

# DOSSIER PROFESSIONNEL (DP)

*Nom de naissance* ▶ WETZSTEIN  
*Nom d'usage* ▶ WETZSTEIN  
*Prénom* ▶ Anthony  
*Adresse* ▶ 9 rue de la Hohwarth 67540 OSTWALD

## 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.

- ▶ Intitulé de l'exemple n° 1 Ad Temporis Finem ..... p. 5
- ▶ Intitulé de l'exemple n° 2 Cinéma ..... p. 8
- ▶ Intitulé de l'exemple n° 3 Sessions..... p. 10

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

- ▶ Intitulé de l'exemple n° 1 Ad Temporis Finem ..... p. 13
- ▶ Intitulé de l'exemple n° 2 Forum..... p. 16
- ▶ Intitulé de l'exemple n° 3 Cinéma ..... p. 18

**Titres, diplômes, CQP, attestations de formation** *(facultatif)* p. 21

**Déclaration sur l'honneur** p. 22

**Documents illustrant la pratique professionnelle** *(facultatif)* p. 23

**Annexes** *(Si le RC le prévoit)* p. 24

# **EXEMPLES DE PRATIQUE PROFESSIONNELLE**

## Activité-type 1

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

Exemple n°1 ► *Projet Ad Temporis Finem*

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

1. Rédaction d'un cahier des charges pour définir un **Minimum Viable Product (MVP)**.
2. Documentation sur le **SEO** et le **RGPD** : ces notions influencent la construction de l'agencement des pages et le contenu des formulaires.
3. Création d'une charte graphique. Adopter des solutions permettant une bonne **accessibilité** et **ergonomie**. Tester les contrastes avec des outils comme [www.whocanuse.com](http://www.whocanuse.com), vérifier la lisibilité de la typographie pour des personnes malvoyantes ou dyslexiques.
4. Réalisation d'une **maquette** fidèle à l'aspect final sans les éléments interactifs (« **mockup** ») : créer des versions pour les appareils mobiles, les écrans de type tablette et les écrans de bureau et ordinateurs portables. L'outil **Figma** (application web de graphisme) m'a permis de réaliser cette tâche.
5. Utilisation du **Framework Symfony** : la partie **front-end** de l'application est la partie de l'application visible aux utilisateurs. Symfony contient **Twig**, un gestionnaire de **template** avec des fonctions spécifiques (« filters », filtres Twig). Un template est un élément réutilisable, intégré dans un **layout** (agencement de la page).
6. Création d'un ou plusieurs **layout** : dans ce projet un layout spécifique pour la partie jeu, un layout spécifique pour toutes les autres pages. Utilisation du **HTML5**, et **syntaxe Twig** avec blocs.
7. Créer le **template** de la page : utilisation de **HTML5**. Ajouter des **classes** et/ou des **identifiants** (id) avec noms pertinents à des éléments clefs. S'il y a des images, utiliser l'attribut « alt » pour donner du texte de remplacement aux images pour permettre la lecture via un lecteur d'écran. Pour augmenter le référencement, utiliser des balises **sémantiques** pertinentes comme **main** pour le contenu principal, **strong** pour mettre du contenu en valeur, **aside** pour le contenu accessoire, **nav** pour la navbar.
8. Suivre les bonnes **recommandations SEO** (Titres pertinents, meta description, keywords, liens compréhensibles par un humain, sitemap, réseaux sociaux pour le maillage externe)
9. Création d'un feuillet de **style en cascade (CSS)** pour la page. Utilisation de sélecteurs différents pour cibler les classes et id stratégiques et leur attribuer des propriétés spécifiques.
10. Agencer la page comme la maquette le prévoit : rendre le site « **responsive** » en utilisant des **requêtes media (media queries)** pour définir le comportement de certains éléments de la page selon la taille d'écran en définissant des **points de rupture ou breakpoints** choisis stratégiquement.
11. Ajouter de l'interactivité avec **Javascript** via l'utilisation de la balise **<script>**. Utilisation de **jQuery** via CDN ou utilisation locale. Sur certaines pages notamment sur le jeu, utiliser des **requêtes asynchrones — AJAX —** pour modifier des éléments de la page sans les rafraîchir.
12. Si la page réalisée est un formulaire, faire apparaître les champs provenant d'un **FormType** de Symfony pour valider des données et échapper les caractères dangereux. Si le formulaire est celui d'**inscription** vérifier que le **champ du mot de passe est répété** et qu'un lien vers les conditions générales d'utilisation ainsi qu'une **checkbox** à cocher sont présents pour **vérifier le consentement de l'utilisateur** (respect d'un volet du RGPD). Toujours pour l'inscription ne mettre

# DOSSIER PROFESSIONNEL (DP)

que des champs d'entrées pour un pseudonyme et un email, le pseudo étant nécessaire au fonctionnement du jeu et l'email pour la réinitialisation du mot de passe. C'est le respect de **la minimisation des données**. Créer une page Terms of Service qui permet de dire pourquoi on en a besoin (finalité des données).

