



*Projet réalisé dans le cadre de la présentation au Titre*

**Développeur Web et Web Mobile**

*présenté par*

**Anton VOLEANIN**

[**SOMMAIRE**](#_hwxrebqq0mug) **2**

[**INTRODUCTION**](#_w8lnef24xvcs) **4**

[**LISTE DES COMPÉTENCES DU RÉFÉRENTIEL QUI SONT COUVERTES PAR LE PROJET**](#_5iqtdmcl2lvu) **6**

[1. Développer la partie front-end d’une application web ou web mobile sécurisée](#_uvhrxsf686mt) 6

[1.1. Maquetter une application](#_aezashnhefzl) 6

[1.2. Réaliser une interface utilisateur web statique et adaptable](#_ctkracq1xgdj) 7

[1.3. Développer une interface utilisateur web dynamique](#_lqhx02sfixl1) 8

[2. Développer la partie back-end d’une application web ou web mobile sécurisée](#_8bjehqvunr57) 9

[2.1. Créer une base de données](#_j8uldkum57or) 9

[2.2. Développer les composants d’accès aux données](#_ui3g3klxekh5) 9

[2.3. Documenter le déploiement d’une application dynamique web ou web mobile](#_yh8507cm1oqs) 10

[2.4. Sécurité](#_hxeo4wm0p96e) 10

[**RÉSUMÉ DE PROJET**](#_9aioy1vfd725) **11**

[**CAHIER DES CHARGES**](#_608swhs5ifuu) **12**

[1. Besoins et objectifs de l’application](#_j7atbd6xsad1) 12

[1.1 Besoins](#_dnq1nlm6gicx) 12

[1.2 Objectifs](#_tql8bnk5siqv) 12

[2. Exigences Fonctionnelles](#_2rzz56npkyv2) 13

[2.1 Interfaces Utilisateur](#_xrjf41t8pu1h) 13

[2.2 Développement Front-End](#_aucqaqflhzz3) 13

[2.2.1 Conception de l'Interface Utilisateur](#_1zqxq4hyq1ib) 13

[2.2.2 Implémentation de l'Interface Utilisateur Web](#_29fmtyrbrotw) 13

[2.2.3 Développement de l'Interface Utilisateur Web Dynamique](#_32hrhjnig9ww) 13

[2.3 Développement Back-End](#_ee3fwv3hnne1) 14

[2.3.1 Création de la Base de Données](#_tq8gx0icmbh9) 14

[2.3.2 Accès aux Données et Composants Métier Côté Serveur](#_gf271fxjjz5c) 14

[2.3.3 Documentation et Sécurité](#_ahkjgx3q1qic) 14

[3. Exigences Non-Fonctionnelles](#_5ibg30t9who9) 15

[3.1 Performance](#_1ubyet5k7ich) 15

[3.2 Sécurité](#_arw0eyju0s9q) 15

[3.3 Scalabilité](#_hr96k6igg1oj) 15

[4. Livrables](#_bexldzt5ca2o) 15

[**SPÉCIFICATIONS TECHNIQUES**](#_vkblltiggd4h) **16**

[1. Outils de Développement](#_ov6f4f8enmzd) 16

[php bin/console (Symfony Console)](#_wc3clqvcibzk) 16

[npm (Node Package Manager)](#_ov6f4f8enmzd) 16

[Composer (composer)](#_ov6f4f8enmzd) 17

[WampServer (Wamp)](#_wcnugrit0tx1) 17

[Visual Studio Code (VSCode)](#_day3zpfyp2m4) 17

[2. Front-end](#_vtvsgqo4gafr) 18

[3. Back-end](#_swbc5e5umdvh) 18

[4. Déploiement](#_vcljbdi3ktz3) 18

[**RÉALISATIONS PERSONNELLES**](#_oouepr01jt0) **19**

[1. Exemples de maquettes](#_2uo0ztjcpiln) 19

[2. “Like” et “Dislike” (la gestion des interactions utilisateur pour les questions et les réponses)](#_cobfr77bw2am) 20

[3. Récupérer (Fetch) les questions et réponses récentes par utilisateur sur la page de profil](#_glgd0gt26rak) 22

[4. Filtrage Dynamique par Catégorie](#_jqos0f9148es) 23

[5. Intégration de highlight.js](#_eh4dtu840n0r) 25

[6. Barre de recherche](#_k5eckehh4x8z) 26

[7. MCD](#_58sygappe5fo) 28

[**PRÉSENTATION DU JEU D’ESSAI**](#_wn0gd33pmqnq) **29**

[Utilisation de Jeux de Données](#_majj9hsarc7w) 29

[Tests des Fonctionnalités Utilisateurs](#_fgm8lgekn1z3) 29

[Tests des Formulaires](#_tflf4igyh7sl) 30

[Tests de Highlight.js](#_1vaf8jstss0c) 30

[**VULNÉRABILITÉS DE SÉCURITÉ ET VEILLE**](#_qmcffisetb8e) **31**

[1. SÉCURITÉ](#_63y6ln1bf5z0) 31

[Hachage des Mots de Passe](#_k011odbvs294) 31

[Utilisation du Bundle Symfony Security](#_nbsl0t4eu577) 31

[Gestion des Inscriptions avec Symfony](#_63pj18lf65y7) 31

[Attribution du Rôle "ROLE\_ADMIN"](#_ffxnfiz8yy4a) 31

[Vérifications avec {% if app.user %}](#_d8mx8ua2yiu0) 32

[2. VEILLE](#_53udo7akj8ej) 32

[Recherche Google et Stack Overflow](#_mn5xv6kcc24t) 32

[Chaînes YouTube](#_4lhwr47kfi1m) 32

[Podcast Spotify - "Syntax"](#_qwz36dd0348m) 33

[Avantages de la Veille Informatique](#_j8lmhh3eo9qo) 33

[Défis et Tendances Futures](#_fqprs340qbbp) 33

[**SITUATION DE TRAVAIL AYANT NÉCESSITÉ UNE RECHERCHE + EXTRAIT DU SITE**](#_izpxo1cr7p7a) **34**

# 

# **INTRODUCTION**

Depuis mon plus jeune âge, j'ai toujours été passionné par la technologie et le design.

Après mes études secondaires, j'ai choisi de poursuivre des études universitaires en Marketing. Cependant, ma curiosité pour le design et la créativité ne m'a pas quitté. Au cours de mes études universitaires, j'ai commencé à apprendre à utiliser les programmes Adobe par simple plaisir. J'ai réalisé que le design web ne devait plus être un simple passe-temps, mais bien une voie de carrière prometteuse.

Après avoir réussi à obtenir mon diplôme universitaire, j'ai pris la décision de me lancer à plein temps en tant que web designer freelance. Au fil des années, j'ai perfectionné mes compétences et constitué un portfolio de projets diversifiés. Cependant, j'ai toujours eu un rêve: créer un site web complet de A à Z. Mon objectif ultime était de devenir un développeur Full-stack, et je savais que je devais élargir mes connaissances pour y parvenir.

C'est ainsi que j'ai découvert un programme de développement web à l'AFPA. Après avoir réussi les tests d'admission, j'ai intégré la formation DWWM à l'AFPA. Au cours de cette formation, j'ai eu la chance de décrocher un stage au sein de la plateforme d'e-learning Academie WS, qui m'a offert une opportunité exceptionnelle: **développer un forum à partir de zéro en utilisant le framework Symfony**. J'ai travaillé sous la supervision de mon

responsable de stage, et j'ai également eu l'occasion de collaborer avec deux de mes collègues de formation, qui travaillaient sur un projet différent. Nous nous sommes mutuellement informés de nos avancées, de nos réalisations et de nos fonctionnalités intéressantes.

À la fin de ce stage, j'ai réussi à respecter les délais et à créer un forum fonctionnel et épuré pour la plateforme d'e-learning en ligne en pleine croissance, Academie WS.

C'est avec enthousiasme que je partage cette étape de mon parcours professionnel avec vous, et je suis impatient de continuer à apprendre, grandir et réaliser mon rêve de devenir un développeur Full-stack accompli.

# **LISTE DES COMPÉTENCES DU RÉFÉRENTIEL QUI SONT COUVERTES PAR LE PROJET**

## **Développer la partie front-end d’une application web ou web mobile sécurisée**

### Maquetter une application

Pendant cette activité, mon rôle principal était de concevoir des maquettes visuelles des interfaces utilisateur pour le site web **d'Académie WS**, tout en intégrant l'identité de marque existante. J'ai été grandement inspiré par le célèbre site **Stack Overflow**. Mon travail consistait à créer des designs pour les différentes pages, les éléments d'interface et l'organisation du contenu, en veillant à maintenir la cohérence avec les éléments visuels de la marque [**Académie WS**](https://academiews.fr/).

