [Tiktok](#)

*Réseaux sociaux*

```
/* Breakpoint 1 : Smartphone */
@media (min-width: 360px) and (max-width: 767px) {
```

```
/* Breakpoint 2 : Tablette Portrait - PC avec naviga
@media (min-width: 768px) and (max-width: 999px) {
```

```
/* Breakpoint 3 : Tablette Paysage - PC Laptop - Tout
@media (min-width: 1000px) and (max-width: 1279px) {
```

*Responsive, media queries*

## Éléments liés à la sécurité

```
<input type="hidden" name="_csrf_token" value="{{ csrf_token('authenticate') }}">
```

```
new CsrfTokenBadge('authenticate', $request->request->get('_csrf_token')),
```

Cross Site Request Forgery security

```
'constraints' => [
    new NotBlank([
        'message' => 'Veuillez entrer un mot de passe',
    ]),
    new Length([
        'min' => 12,
        'minMessage' => 'Votre mot de passe doit contenir 12 caractères minimum',
        'max' => 4096,
    ]),
    new Regex([
        'pattern' => '/^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-z\d@$!%*?&]+$',
        'message' => 'Le mot de passe doit contenir au moins une minuscule, une majuscule, un chiffre et un caractère spécial.'
    ]),
],
]
```

\$2y\$ 10\$ 6z7GKa9kpDN7KC3lCW1Hi. fd0/to7Y/x36WUKNP0IndHdkdR9Ae3K



Protection contre l'attaque par dictionnaire : contraintes et hachage de mot de passe

```
public function buildForm(FormBuilderInterface $builder, array $options): void
{
    $builder
        ->add('id_sortie', HiddenType::class)
        ->add('nom_sortie', TextType::class, [ ... ])
        ->add('description_sortie', TextareaType::class, [ ... ])
        ->add('tarif_sortie', NumberType::class, [ ... ])
        ->add('image_sortie', UrlType::class, [ ... ])
        ->add('credit', TextType::class, [ ... ])
        ->add('save', SubmitType::class, [ ... ]);
}
```

Faible Cross Site Scripting – Type de données dans un formulaire Symfony

```
'constraints' => [
    new All([
        'constraints' => [
            new File([
                'maxSize' => '5M', // Limite la taille à 5 Mo
                'mimeType' => ['image/jpeg', 'image/png', 'image/webp'],
                'mimeTypeMessage' => 'Veuillez télécharger une image valide (JPEG, PNG, WEBP)',
            ]),
        ],
    ]),
],
),
```