## 2. Précisez les moyens utilisés :

Outil en ligne de graphisme : **Figma**.

Utilisation de services de test d'accessibilité : **<https://www.whocanuse.com/>**

Utilisation d'un langage de balisage : **HyperText Markup Language (version 5)**.

Utilisation d'un langage de présentation : **Cascading Style Sheets (CSS)**.

Utilisation d'un Framework : **Symfony 6.3**.

Utilisation d'un moteur de template : **Twig 3**.

Utilisation d'un langage de manipulation du Document Object Model (DOM) pour l'interactivité : **Javascript**.

Utilisation d'une librairie Javascript : **JQuery**.

## 3. Avec qui avez-vous travaillé ?

# DOSSIER PROFESSIONNEL (DP)

Formateurs Elan Formation

## 4. Contexte

Nom de l'entreprise, organisme ou association		<b>Elan Formation</b>
Chantier, atelier, service	▶	Equipe Développeur web et web mobile
Période d'exercice	▶	Du 04/09/2023 au 16/02/2024

## 5. Informations complémentaires (facultatif)

Projet de présentation d'examen

## Activité-type 1

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

Exemple n°2 ► Exercice Cinéma

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

1. Création d'une maquette pour l'exercice : type **mock-up**.
2. Utilisation d'un Framework propriétaire Elan Formation pour se préparer à l'utilisation de Symfony.
3. Création de pages avec langage **HTML**. Créer des **éléments réutilisables : templates**, les intégrer dans une **mise en page : « layout »**. Utilisation des balises sémantiques.
4. Ajout de style pour la page via **CSS**.
5. Pour les pages d'inscription et de connexion : création d'un formulaire avec les champs required – c'est n'est pas une protection, donc le formulaire pointe vers une page de vérification de données.
6. Utilisation de Javascript pour la création **d'un mode clair et d'un mode sombre** pour l'habillage du site.

### 2. Précisez les moyens utilisés :

Environnement de développement : **Laragon**.

Langage de balisage : **HTML5**.

Langage de présentation : **CSS**.

Langage de manipulation du DOM : **Javascript**.



# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

Equipe Elan Formation

## 4. Contexte

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

Chantier, atelier, service ► Equipe Développeur web et web mobile

Période d'exercice ► Du 13/03/2023 au 02/06/2023

## 5. Informations complémentaires (facultatif)

Cliquez ici pour taper du texte.

## Activité-type 1

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

Exemple n°3 ► Exercice sessions

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

1. Lecture du **cahier des charges fourni**. Cet exercice impose un minimum de fonctionnalité à l'application et pour la partie front nécessite une interface type intranet et la présence d'un moyen de passer d'un mode sombre à un mode clair ou vice-versa.
2. Création d'une **charte graphique**. Adopter des solutions permettant une bonne **accessibilité** et ergonomie. Tester les contrastes avec des outils comme [www.whocanuse.com](http://www.whocanuse.com), vérifier la lisibilité de la typographie pour des personnes malvoyantes.
3. Réalisation d'une maquette fidèle à l'aspect final sans les éléments interactifs (« **mockup** »). Création de vues bureau uniquement dans le cadre de ce projet. L'outil **Figma** (application web de graphisme) m'a permis de réaliser cette tâche.
4. Utilisation du **Framework Symfony** et des fonctionnalités de **Twig**.
5. Création d'un **layout** spécifique pour la page d'accueil et d'un **layout** pour toutes les autres pages, avec la syntaxe **Twig**.
6. Créer le **template** de la page : utilisation de **HTML5 & Twig**. Ajouter des classes et/ou des identifiants (id) avec noms pertinents à des éléments clefs. S'il y a des **images**, utiliser l'**attribut « alt »** pour créer du texte remplaçant l'image pour les lecteurs d'écran.
7. Création d'un **feuille de style en cascade (CSS)** pour la page. Utilisation de **sélecteurs différents** pour cibler les classes et id stratégiques et leur attribuer des propriétés spécifiques.
8. Créer un site responsive efficace avec l'utilisation du **responsive design** en utilisant des **requêtes media (media queries)** pour définir le comportement de certains éléments de la page selon la taille d'écran. Création d'un « **menu burger** » pour les petits écrans.
9. Ajouter de l'interactivité avec **Javascript** via l'utilisation de **<script>** : script spécial pour pouvoir passer du mode sombre au clair avec des **classList.toggle()**.
10. Une page comporte une fonctionnalité spéciale : **placer des sessions physiques sur une carte : consultation et utilisation de l'API Leaflet**.
11. Si la page réalisée est un formulaire, utilisation des FormTypes de Symfony pour valider des données et échapper les caractères dangereux.