**Stack Overflow** est une référence en matière d'ergonomie et de conception d'interfaces utilisateur, et j'ai cherché à appliquer les meilleures pratiques de ce site à la conception des maquettes pour le forum d'Académie WS. Cela incluait la création de mises en page intuitives, la facilitation de la navigation, et la garantie d'une expérience utilisateur optimale pour les visiteurs du site.

En m'inspirant de Stack Overflow, j'ai cherché à créer des maquettes qui non seulement reflètent l'identité de marque d'Académie WS, mais qui offrent également une convivialité exceptionnelle et une présentation visuelle attrayante pour les utilisateurs. Cela a contribué à garantir que les interfaces utilisateur du site étaient à la fois esthétiques et fonctionnelles, tout en offrant une expérience similaire à celle d'un site renommé comme [**Stack Overflow**](https://stackoverflow.com/).

### Réaliser une interface utilisateur web statique et adaptable

J'ai utilisé mes connaissances en HTML/CSS pour créer toutes les pages principales du forum.

**Intégration de Bootstrap pour le design et l'expérience utilisateur:** J'ai utilisé Bootstrap, un framework CSS, pour concevoir une interface utilisateur attrayante et conviviale. Les nombreux composants et styles prédéfinis de Bootstrap ont été exploités pour garantir une cohérence visuelle sur l'ensemble du forum, assurant ainsi une expérience utilisateur harmonieuse.

**Éco-conception:** J'ai pris en compte les principes d'éco-conception pour minimiser l'impact environnemental des interfaces, en optimisant notamment les ressources (images, scripts) pour réduire la consommation d'énergie et les émissions de CO2.

**Accessibilité Numérique:** J'ai veillé à ce que les interfaces soient accessibles à un large public, en tenant compte des recommandations d'accessibilité numérique. Cela inclut la mise en place de balises HTML sémantiques, l'ajout d'attributs d'accessibilité, et la vérification de la compatibilité avec les technologies d'assistance.

**Responsive et Conception "Mobile First":** J'ai suivi les principes de conception "mobile first" et de "responsivité", ce qui signifie que j'ai conçu les interfaces en pensant d'abord à l'expérience sur les appareils mobiles, puis j'ai adapté leur mise en page pour les écrans plus larges. Cela garantit une expérience utilisateur optimale sur une variété de dispositifs, du mobile au bureau.

**Conformité aux Standards du W3C:** J'ai assuré la conformité des pages web aux normes du World Wide Web Consortium (W3C) pour garantir la qualité du code HTML, CSS et JavaScript, ainsi que la compatibilité avec différents navigateurs.

**Prise en Compte des Situations d'Handicap:** J'ai intégré des fonctionnalités et des ajustements pour tenir compte des situations d'handicap, notamment en offrant des alternatives pour les médias, en permettant une navigation clavier, et en garantissant un contraste adéquat pour les personnes malvoyantes.

### Développer une interface utilisateur web dynamique

Pour répondre à l'exigence de développement d'une interface utilisateur web dynamique, j'ai mis en place plusieurs composants et technologies complémentaires. Voici comment j'ai abordé cette tâche :

* **Utilisation de Twig pour la génération dynamique de contenu:** J'ai exploité le moteur de template Twig, intégré à Symfony, pour créer des pages web dynamiques. Cela signifie que le contenu des pages pouvait être généré en temps réel en fonction des données provenant du backend. Par exemple, les dernières questions, les réponses aux discussions et les détails des utilisateurs étaient actualisés sans nécessiter de rechargement complet de la page.
* **Intégration de vidéos YouTube dynamiques:** Afin d'enrichir l'expérience des utilisateurs, j'ai utilisé JavaScript pour intégrer dynamiquement les vidéos les plus récentes de l'Académie WS Steak au Verre Flou sur la page du forum. Cette fonctionnalité a permis aux utilisateurs de visionner directement les derniers contenus vidéo sans quitter le site. Pour ce faire, j'ai fait appel à l'API YouTube, ce qui m'a permis de récupérer les informations sur les vidéos les plus récentes et de les afficher de manière dynamique sur la page.
* **Utilisation de jQuery et AJAX pour l'interactivité:** Pour des éléments tels que les likes et la recherche, j'ai fait usage de jQuery, une bibliothèque JavaScript, en conjonction avec AJAX. Cette combinaison a permis une interaction fluide avec le serveur en arrière-plan, sans nécessiter le rechargement complet de la page. Ainsi, les utilisateurs pouvaient aimer une réponse ou effectuer des recherches en temps réel, améliorant ainsi leur expérience de navigation.
* La pagination dans le projet en utilisant le bundle **KnpPaginatorBundle**

En somme, l'utilisation de ces technologies et approches a abouti à la création d'un forum en ligne dynamique et interactif. Cette solution offre aux utilisateurs une expérience fluide, tout en maintenant une interface esthétiquement plaisante et fonctionnelle.

## **Développer la partie back-end d’une application web ou web mobile sécurisée**

### Créer une base de données

**Documentation Technique (MCD):** Créer une représentation visuelle de la structure de la base de données en utilisant des outils comme PHPMyAdmin Designer. Définir les entités, les relations et les attributs pour une conception précise de la base de données.

**Gestion des Contraintes et Optimisation des Accès:** Définir et appliquer des contraintes telles que les clés primaires, les clés étrangères, et garantir l'intégrité des données. Utiliser le système d'authentification de Symfony et les rôles (comme "ROLE\_ADMIN") pour gérer l'accès des utilisateurs à différentes parties de la base de données.

**Système de Gestion de Base de Données (SGBD):** Sélectionner le SGBD approprié pour le projet, en l'occurrence MySQL, et mettre en place la configuration requise. Assurer la compatibilité avec les besoins de stockage, d'interrogation et de gestion des données.

### Développer les composants d’accès aux données

Développer et mettre en place tous les composants nécessaires pour la création et la gestion d'un forum en ligne, en se concentrant sur les entités principales telles que les utilisateurs (User), les questions (Question), les réponses (Answer) et les catégories (Category).

**Sous-Tâches**

* Conception de la Base de Données
* Gestion des Requêtes SQL Dynamiques
* Implémentation de l'ORM Doctrine
* Sécurisation des Données Utilisateur
* Création d'un Back-Office d'Administration
* Développement des Composants Métier Côté Serveur
* Optimisation de la Performance
* Tests et Débogage
* Documentation Technique
* Formation et Support

### Documenter le déploiement d’une application dynamique web ou web mobile

**Configuration des Environnements:**

J'ai configuré les environnements de développement et de production en utilisant des fichiers de configuration .env et .env.local. Cela inclut la gestion des paramètres de base de données.

**Contrôle de Version avec Github:**

J'ai utilisé Github pour gérer le contrôle de version. J'ai créé des branches pour chaque fonctionnalité, "commité" mes modifications et soumis mes changements pour fusion.

### Sécurité

Un aspect essentiel de mon travail dans ce projet était de garantir la sécurité des données et de l'ensemble du site web. Pour ce faire, j'ai entrepris les actions suivantes:

* Mise en place de procédures de gestion des utilisateurs avec des contrôles d'accès stricts basés sur les rôles, assurant que seuls les utilisateurs autorisés avaient accès à certaines parties du site.
* Utilisation de techniques de hachage pour stocker les mots de passe des utilisateurs de manière sécurisée dans la base de données, garantissant ainsi la confidentialité des informations d'identification.

# **RÉSUMÉ DE PROJET**

Pendant ma période en entreprise (Académie WS), dans le cadre de ma formation en **Développeur Web et Web Mobile à l'Afpa**, j'ai travaillé sur la création d'un forum en ligne appelé **"Steak au Verre Flou"** pour la plateforme de e-learning Académie WS. L'objectif principal de ce projet était de créer un forum de référence pour les utilisateurs de l'Académie WS, ainsi que de promouvoir cette plateforme.

Dans la partie **Front-end** du projet, j'ai utilisé l'outil **Figma** pour concevoir les maquettes des interfaces du forum, définissant ainsi son apparence et l'expérience utilisateur (UI/UX). J'ai utilisé mes connaissances en HTML/CSS pour créer toutes les pages principales du forum, en intégrant le framework Bootstrap pour garantir la réactivité et l'adaptabilité du site. Pour dynamiser l'interface utilisateur, j'ai utilisé le moteur de template Twig, le framework Symfony, JavaScript pour les fonctionnalités interactives, et jQuery+AJAX pour des éléments comme les likes et la recherche.