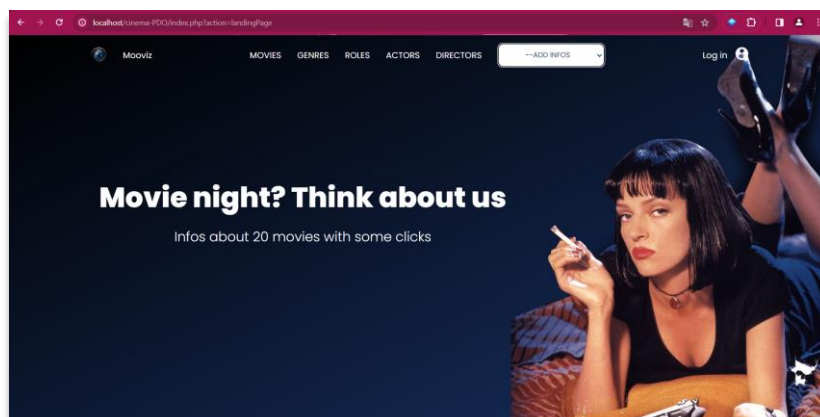
```
// Upload d'images spécifiquement lié à l'entité Espace
public function uploadImages(array $imageFiles, string $altImage, $espace)
{
    foreach ($imageFiles as $imageFile) {

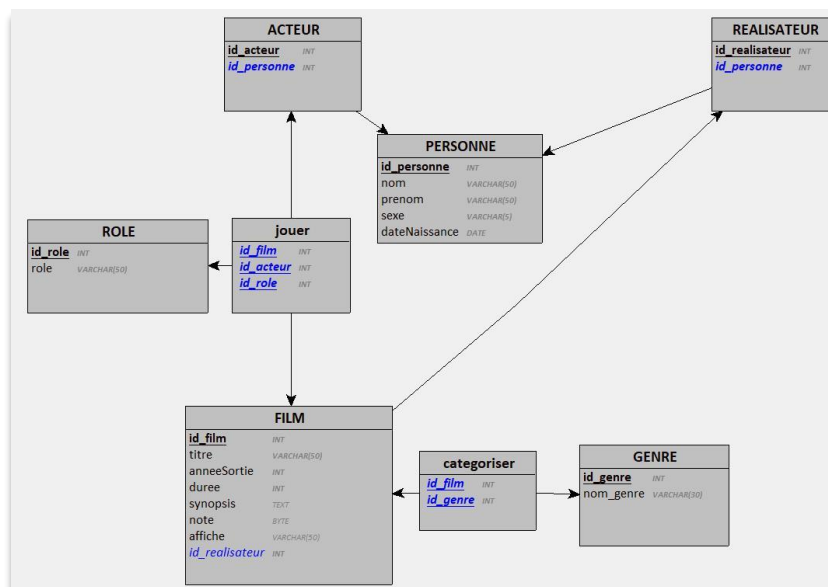
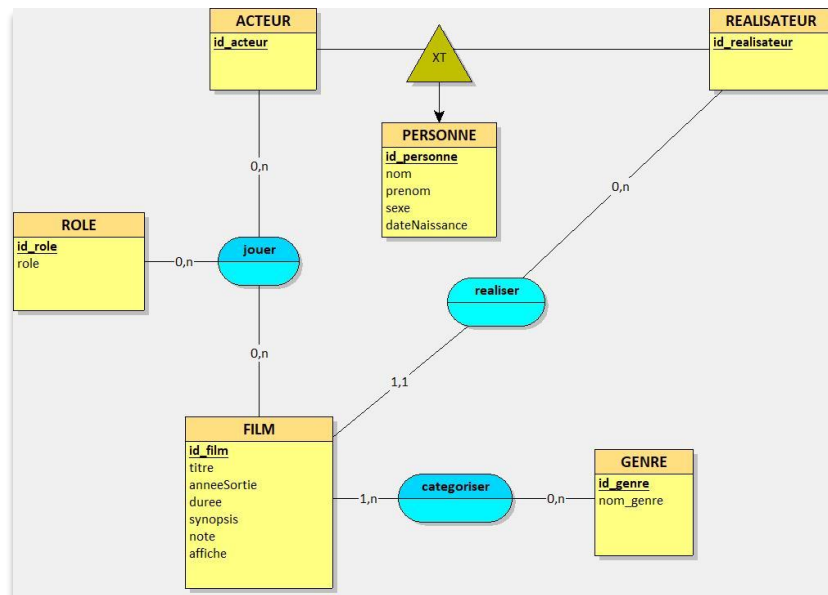
        $newFileName = md5(uniqid()) . '.' . $imageFile->guessExtension();
        $imageFile->move($this->parameterBag->get('images_directory'), $newFileName);
        $image = new Image();
        $image->setLienImage($newFileName);

        $image->setEspace($espace);
        $image->setAltImage($altImage);
        $espace->addImage($image);
        $this->entityManager->persist($image);
        $this->entityManager->persist($espace);
    }
    $this->entityManager->flush();
}
```

*Faillie upload - contraintes et identifiant unique*

## Exercice Cinéma « Mooviz »





```

3 namespace Model;
4
5 abstract class Connect { //La classe est abstraite car on n'instanciera jamais
    la classe Connect puisqu'on aura seulement besoin d'accéder à la méthode
    "seConnecter"
6
7     const HOST = "db";
8     const DB = "cinema_v2";
9     const USER = "root";
10    const PASS = "pass";
11
12    public static function seConnecter() { //on se contente de déclarer la
        connexion à la base de données
13        try {
14            return new PDO("mysql:host=".self::HOST.";dbname=".self::DB,
                self::USER,self::PASS);
15        } catch(PDOException $ex) {
16            return $ex->getMessage();
17        }
18    }
19 }

```

```

1 docker-compose.yml
2 version: '3.8'
3 services:
4     php:
5         image: php:8.2-apache
6         container_name: php82
7         ports:
8             - 82:80
9         volumes:
10            - ../var/www/html
11        build:
12            context: .
13            dockerfile: Dockerfile
14    db:
15        image: mysql:8.0
16        container_name: mysql80
17        command: --default-authentication-plugin=mysql_native_password
18        ports:
19            - 3307:3306
20        environment:
21            MYSQL_DATABASE: cinema_v2
22            MYSQL_USER: root
23            MYSQL_ROOT_PASSWORD: pass
24        volumes:
25            - ../mysql:/var/lib/mysql
26    phpmyadmin:
27        image: phpmyadmin/phpmyadmin
28        container_name: phpmyadmin
29        environment:
30            PMA_ARBITRARY: 1
31            PMA_HOST: db
32            PMA_USER: root
33            PMA_PASSWORD: pass
34        ports:
35            - 8888:80