### 2. Précisez les moyens utilisés :

# DOSSIER PROFESSIONNEL (DP)

Outil en ligne de graphisme : **Figma**.

Utilisation de services de test d'accessibilité : <https://www.whocanuse.com/>

Utilisation d'un langage de balisage : **HyperText Markup Langage (version 5)**.

Utilisation d'un langage de présentation : **Cascading Style Sheets (CSS)**.

Utilisation d'un Framework : **Symfony 6.3**.

Utilisation d'un moteur de template : **Twig 3**.

Utilisation d'un langage de manipulation du Document Object Model (DOM) pour l'interactivité : **Javascript**.

Utilisation d'une librairie Javascript : **Leaflet**.

## 3. Avec qui avez-vous travaillé ?

Equipe Elan Formation

## 4. Contexte

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

Chantier, atelier, service ► **Equipe Développeur web et web mobile**

Période d'exercice ► Du **04/09/2023** au **16/02/2024**

## 5. Informations complémentaires (facultatif)

## DOSSIER PROFESSIONNEL (DP)

Cliquez ici pour taper du texte.

## Activité-type 2

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

Exemple n°1 ► *Ad Temporis Finem*

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

1. Lecture de la documentation RGPD.
2. Lancement serveur **MySQL**.
3. Paramétrage **.env** Symfony et connexion à la **BDD** avec le **SGBD**.
4. Utilisation de la méthode **Merise** pour **modéliser** les données du projet. Création d'un **MCD (Modèle Conceptuel de Données)** via l'utilisation du logiciel **Looping**.
5. Définir les liens des **entités** entre elles (**Un à Plusieurs, Plusieurs à Un, Plusieurs à Plusieurs**, etc.).
6. Création du **MLD (Modèle Logique de Données)** correspondant via l'utilisation du logiciel **Looping**. Définir les **types de données** et si elles peuvent être nullable ou non.
7. Choix d'un framework pour mieux gérer le projet, **séparer les responsabilités**. Choix du framework **Symfony** qui suit un **design pattern architectural**. Le code du projet repose sur le paradigme de **programmation orienté objet (POO)**. Nous allons donc créer des classes qui sont composées **d'attributs et de méthodes**. Les propriétés de la classe sont **encapsulées** (status **private**), elles ne sont pas accessibles directement mais via des **setters** et des **getters** de l'**entité**. Dans un ce cadre de travail, les **controller** que nous allons créer **hériteront** des méthodes et propriétés de la classe native **AbstractController** des controller Symfony. Les objets que nous allons créer seront les **instances** de cette classe.
8. Lancement console Symfony : création des entités via la commande **symfony console make:entity**
9. Paramétrage des entités, utiliser des données pertinentes.
10. Création de la Base de Données Objet avec **Doctrine (Object-Relational Mapping)** et **migration** pour la lier avec la base de données MySQL.
11. Création de **controller** via **symfony CLI** (command line interface) : « **symfony console make :controller** ».
12. Définir la route du **controller** et son nom. Y ajouter des paramètres additionnels en annotation si besoin, souvent un ou plusieurs **Repository** pour récupérer différents types de données.
13. Créer des requêtes **Doctrine Query Langage** personnalisée ou utiliser une méthode générale pour récupérer des données. Les **repository** sont la couche **Model** dans le Framework Symfony. Passer ces données à la vue via le Controller. Ces requêtes **DQL** sont **préparées** : elles sont écrites avec des **placeholder** pendant l'étape de préparation, puis pendant l'étape de **compilation** on utilise des **setParameter** pour valider les types de données puis on exécute la requête.
14. Si la **vue** est un formulaire utilisation de **FormTypes Symfony** : formulaires paramétrables qui **restreignent** les champs à un type de données, avec ajout d'autres **contraintes** paramétrables comme **NotBlank, Regex ou Range**.
15. Pour l'hébergement de fichiers, créer un service chargé de déplacer les fichiers hébergés dans un dossier uploads et de les renommer en leur attribuant un **uniqID**. Dans le **formType**, rajouter des **contraintes** sur le **poids**, la **taille** et le **MIME type** du fichier (c-a-d son extension) : autoriser uniquement les formats image.
16. De base, les FormType Symfony génèrent un **jeton CSRF** pour se protéger de la faille du même nom. Vérifier s'il est bien présent avec des tests, sinon l'ajouter manuellement.