Du côté **Back-end**, j'ai utilisé PHPMyAdmin pour concevoir la structure de la base de données en suivant le modèle conceptuel de données (MCD). J'ai également utilisé l'EntityManagerInterface pour mettre en place et gérer les interactions avec la base de données. J'ai développé des composants d'accès aux données SQL et des composants métier côté serveur en utilisant Query Builder, l'ORM Doctrine, et EasyAdmin pour créer une interface d'administration conviviale.

En ce qui concerne la sécurité et le déploiement, j'ai mis en place des procédures de gestion des utilisateurs, configuré les environnements de développement et de production, et utilisé **GitHub** pour le contrôle de version, assurant ainsi un déploiement sécurisé et efficace du forum **"Steak au Verre Flou"**.

# **CAHIER DES CHARGES**

## 1. Besoins et objectifs de l’application

### 1.1 Besoins

* Concevoir des interfaces utilisateur interactives et conviviales, en utilisant Figma pour créer des maquettes détaillées des interfaces du forum, garantissant ainsi une expérience utilisateur optimale.
* Intégrer des fonctionnalités dynamiques pour améliorer l'interactivité de l'application, en utilisant Bootstrap, JavaScript, jQuery et AJAX pour rendre le forum vivant et engageant.
* Développer des composants back-end sécurisés pour gérer les données et les opérations métier.
* Documenter le processus de déploiement de l'application.
* Assurer la sécurité des données et de l'application dans son ensemble.

### 1.2 Objectifs

* Concevoir des interfaces utilisateur web adaptatives en utilisant les meilleures pratiques de développement.
* Implémenter des fonctionnalités dynamiques pour améliorer l'interactivité de l'application.
* Développer des composants back-end robustes pour gérer efficacement les données et les opérations métier.
* Documenter le processus de déploiement de l'application, y compris la gestion des environnements et du contrôle de version.
* Garantir la sécurité des données et de l'application en mettant en place des contrôles d'accès et en utilisant des techniques de sécurité appropriées.

## 2. Exigences Fonctionnelles

### 2.1 Interfaces Utilisateur

* Les interfaces utilisateur doivent être adaptatives et dynamiques, conformément aux directives établies lors des activités.

### 2.2 Développement Front-End

##### ***2.2.1 Conception de l'Interface Utilisateur***

* Pour cette tâche, la création de maquettes pour les pages d'accueil, de questions-réponses, de profil utilisateur, de vidéos et de publication de questions est requise.
* Il est également nécessaire d'intégrer les recommandations de sécurité, de suivre les principes d'éco-conception, de respecter les normes d'accessibilité numérique, d'adopter une approche "Mobile First", de garantir la conformité aux normes du W3C et de tenir compte des besoins des utilisateurs en situation de handicap.

##### ***2.2.2 Implémentation de l'Interface Utilisateur Web***

* L'implémentation de l'interface utilisateur web est la prochaine étape de notre projet, basée sur les concepts développés lors des activités précédentes.
* L'intégration du framework Bootstrap pour garantir la convivialité et la réactivité est un élément clé.
* Il est essentiel d'inclure des images réactives et de maintenir la conformité aux normes du W3C pour assurer la qualité du code.

##### ***2.2.3 Développement de l'Interface Utilisateur Web Dynamique***

* Cette phase implique le développement de l'interface utilisateur web dynamique en utilisant Bootstrap et JavaScript pour créer une expérience interactive.
* Des éléments tels que la création d'un menu "Burger", l'intégration de YouTube et l'ajout d'un filtrage interactif par catégorie font partie de cette tâche.
* La mise en œuvre de la fonction "J'aime/Je n'aime pas" avec jQuery et AJAX pour le traitement asynchrone est un aspect crucial.
* Garantir la pagination à l'aide du bundle KnpPaginatorBundle fait également partie de cette activité.

### 2.3 Développement Back-End

##### **2.3.1 Création de la Base de Données**

* La création d'une base de données relationnelle implique diverses tâches, notamment la conception de la structure de la base de données, la gestion des requêtes SQL de manière dynamique et la mise en œuvre de l'ORM Doctrine.
* Des mesures de sécurité telles que le hachage des mots de passe pour un stockage sécurisé et la protection des données des utilisateurs sont essentielles.
* Garantir des contrôles d'accès stricts et des rôles, y compris la gestion des utilisateurs et de l'administration, fait partie de cette tâche.

##### ***2.3.2 Accès aux Données et Composants Métier Côté Serveur***

* Le développement et la mise en œuvre des composants nécessaires à la création et à la gestion d'un forum en ligne sont l'objectif principal.
* Cela inclut des entités telles que les utilisateurs, les questions, les réponses et les catégories.
* L'utilisation de l'ORM Doctrine pour mapper les classes PHP aux tables de la base de données et la gestion de la structure de la base de données à l'aide des commandes Doctrine sont des aspects clés.
* La création d'un système de gestion des utilisateurs avec un contrôle d'accès basé sur les rôles (par exemple, "ROLE\_ADMIN") est essentielle pour la sécurité et le contrôle d'accès.
* L'intégration du bundle EasyAdmin pour le développement rapide d'un back-office d'administration fait partie de cette activité.

##### ***2.3.3 Documentation et Sécurité***

* La documentation du processus de déploiement est essentielle pour garantir un déploiement en douceur de l'application "Steak au verre flou".
* La configuration des environnements, la gestion du contrôle de version avec GitHub et la sécurisation des données et du site web sont des tâches cruciales.
* Cela comprend la mise en place de procédures de gestion des utilisateurs avec des contrôles d'accès stricts basés sur les rôles et l'utilisation de techniques de hachage pour un stockage sécurisé des mots de passe.

### 3. Exigences Non-Fonctionnelles

##### **3.1 Performance**

* Les temps de réponse doivent être optimaux.
* Gestion des charges élevées.

##### **3.2 Sécurité**

* Les données utilisateur doivent être sécurisées.
* L'authentification des utilisateurs doit être robuste.

##### **3.3 Scalabilité**

* L'application doit être prête à évoluer pour gérer une augmentation de l'utilisation.

### 4. Livrables

* Code source de l'application
* Historique de tous les commits

# **SPÉCIFICATIONS TECHNIQUES**

## Outils de Développement

### php bin/console (Symfony Console)

* **Objectif:** L'outil **php bin/console** est la console de Symfony, qui offre un ensemble de commandes pour gérer divers aspects de l'application web Steak au Verre Flou. Les spécifications techniques concernant son utilisation sont les suivantes :
* **Gestion de la Base de Données:** Utilisation de **php bin/console doctrine:database:create** pour créer la base de données spécifiée dans la configuration. Cette commande doit être utilisée avec précaution, en garantissant que les informations de connexion à la base de données sont correctes et sécurisées.
* **Génération d'Entités:** Utilisation de **php bin/console make:entity** pour générer des classes d'entités PHP, définissant ainsi la structure des données du projet. Il est essentiel de suivre les conventions de nommage et les relations entre les entités.
* **Migrations de Base de Données:** Utilisation de **php bin/console make:migration** et **php bin/console doctrine:migrations:migrate** pour effectuer des mises à jour de la base de données tout en préservant les données existantes. Les migrations doivent être gérées de manière à maintenir l'intégrité de la base de données.

### npm (Node Package Manager)

* **Objectif:** npm est utilisé pour gérer les dépendances JavaScript, les scripts de build, et les packages nécessaires au développement de l'interface utilisateur web du forum Steak au Verre Flou.
* **Installation des Dépendances:** Utilisation de la commande **npm install** pour installer toutes les dépendances du projet définies dans le fichier **package.json**. Cela garantit que les packages JavaScript requis sont disponibles localement.
* **Scripts de Build:** Définition de scripts de build personnalisés dans le fichier package.json pour automatiser des tâches telles que la minification des fichiers JavaScript, la compilation des fichiers CSS, etc.
* **Mise à Jour des Dépendances:** La commande **npm update** est utilisée pour mettre à jour les dépendances vers les versions les plus récentes, tout en respectant les contraintes de version spécifiées.