```

#### Dockerfile

```

1 FROM php:8.2-apache
2
3 RUN apt-get update && apt-get upgrade -y
4 RUN docker-php-ext-install mysqli pdo pdo_mysql && docker-php-ext-enable mysqli pdo_mysql
5
6 EXPOSE 80

```

## Exercice NoSQL – MongoDB

```
> db.clients.insertOne({ name: "SANCHEZ", ville:"Seattle"})
< {
  acknowledged: true,
  insertedId: ObjectId('65c0b953641178b51d131ae3')
}
```

```
_id: ObjectId('65c0ba13641178b51d131ae4')
name: "SANCHEZ"
ville: "Seattle"
```

```
> db.clients.deleteMany({})
< {
  acknowledged: true,
  deletedCount: 1
}
```

```
{ } dbcommerce.clients.json > ...
1  [{
2    "_id": {
3      "$oid": "65c0ba13641178b51d131ae4"
4    },
5    "name": "SANCHEZ",
6    "ville": "Seattle"
7  },
8  {
9    "_id": {
10     "$oid": "65c0ba64641178b51d131ae5"
11   },
12   "name": "SMITH",
13   "ville": "Seattle"
14 },
15 {
16   "_id": {
17     "$oid": "65c0ba76641178b51d131ae6"
18   },
19   "name": "GRIFFIN",
20   "ville": "Quahog"
21 }]
```

## Certificat SSL et HTTPS

SSL: SECURE SOCKETS LAYER

Le SSL est une technologie standard de sécurisation des connexions Internet par le chiffrement des données transitant entre un navigateur et un site web (ou entre deux serveurs). Durant leur transfert, les données (personnelles, financières, etc.) sont ainsi protégées des hackers qui ne peuvent ni les voir ni les détourner.

HTTPS: HyperText Protocol Secure

Lorsqu'un site web est sécurisé par un certificat TLS/SSL, son URL commence par la mention HTTPS. Pour connaître les détails du certificat (Autorité de certification ou AC émettrice, raison sociale du propriétaire du site, etc.), les internautes peuvent cliquer sur le cadenas dans la barre d'adresse du navigateur.

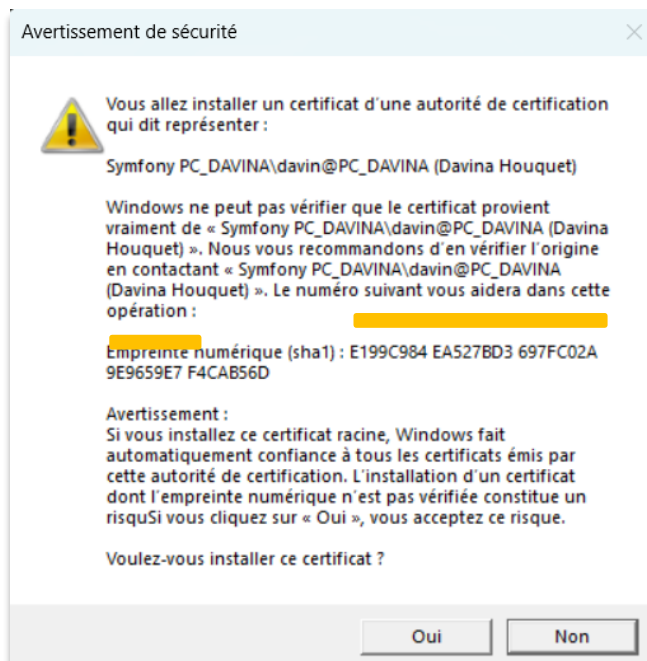


Mise en place

Il est possible d'effectuer cette mise en place en effectuant la commande `symfony.exe server :ca :install`.

```
PS C:\laragon\www\Au Paradis d Eve> symfony.exe server:ca:install
You might have to enter your root password to install the local Certificate Authority certificate
The local CA is now installed in the system trust store!
Generating a default certificate for HTTPS support
```

```
[OK] The local Certificate Authority is installed and trusted
```



Un avertissement apparaîtra alors pour confirmer l'installation du certificat. Une empreinte numérique est alors fournie.

Et lorsqu'on redémarre le serveur, on peut constater que le lien commence désormais par `https`.

```
[OK] Web server listening
The Web server is using PHP CGI 8.1.10
https://127.0.0.1:8000
```

```
https://127.0.0.1:8000/home
```