17. Pour l'inscription, utilisation de **maildev**, un service de messagerie locale pour demander la validation du compte via un clic sur un lien.
18. Utilisation de **bundle** (collections de code et de fichiers écrits pour être utilisés par Symfony) additionnel comme **Reset-password-bundle**. Utilisation de la documentation pour l'installation du bundle.
19. Le **make bundle** de symfony permet de créer un ensemble de controller et de formType paramétrables qui permettent de faire appel aux fonctions de **hachage fort** lors de l'inscription. Dans le formType d'enregistrement utiliser les **Contraintes** pour forcer une **Regular Expression** sur le mot de passe. Mis en place du **throtling** en modifiant le **security.yaml**. Un volet majeur du **RGPD** étant la **sécurité des données utilisateurs**, il est primordial d'implémenter ces mesures de sécurité.

## 2. Précisez les moyens utilisés :

Environnement de développement Laragon.

SGBD : Server MySQL.

Interface d'interaction avec le SGBD : HeidiSQL.

Framework Symfony.

Langage de script généraliste : PHP.

Outil de Modélisation Conceptuelle de Données : Looping.

Utilisation d'un bundle Symfony : password-reset-bundle.

Utilisation d'un bundle Symfony : Knp-Paginator.

Serveur SMTP + interface web de visionnage de mails : maildev.

## 3. Avec qui avez-vous travaillé ?

# DOSSIER PROFESSIONNEL (DP)

Equipe pédagogique Elan Formation

## 4. Contexte

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

Chantier, atelier, service ► Equipe Développeur web et web mobile

Période d'exercice ► Du 04/09/2023 au 16/02/2024

## 5. Informations complémentaires (facultatif)

Cliquez ici pour taper du texte.

## Activité-type 2

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

Exemple n°2 ► Exercice Forum

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

1. Réalisation d'un **MCD** et d'un **MLD**.
2. Etablir la nature des liens des **entités** entre elles.
3. Définition des **types de données**.
4. Démarrage du **service Apache**.
5. Démarrage du **server MySQL**.
6. En **PHP**, paramétrage des **paramètres de connexion PDO**.
7. Projet réalisé en utilisant le **design pattern MVP**.
8. Utilisation d'un **routeur frontal (index)**.
9. Création de requêtes en **SQL** dans les **Model** pour la récupération de données.
10. Création des **Controller** pour la récupération de données.
11. Pour les pages avec des entrées utilisateurs, utiliser les fonctions de **PHP** d'assainissement et de validation `filter_var` avec les **flag** correspondants (`FILTER_VALIDATE_EMAIL`, `FILTER_SANITIZE_FULL_SPECIAL_CHARS`).
12. Utiliser une **Regex (Regular Expression)** ou plusieurs via la fonction native **PHP** `preg_match( )` pour demander un motif de mot de passe précis conforme aux recommandations de la CNIL pour respecter le **RGPD**. Pour l'hébergement de fichier vérifier le **MIME** et la **taille du fichier** avec l'hébergement.

### 2. Précisez les moyens utilisés :

Utilisation d'un environnement de développement : Laragon

Utilisation et paramétrage d'un server MySQL

Utilisation d'un service Apache

Langage de script généraliste : PHP

Outil de Modélisation Conceptuelle de Données : Looping



# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

Equipe pédagogique Elan Formation

## 4. Contexte

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

Chantier, atelier, service ► Equipe Développeur web et web mobile

Période d'exercice ► Du 04/09/2023 au 16/02/2024

## 5. Informations complémentaires (facultatif)

Cliquez ici pour taper du texte.

## Activité-type 2

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

Exemple n°3 ► Cinéma

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

1. Réalisation d'un **MCD** et d'un **MLD**.
2. Etablir la nature des **liens** des **entités** entre elles.
3. Définition des **types de données**.
4. Démarrage du service **Apache**.
5. Démarrage du serveur **MySQL**.
6. En **PHP**, paramétrage des **paramètres de connexion PDO**.
7. Projet réalisé en utilisant le **design pattern MVP**.
8. Utilisation d'un routeur **frontal (index)**.
9. Création dans le dossier model du fichier de la connexion **PDO**.
10. Création des **Controller** pour la récupération de données.
11. Pour les pages avec des entrées utilisateurs, utiliser les fonctions de **PHP d'assainissement** et de **validation filter\_var** avec les **flag** correspondants (FILTER\_VALIDATE\_EMAIL, FILTER\_SANITIZE\_FULL\_SPECIAL\_CHARS).
12. Utiliser une **Regex (Regular Expression)** ou plusieurs via la fonction native PHP **preg\_match( )** pour demander un motif de mot de passe précis conforme aux recommandations de la **CNIL** pour respecter le **RGPD**.
13. Utilisation de **Docker** : création de fichiers **docker-compose** et **DockerFile** pour créer des **conteneurs**, sortes de machines virtuelles qui permettent **d'embarquer le projet et son environnement**. Dès lors pour lancer le projet il suffira de lancer le service Docker et de se rendre sur le **serveur local** sur le **bon port**.