### Composer (composer)

* **Objectif:** Composer est un gestionnaire de dépendances PHP utilisé pour gérer les bibliothèques PHP requises par le projet Steak au Verre Flou.
* **Installation des Dépendances PHP:** Utilisation de **composer install** pour installer toutes les dépendances PHP spécifiées dans le fichier **composer.json**. Cela garantit que les bibliothèques tierces requises par Symfony et d'autres composants sont disponibles et fonctionnelles.
* **Mise à Jour des Dépendances PHP:** La commande **composer update** est utilisée pour mettre à jour les dépendances PHP vers les versions les plus récentes, tout en respectant les contraintes de version spécifiées dans le fichier **composer.json**.
* **Ajout de Nouvelles Dépendances:** Utilisation de **composer require <package-name>** pour ajouter de nouvelles dépendances PHP au projet. Cette commande doit être accompagnée d'une documentation appropriée sur l'utilisation de la nouvelle bibliothèque ou du composant.

### WampServer (Wamp)

* **Objectif:** WampServer est un environnement de développement web qui permet de configurer un serveur web local pour le développement de l'application Steak au Verre Flou.
* **Configuration du Serveur Local:** WampServer est utilisé pour configurer un environnement de développement local comprenant un serveur web Apache, une base de données MySQL et PHP. Cela permet de développer et de tester l'application localement avant le déploiement.
* **Gestion de la Base de Données:** WampServer facilite la gestion de la base de données MySQL, notamment la création de bases de données, l'importation/exportation de données et la configuration de l'accès à la base de données.

### Visual Studio Code (VSCode)

* **Objectif:** Visual Studio Code est l'environnement de développement intégré (IDE) utilisé pour écrire, tester et déboguer le code source de l'application Steak au Verre Flou.
* **Éditeur de Code:** VSCode est utilisé comme éditeur de code principal pour PHP, JavaScript, HTML, CSS et d'autres langages web. Il offre des fonctionnalités telles que la coloration syntaxique, l'autocomplétion, et l'intégration avec les outils de développement.

## Front-end

* Langages de Programmation: **HTML, CSS, JavaScript**
* Framework CSS: **Bootstrap**
* Moteur de Templates: **Twig (intégré à Symfony)**
* Bibliothèques JavaScript: **jQuery**
* API Externe: **YouTube Data API (pour afficher les dernières vidéos YouTube)**
* Principes de Conception: **Responsive Web Design, Mobile First**
* Normes de Conformité: **HTML5, CSS3, JavaScript ES6, W3C Accessibility Guidelines**
* Conception des Maquettes: **Figma**

## Back-end

* Langage de Programmation: **PHP**
* Framework PHP: **Symfony**
* ORM (Object-Relational Mapping): **Doctrine**
* Base de Données: **MySQL (conçue à l'aide du modèle conceptuel de données - MCD)**
* Sécurité: **Hachage des mots de passe, Gestion des sessions utilisateur, Contrôle d'accès basé sur les rôles, symfony/security-bundle**
* Bibliothèque JavaScript: **jQuery**
* Back-Office d'Administration: **EasyAdmin (intégré à Symfony)**
* Système de Contrôle de Version: **GitHub**

## Déploiement

* Serveur d'Hébergement: **OVH (ovh.com)**

**Configuration de l'Environnement de Production:** OVH permet de configurer un environnement de production sécurisé pour l'application. Cela inclut la configuration du serveur web, de la base de données, des certificats SSL, et d'autres paramètres essentiels.

**Gestion des Fichiers et des Dossiers:** OVH offre un espace de stockage pour les fichiers et les ressources de l'application.

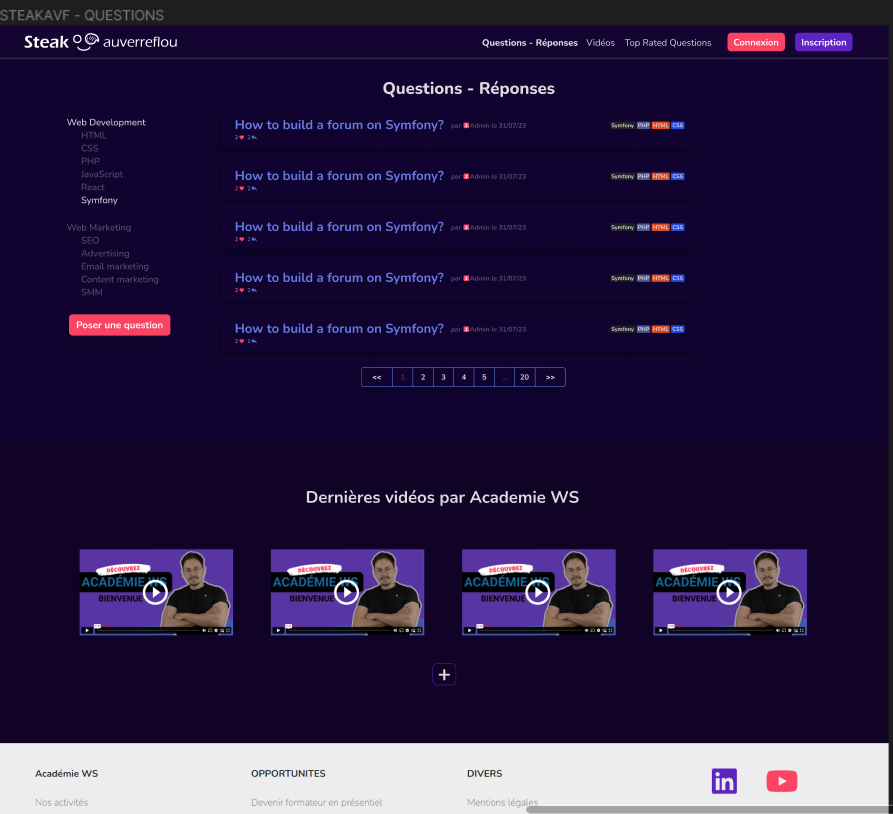
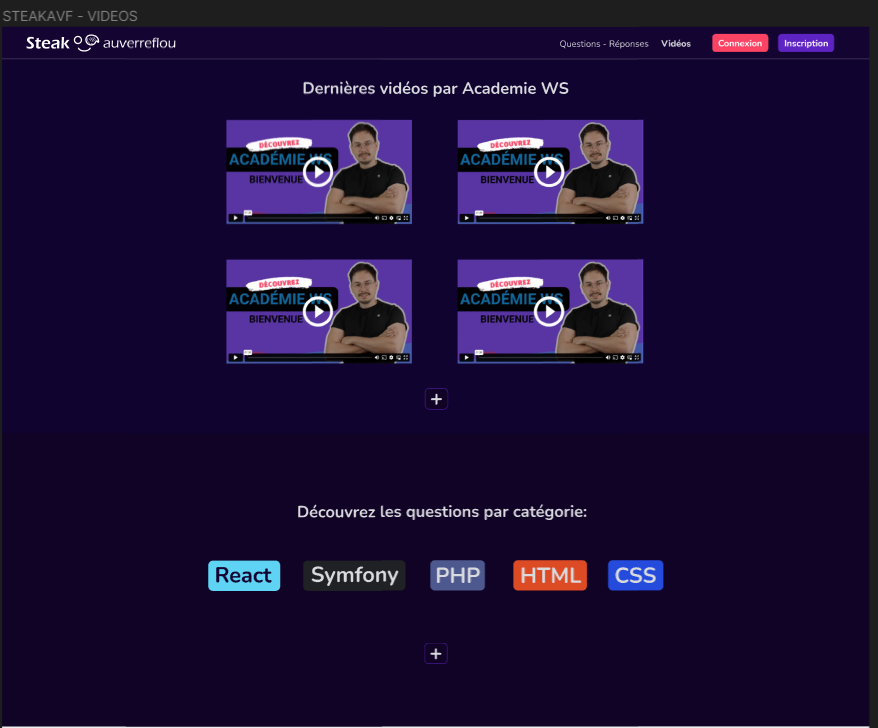
**Sécurité:** OVH propose des mesures de sécurité, telles que la configuration du pare-feu, la surveillance de la sécurité, et la gestion des certificats SSL, pour protéger l'application contre les menaces en ligne.