### 2. Précisez les moyens utilisés :

# DOSSIER PROFESSIONNEL (DP)

Utilisation d'un environnement de développement : Laragon.

Utilisation et paramétrage d'un server MySQL.

Utilisation d'un service Apache.

Langage de script généraliste : PHP.

Outil de Modélisation Conceptuelle de Données : Looping.

Utilisation d'un service de conteneurisation : Docker.

## 3. Avec qui avez-vous travaillé ?

Elan Formation

## 4. Contexte

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

Chantier, atelier, service ► Equipe Développeur web et web mobile

Période d'exercice ► Du 13/03/2023 au 02/06/2023

## 5. Informations complémentaires (facultatif)

## DOSSIER PROFESSIONNEL (DP)

Cliquez ici pour taper du texte.

## DOSSIER PROFESSIONNEL (DP)

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

*(facultatif)*

Intitulé	Autorité ou organisme	Date
Licence Sciences de la vie	Université de Strasbourg	2017
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 .....*WETZSTEIN Anthony*..... ,  
déclare sur l'honneur que les renseignements fournis dans ce dossier sont exacts et que je  
suis l'auteur(e) des réalisations jointes.

Fait à ..*OSTWALD*..... le ..*29/01/2024*.....  
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)

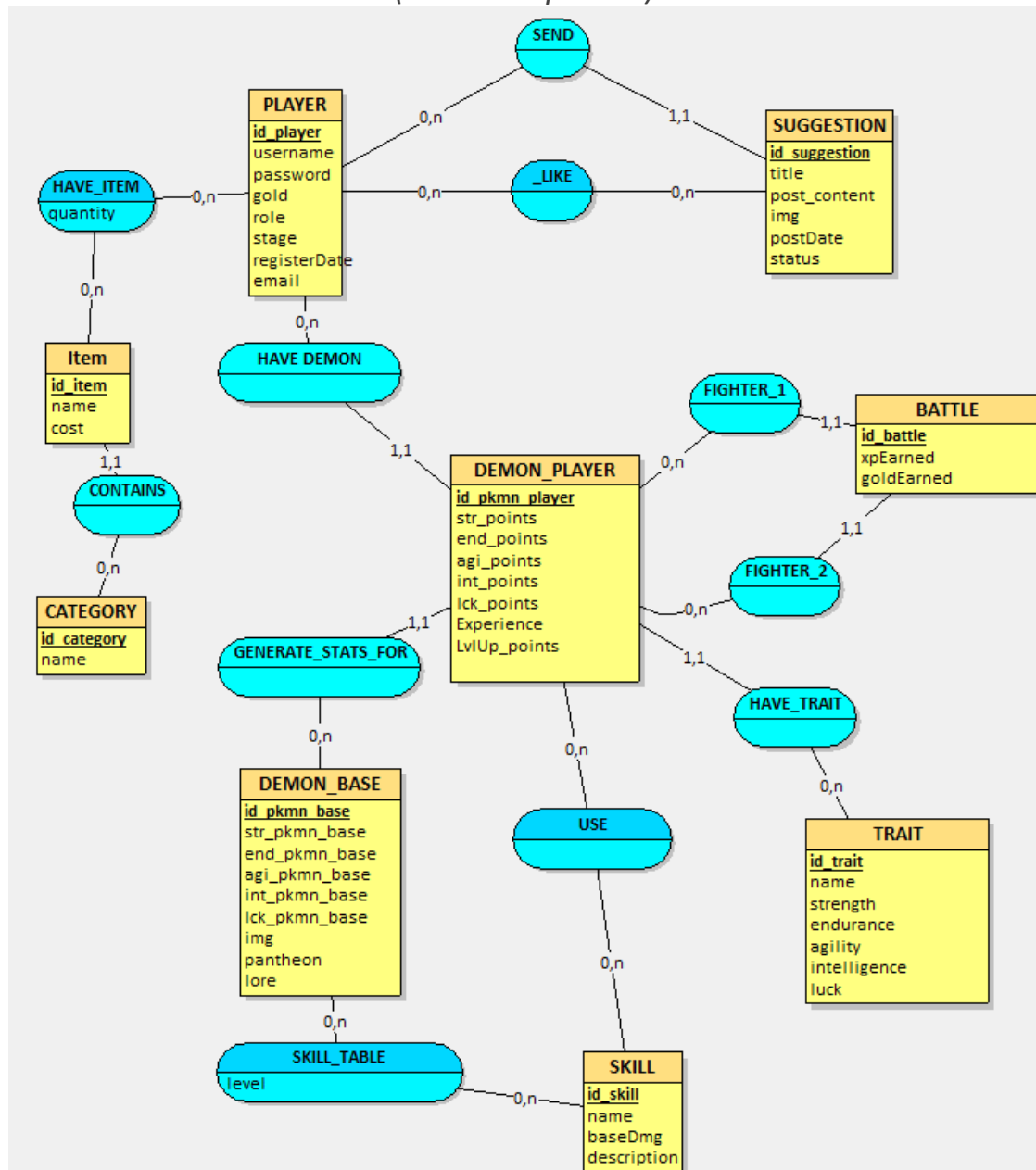


Figure 1: MCD Ad Temporis Finem





```

case "password": //must have at least one uppercase, one lowercase, one special char, must be set, at least 8 chars
//long, must have a numerical
if (strlen($fieldValue) >= 8 && preg_match('/[A-Z]/', $fieldValue) && preg_match('/[a-z]/', $fieldValue)
&& preg_match('/[0-9]/', $fieldValue) && preg_match('/[^\a-zA-Z0-9]/', $fieldValue) && isset($fieldValue))
{
    $permissionPwd = true;
    $pwd = $fieldValue; //stocking password in value
}
else
{
    $_SESSION["msg"] .= "<p>Password security is terrible.
Choose a password having :</p>
<ul>
    <li>An uppercase</li>
    <li>A lowercase</li>
    <li>A special char</li>
    <li>A number</li>
</ul>
<p>Password must be at least 8 chars long</p>"
    ;
}
break;

```

Figure 3: Cinema, Regex

```

1  var map = L.map('map').locate({setView: true, maxZoom: 16}); //If geolock is activated set map on user
2  //Leaflet Init
3
4  L.tileLayer('https://tile.openstreetmap.org/{z}/{x}/{y}.png', {
5      maxZoom: 19,
6      attribution: '&copy; <a href="http://www.openstreetmap.org/copyright">OpenStreetMap</a>'
7  }).addTo(map);
8
9  function onMapClickM(clickMarker) //If there is a marker already a click removes it and user can place a new one
10 {
11     if (!click) {
12         market
13             .setLatLng(clickMarker.latlng)
14             .addTo(map);
15         // clickMarker.latlng.lat = clickMarker.latlng.lat.toFixed(7)
16         // clickMarker.latlng.lng = clickMarker.latlng.lng.toFixed(8)
17         inputCoordinates.value = clickMarker.latlng.lat + " " + clickMarker.latlng.lng
18         console.log(clickMarker.latlng)
19         click = true;
20     } else {
21         map.removeLayer(market);
22         click = false;
23     }
24 }
25
26 var lc = L.control //Leaflet addon use
27 .locate({
28     position: "topright",
29     strings: {
30         title: "Votre position"
31     },
32 })
33
34 .addTo(map);
35
36 const inputCoordinates = document.querySelector("#session_coordonnees")
37 const popup2 = L.popup().setContent("Lieu de déroulement de la session");
38 const market = L.marker().bindPopup(popup2);
39 var click = false
40
41 if (!click)
42 {
43     map.on('click', onMapClickM);
44     console.log(click)
45 }

```

Figure 4 Carte Leaflet Sessions

```

1  <?php
2
3  namespace Models;
4
5  98 references | 0 implementations
6  abstract class Connect
7  {
8      1 reference
9      const HOST = "database";
10     1 reference
11     const DB = "cinema";
12     1 reference
13     const USER = "root";
14     1 reference
15     const PASS = "pass";
16
17     71 references | 0 overrides
18     static function connexion()
19     {
20         try {
21             return new \PDO(
22                 'mysql:host='. self::HOST .';dbname='. self::DB .';charset=utf8mb4',
23                 self::USER,
24                 self::PASS,
25                 [\PDO::ATTR_ERRMODE => \PDO::ERRMODE_EXCEPTION],
26             );
27         } catch (\PDOException $e)
28         {
29             return $e->getMessage();
30         }
31     }
32 }