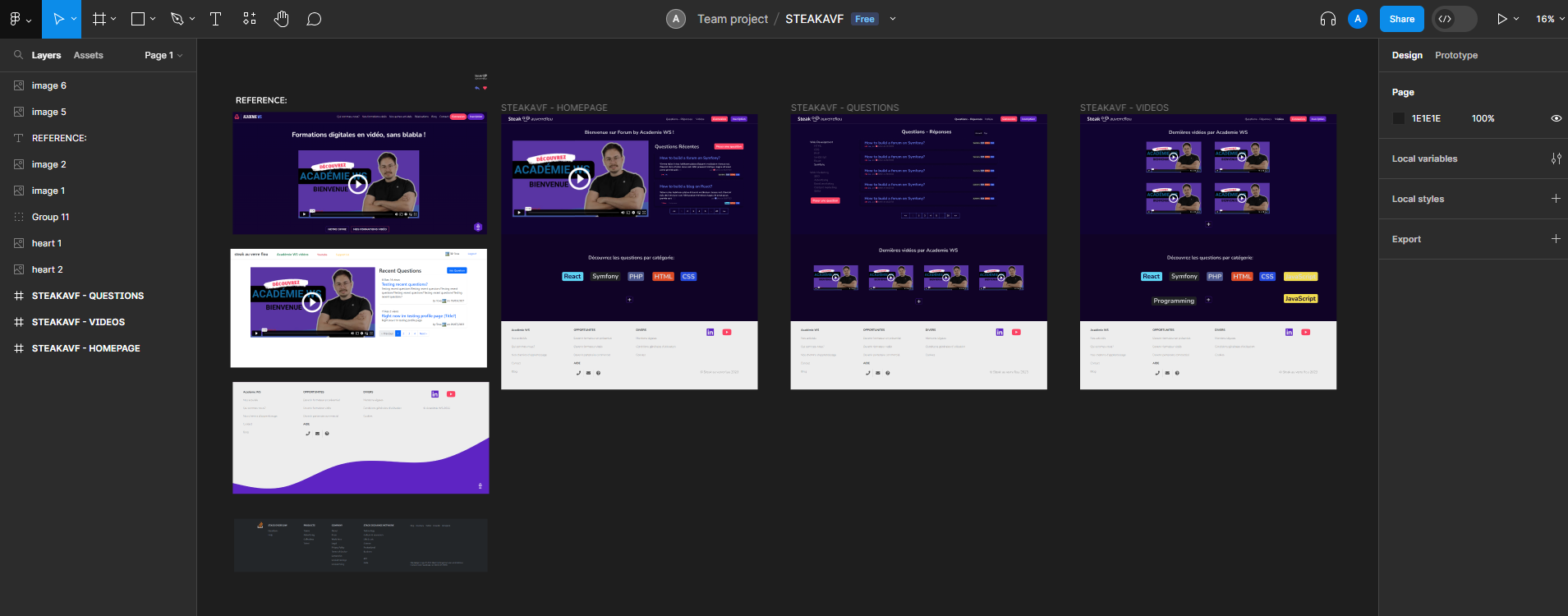
* Outil de Synchronisation de Fichiers: **Rsync** (Rsync est un outil de ligne de commande utilisé pour la synchronisation de fichiers et de répertoires entre différents serveurs ou environnements.)

# **RÉALISATIONS PERSONNELLES**

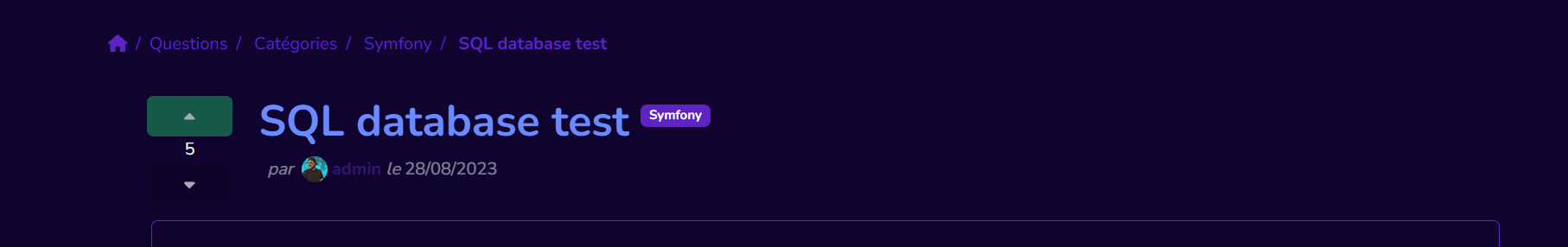
## Exemples de maquettes

J'ai utilisé l'outil **Figma** pour créer les maquettes des interfaces du forum, définissant ainsi son apparence et son expérience utilisateur (UI/UX)

[**lien vers le projet Figma**](https://www.figma.com/file/Y55vkNTl9VuVFSRCbTZ6Mn/STEAKAVF?type=design&node-id=0%3A1&mode=design&t=o0MIf0OSmburyuBO-1)

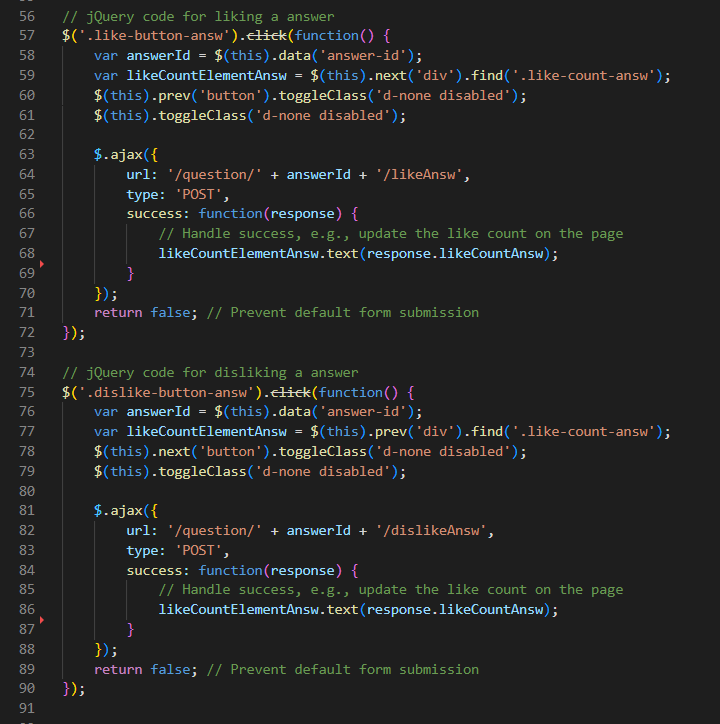


## “Like” et “Dislike” (la gestion des interactions utilisateur pour les questions et les réponses)

J'ai écrit du code **PHP** et utilisé le framework **Symfony** pour créer deux actions distinctes : **"like" et "dislike"**. Ces actions permettent aux utilisateurs de donner leur avis sur les questions et les réponses en les "likes" ou en les "dislikes". J'ai veillé à ce que ces actions soient efficaces en utilisant **l'EntityManager** pour interagir avec la base de données et en renvoyant les résultats au format **JSON** pour une expérience utilisateur fluide.

**likeQuestionAjax:** Cette fonction gère le "like" d'une question via la route **/question/{id}/like**. Elle prend en paramètres **l'ID** de la question, une instance **Request**, et **l'EntityManagerInterface** pour accéder à la base de données. Elle incrémente le compteur de likes de la question et renvoie la nouvelle valeur au format **JSON**.

* **dislikeQuestionAjax:** Cette fonction gère le "dislike" d'une question via la route **/question/{id}/dislike**. Elle fonctionne de manière similaire à **likeQuestionAjax**, mais décrémente le compteur de likes de la question.

J'ai implémenté du code **jQuery** pour gérer les événements de clic sur les boutons **"like" et "dislike"**. Cela permet aux utilisateurs d'interagir facilement avec les questions et les réponses en temps réel, sans nécessiter de rechargement de la page. Lorsqu'un utilisateur clique sur l'un de ces boutons, une requête **AJAX** est envoyée au serveur pour mettre à jour le nombre de "likes" ou de "dislikes" correspondants, et les résultats sont affichés immédiatement à l'écran.

## Récupérer (Fetch) les questions et réponses récentes par utilisateur sur la page de profil

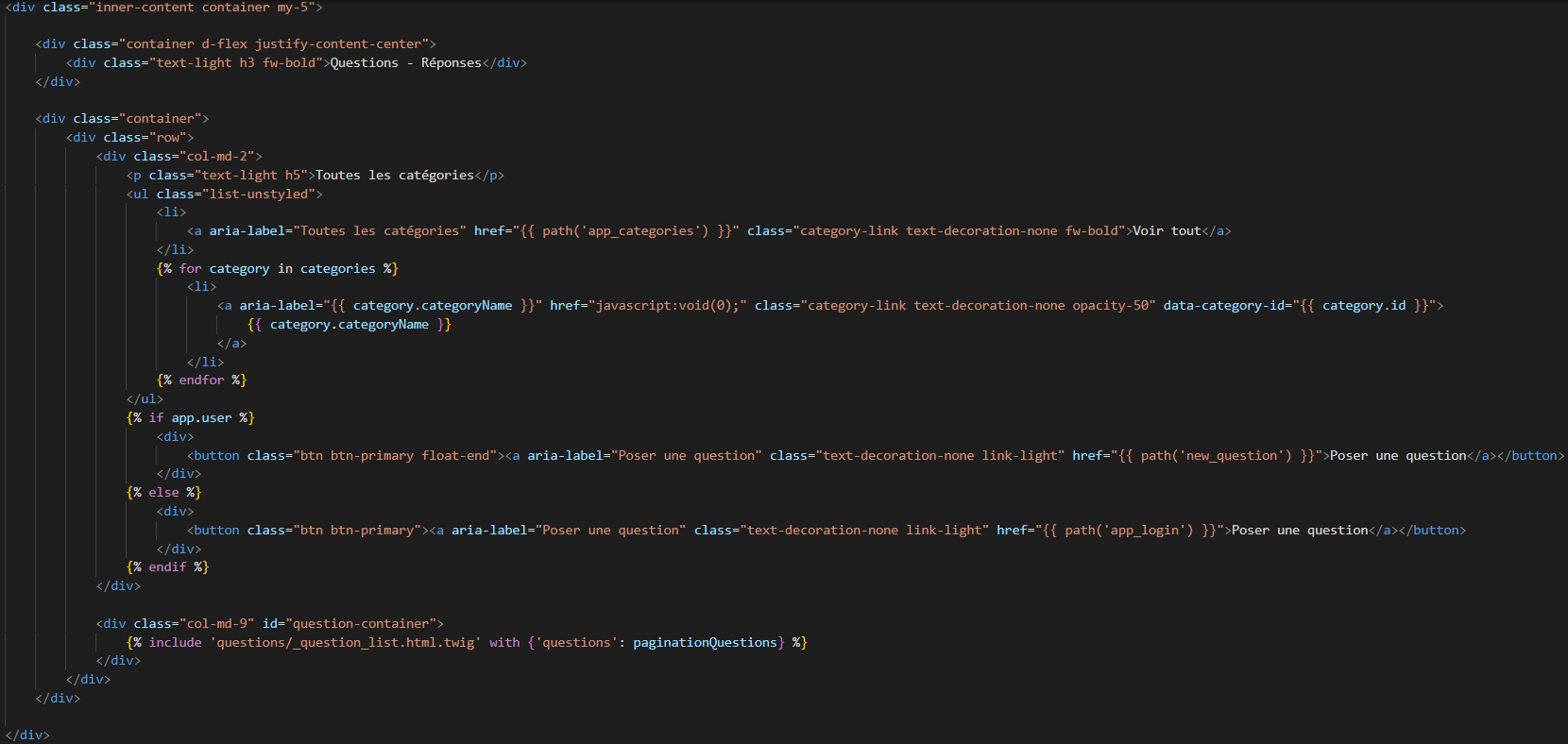
J'ai utilisé **l'EntityManager** pour gérer les interactions avec la base de données. Plus précisément, j'ai obtenu des référentiels (**getRepository**) pour les entités **Question et Answer**. Ensuite, j'ai utilisé la méthode **findBy** pour récupérer les questions et les réponses les plus récentes de l'utilisateur actuellement connecté.

Pour être plus précis, j'ai filtré ces données en fonction de l'utilisateur, trié les résultats par date de création (du plus récent au plus ancien), et limité le nombre d'éléments affichés à 3 au maximum. En utilisant cette approche, j'ai pu obtenir des données spécifiques depuis la base de données pour afficher le profil de l'utilisateur de manière pertinente.

## Filtrage Dynamique par Catégorie

J'ai implémenté une fonctionnalité de sélection de catégorie visant à améliorer l'expérience des utilisateurs de l'application.

* **Identification des Éléments:** J'ai d'abord identifié les liens de catégorie dans le code HTML, en les repérant grâce à leur classe CSS "category-link", ainsi que le conteneur de questions associé.
* **Écouteur d'Événement:** J'ai ajouté un écouteur d'événement "click" à chacun de ces liens de catégorie. Mon objectif était de permettre aux utilisateurs de filtrer les questions en fonction de la catégorie choisie.
* **Gestion du Style:** Lorsqu'un utilisateur clique sur l'un de ces liens de catégorie, j'ai programmé le code pour qu'il réinitialise le style de tous les liens de catégorie. Cela permet de mettre en évidence la catégorie active.
* **Extraction de l'ID de Catégorie:** Pour identifier la catégorie choisie, j'ai extrait l'ID de la catégorie associée à partir de l'attribut "data-category-id" du lien cliqué.
* **Requête AJAX:** Ensuite, j'ai initié une requête AJAX vers une URL spécifique (/questions?category\_id={categoryId}) pour récupérer les questions liées à la catégorie sélectionnée. J'ai veillé à inclure l'en-tête 'X-Requested-With' pour indiquer qu'il s'agissait d'une requête AJAX.
* **Mise à Jour en Temps Réel:** Une fois que j'ai obtenu les données des questions via la requête AJAX, j'ai programmé le code pour mettre à jour dynamiquement le contenu du conteneur de questions, le tout en temps réel et sans nécessiter le rechargement de la page.
* **Affichage des Questions:** J'ai également inclus le code HTML nécessaire pour afficher les questions récupérées, en fournissant des détails tels que le titre, la catégorie, le nombre de "likes", le nombre de vues, le nom de l'utilisateur et la date de création.
* **Pagination:** Enfin, j'ai ajouté une fonctionnalité de pagination pour permettre aux utilisateurs de naviguer facilement entre les différentes pages de questions, garantissant ainsi une expérience utilisateur fluide.



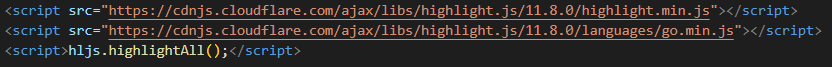
## Intégration de highlight.js

J'ai intégré la bibliothèque "**highlight.js**", une étape cruciale pour améliorer la présentation de notre contenu, en particulier lorsque nous affichons des extraits de code.





**Importation de la Bibliothèque:** J'ai commencé par importer la bibliothèque "highlight.js" dans notre projet en incluant le lien vers le fichier CSS de "highlight.js". Ce fichier contient les styles nécessaires pour la coloration syntaxique du code.

**Activation de la Coloration Syntaxique:** J'ai ajouté un script JavaScript pour activer la coloration syntaxique. Voici un exemple de code JavaScript que j'ai utilisé pour cela

**Utilisation dans nos Questions et Réponses:** Lorsque nous affichons des questions ou des réponses contenant du code, j'ai veillé à ce que le contenu soit encapsulé dans des balises **<pre>** et **<code>**

Cela garantit que "**highlight.js**" est activé pour ces extraits de code, améliorant ainsi la lisibilité de notre contenu.