```

Figure 5 Cinéma connexion PDO avec param Docker

```

<?php ob_start(); ?>

<div class="listFilms">
  <!-- TABLE STARTS HERE -->
  <table>
    <tr>
      <th></th>
      <th>Nom</th>
      <th>Prénom</th>
      <th>Date de naissance</th>
      <th>sexe</th>
    </tr>

    <?php
    foreach($reals as $realisateur)
    {
      ?>

      <tr>
        <td>
          <button class="deleteButton"><a href="index.php?action=deleteReal&id_real=<?=$realisateur["id_realisateur"]>">Supprimer</a></button>
        </td>

        <form method="post" action="index.php?action=updateRealisateur&id_real=<?=$realisateur["id_realisateur"]>" id="nom-real-form">
          <td>
            <label><?=$realisateur["nom"]></label>
            <input type="text" name="nom" value="nom" id="nom-real-input">
            <button type="submit" class="updateButton" class="updateButton" id="nom-real-button">Envoyer</button>
          </td>
        </form>
      </tr>
    }
  </table>
</div>

```

Figure 6: Vue Cinéma listing réalisateurs

```

1  // script.js
2  const darkModeToggle = document.querySelector('#toggleBtn');
3  const html = document.documentElement;
4  const b = document.querySelectorAll("b");
5  const cardP = document.querySelectorAll(".card p");
6  const zoomA = document.querySelectorAll(".zoom a");
7  const button = document.querySelectorAll("button");
8  // Function to toggle dark mode
9  function toggleDarkMode() {
10     html.classList.toggle('dark-mode');
11     b.forEach(link => {
12       link.classList.toggle('dark-mode');
13     });
14     cardP.forEach(link => {
15       link.classList.toggle('dark-mode');
16     });
17     zoomA.forEach(link => {
18       link.classList.toggle('dark-mode');
19     });
20     button.forEach(link => {
21       link.classList.toggle('dark-mode');
22     });
23   }
24
25   // Check for user's preferred mode (light or dark) in local storage
26   const currentMode = localStorage.getItem('mode');
27
28   // Apply the saved mode or default to light mode
29   if (currentMode === 'dark') {
30     toggleDarkMode();
31   }
32
33   // Toggle between light and dark mode when the button is clicked
34   darkModeToggle.addEventListener('click', () => {
35     toggleDarkMode();
36
37     // Save the current mode to local storage
38     if (html.classList.contains('dark-mode'))
39     {
40       localStorage.setItem('mode', 'dark');
41     } else
42     {
43       localStorage.setItem('mode', 'light');
44     }
45   });

```

Figure 7 : Javascript Forum mode clair/sombre

```

services:
  php:
    container_name: php
    image: php:8.3-apache
    ports:
      - 82:80 #82 le port que je veux, 80 par défaut
    volumes: ##emplacement par défaut des fichiers dans un serveur apache
      - ./var/www/html
    build:
      context: . #tu construis via ce dockerfile qui est là : dossier courant
      dockerfile : Dockerfile

  database:
    image: mysql:8.0
    container_name: mysql8
    command: --default-authentication-plugin=mysql_native_password
    ports:
      - 3307:3306
    environment:
      MYSQL_DATABASE: cinema
      MYSQL_ROOT_PASSWORD: pass
    volumes: #l'endroit où sont stockés les infos ds container avec équivalent sur machine locale
      - ./mysql:/var/lib/mysql #stockage contener = stockage machine locale

  phpmyadmin:
    image: phpmyadmin/phpmyadmin
    container_name: phpMA
    environment:
      PMA_ARBITRARY: 1
      PMA_HOST: database
      PMA_USER: root
      PMA_PASSWORD: pass
    ports: #I used 8899 and docker uses 80
      - 8888:80

```

Figure 8 : docker-compose

```

1 FROM php:8.3-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