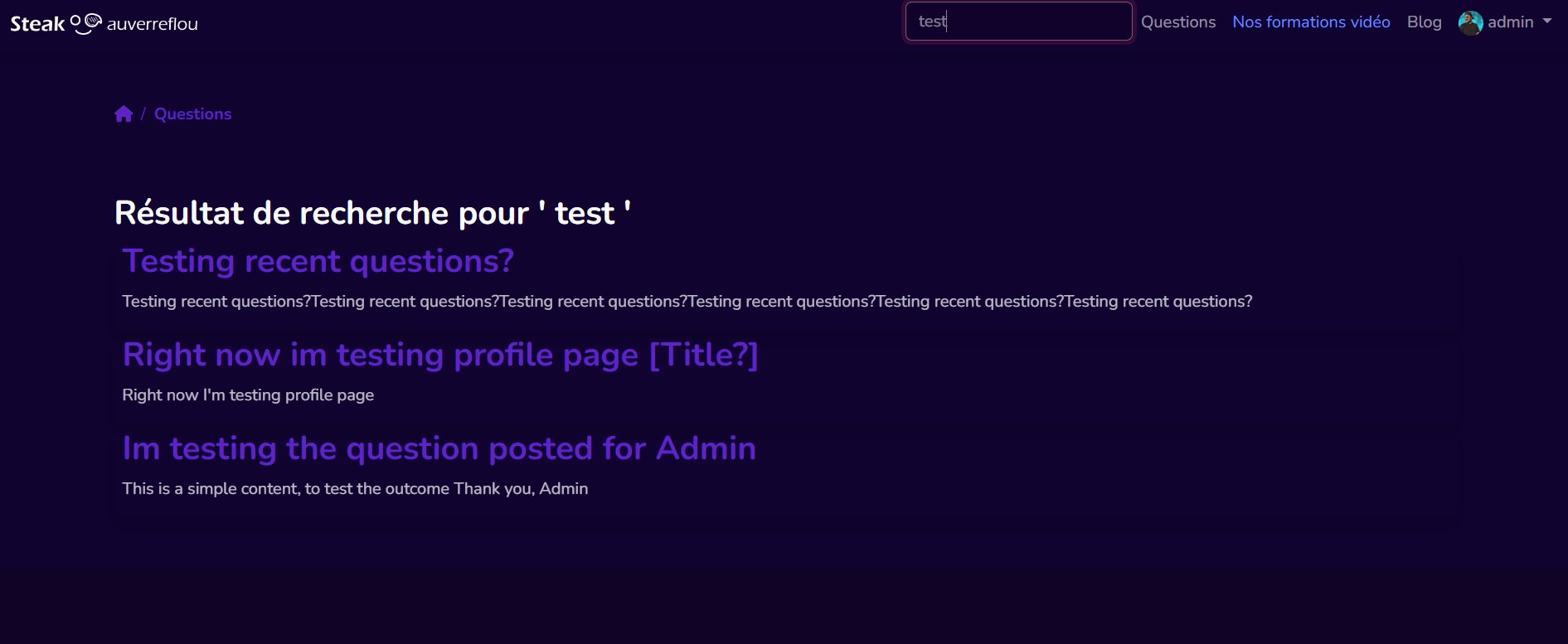
## Barre de recherche

**Animation de la Barre de Recherche:** Lorsque l'utilisateur clique sur le bouton de recherche (".search-button"), la barre de recherche (".search-input") s'anime et change de largeur. Cela crée une expérience utilisateur fluide.

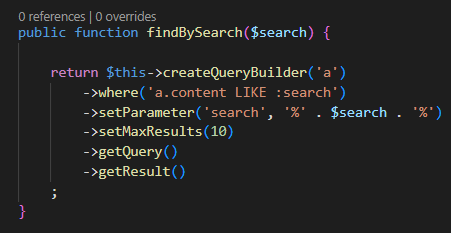


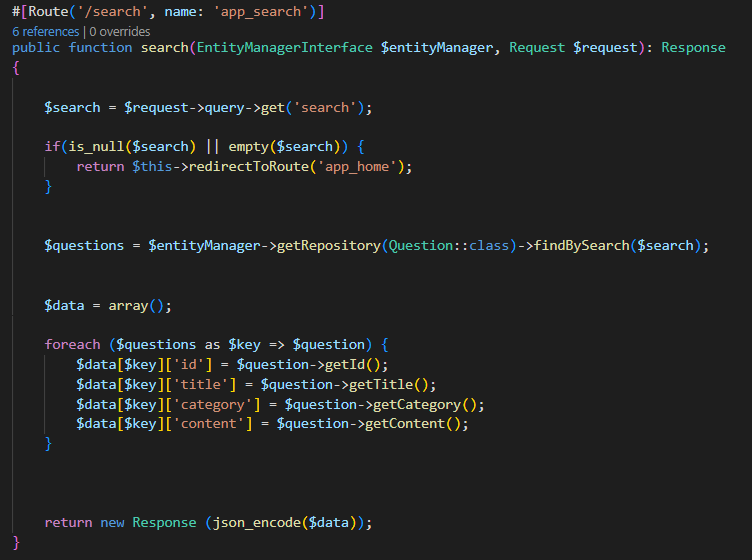


**Recherche en Temps Réel:** Lorsque l'utilisateur appuie sur la touche "Enter" dans la barre de recherche ("#search-content"), une requête AJAX est déclenchée vers l'URL '/search' avec les termes de recherche saisis. Les résultats sont récupérés au format JSON et affichés en temps réel.

**Affichage des Résultats:** Les résultats de recherche sont formatés en HTML dans la variable "contentHtml" et remplacent le contenu existant de la classe "inner-content". Cela permet à l'utilisateur de voir instantanément les résultats de sa recherche.

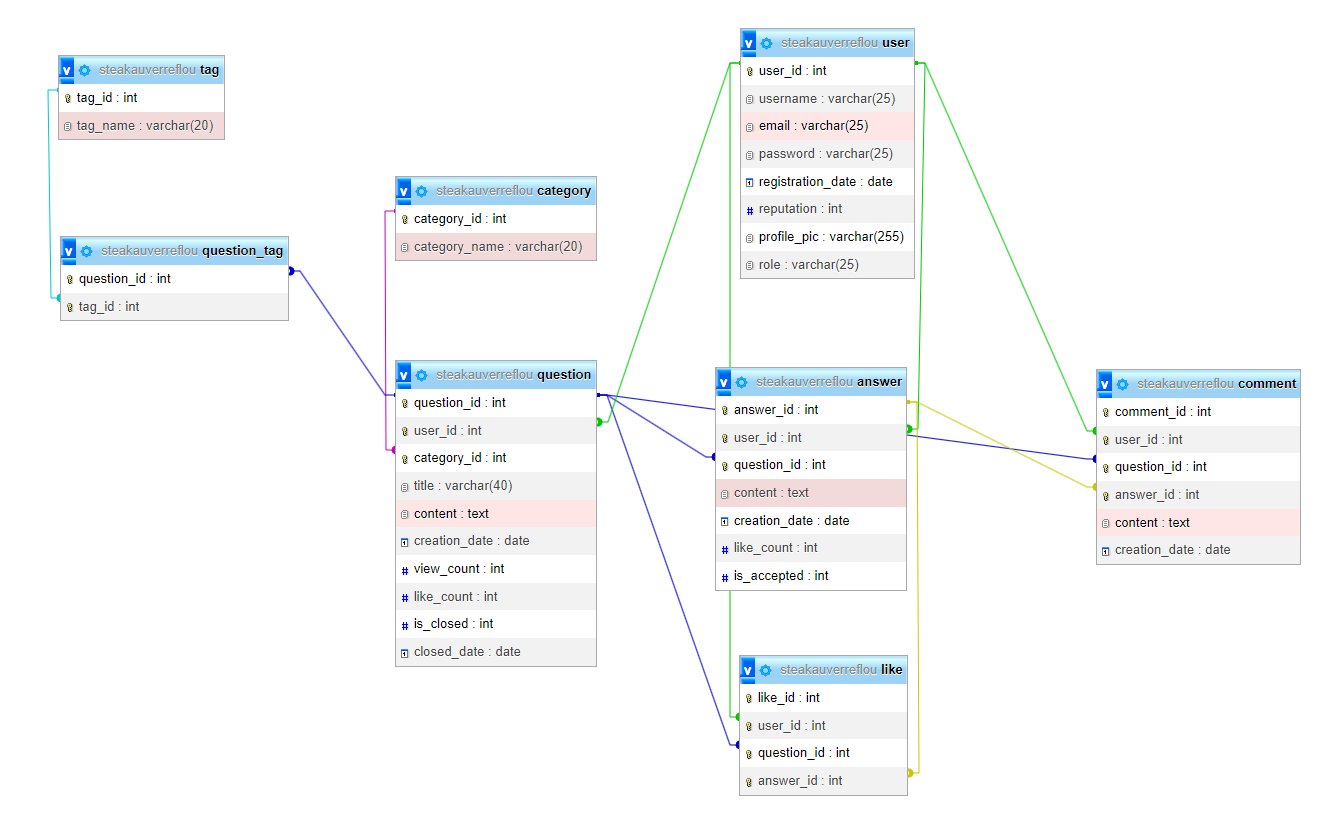
**Requête Backend:** J'ai également écrit le code PHP côté serveur, que j'ai placé dans la classe "**SearchController**", pour gérer la recherche. Ce code effectue une interrogation de la base de données en utilisant les termes de recherche fournis et renvoie les résultats au format **JSON** en réponse à la requête **AJAX** émise depuis le côté client.