```

Figure 8 : DockerFile

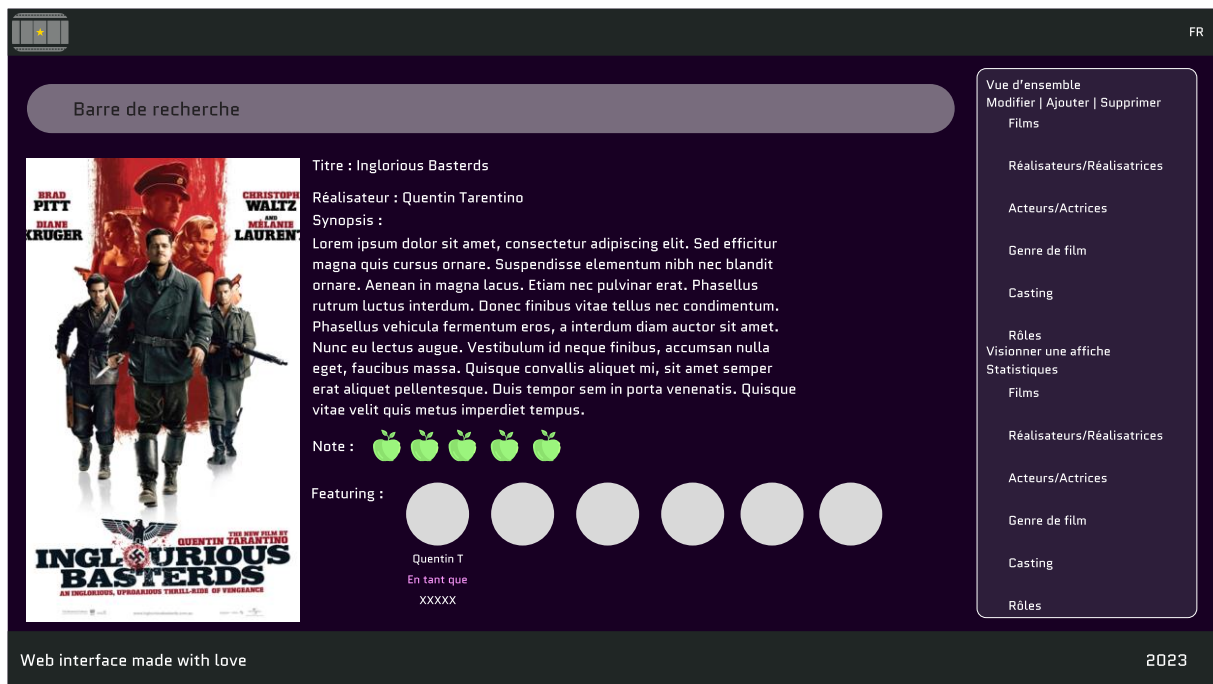


Figure 7 Maquettage Cinema

```
<footer class="footer">
  <div class="footerRow">
    <div class="eulaRow">
      <a class="linkNoBase" href="{{path('app_tos')}}">ToS</a>
    </div>
    <div class="eulaRow">
      <a class="linkNoBase" href="{{path('app_legal')}}">Legal notice</a>
    </div>
    <div class="socialsColumn">
      <p class="socialsText">Socials</p>
      <ul class="socials">
        <li><a class="linkNoBase" href="#"><img class="contactSocial" alt=
        <li><a class="linkNoBase" href="#"><img class="contactSocial" alt=
        <li><a class="linkNoBase" href="#"><img class="contactSocial" alt=
      </ul>
    </div>
    <div class="contactRow">
      <a class="linkNoBase align" href="mailto:admin@atf.com"><img class="cc
    </div>
  </div>
</footer>
```

Figure 8 exemple de balise sémantique footer

```
<nav class="navbar" id="mainMenu">
```

Figure 9 Balise sémantique navbar

```
<aside class="secondRow">
  <div class="longCard">
    <div class="element">
      <p>Turn-based system<
      <img class="elementsV
    </div>
    <div class="element">
      <p>RPG elements</p>
      <img class="elementsV
    </div>
    <div class="element">
      <p>Learn about myths
      <img class="elementsV
    </div>
  </div>
</aside>
```

Figure 11 Balise sémantique aside Ad Temporis

```
<main class="forceRow">
  <div class="ouverture">
    <div class="rowButtons">
      <a class="looksLikeAButtonGre
      <a class="looksLikeAButtonGre
      <div class="specialButton">
        <a href="{{path('game')}}
        <p class="looksLikeAButto
      </div>
    </div>
    <p>You are lost and you won't be able
    defeat your foes. Vanquish mighty
  </div>
```

Figure 10 Balise sémantique main



```
<meta name="description" content="Homepage of Ad Temporis Finem">
<meta name="keywords" content="Homepage,Game">
```

Figure 12 Balises méta avec description et keywords

```
<ul class="socials">
<li><a class="linkNoBase" href="#"> X</a></li>
<li><a class="linkNoBase" href="#"> Instagram</a></li>
<li><a class="linkNoBase" href="#"> TikTok</a></li>
</ul>
```

Figure 14 SEO : maillage externe avec lien réseaux

Username \*

Email \*

Confirm your email \*

Password \*

*Pwd must be at least 12 chars long,  
you need at least 1 UC, 1 LC, 1 number,  
1 special char*

Type your password again \*

You agree to our Terms of Service

☐

[Terms of Service](#)

Fields marked with \* are mandatory.

Validate

Figure 13Ad Temporis Finem Formulaire

```

@media screen and (max-width:500px) {
    #mainMenu {
        display: none;
    }
    .navbar ul {
        flex-direction: column;
        align-items: center;
        flex-wrap: nowrap;
        height: 25rem;
    }
    #menuLabel
    {
        height: 3rem;
        align-items: center;
    }
    #menuShowHide+label:before {
        content: "Show menu";
    }
    #menuShowHide:checked+label:before {
        content: "Hide Menu";
    }
    .card {
        width: 100%;
    }
}

```

Figure 15 Ad Temporis Finem : Media Queries