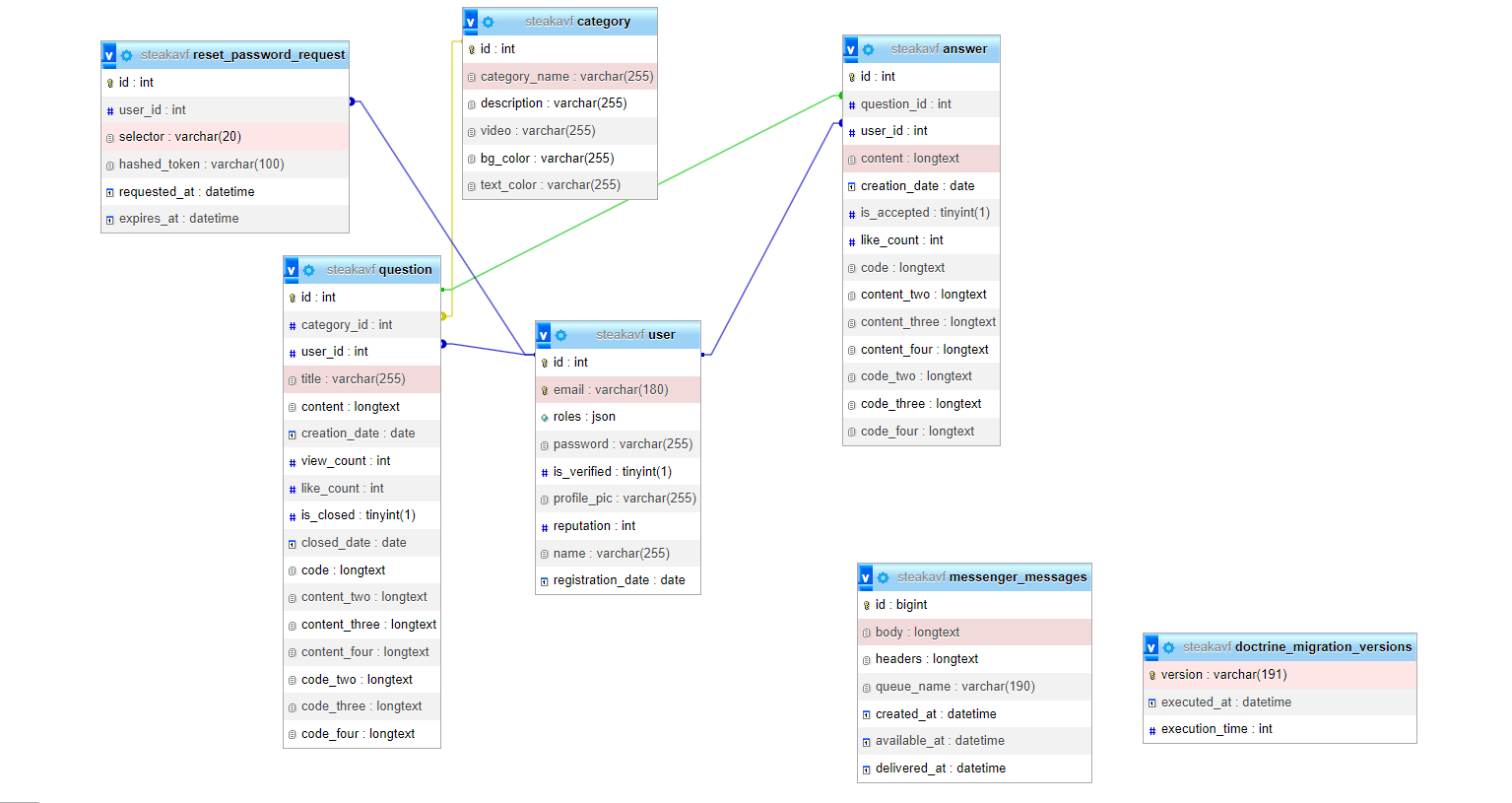




## MCD

J'ai utilisé **PHPMyAdmin Designer** pour concevoir la structure de la base de données en suivant le modèle conceptuel de données (**MCD**)

**Start:**



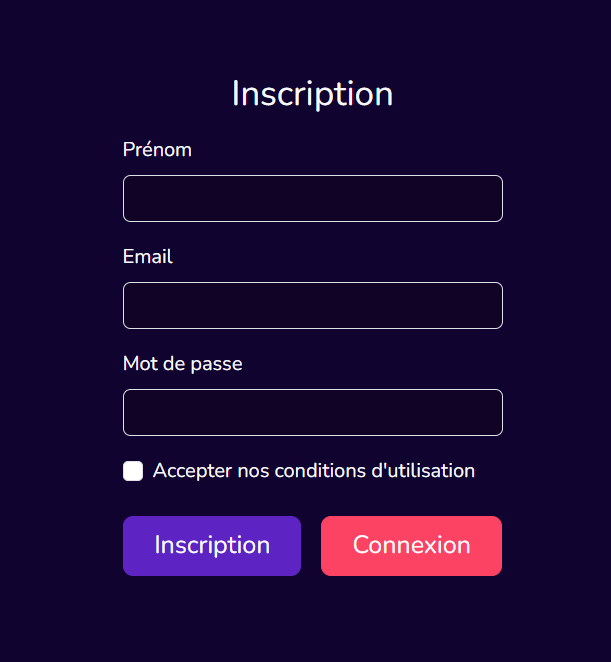
**Final:**

# **PRÉSENTATION DU JEU D’ESSAI**

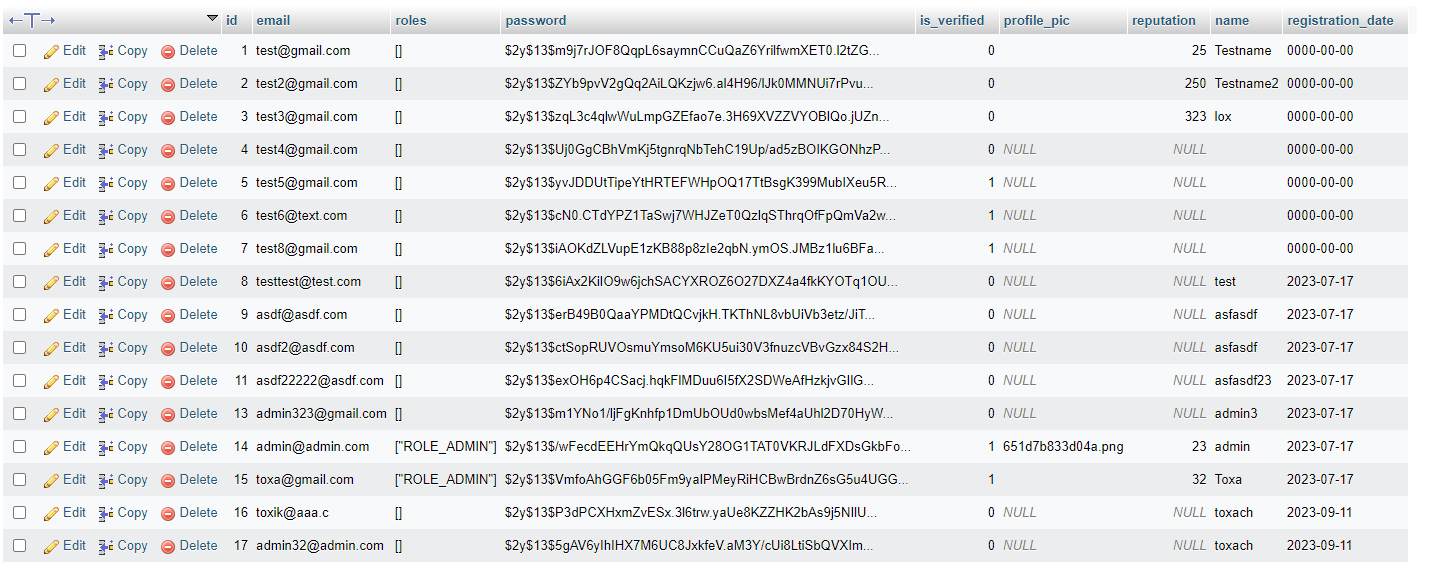
## Utilisation de Jeux de Données

Après la réalisation de chaque fonctionnalité, j'ai utilisé des jeux de données spécifiques pour tester. Au lieu de générateurs automatiques, j'ai créé manuellement des jeux de données pour simuler des scénarios réels.

## Tests des Fonctionnalités Utilisateurs

Par exemple, pour la création d'utilisateurs et l'inscription, j'ai utilisé des jeux de données pour vérifier que tout fonctionnait correctement, y compris la validation des données.

## 



## 

## 

## 

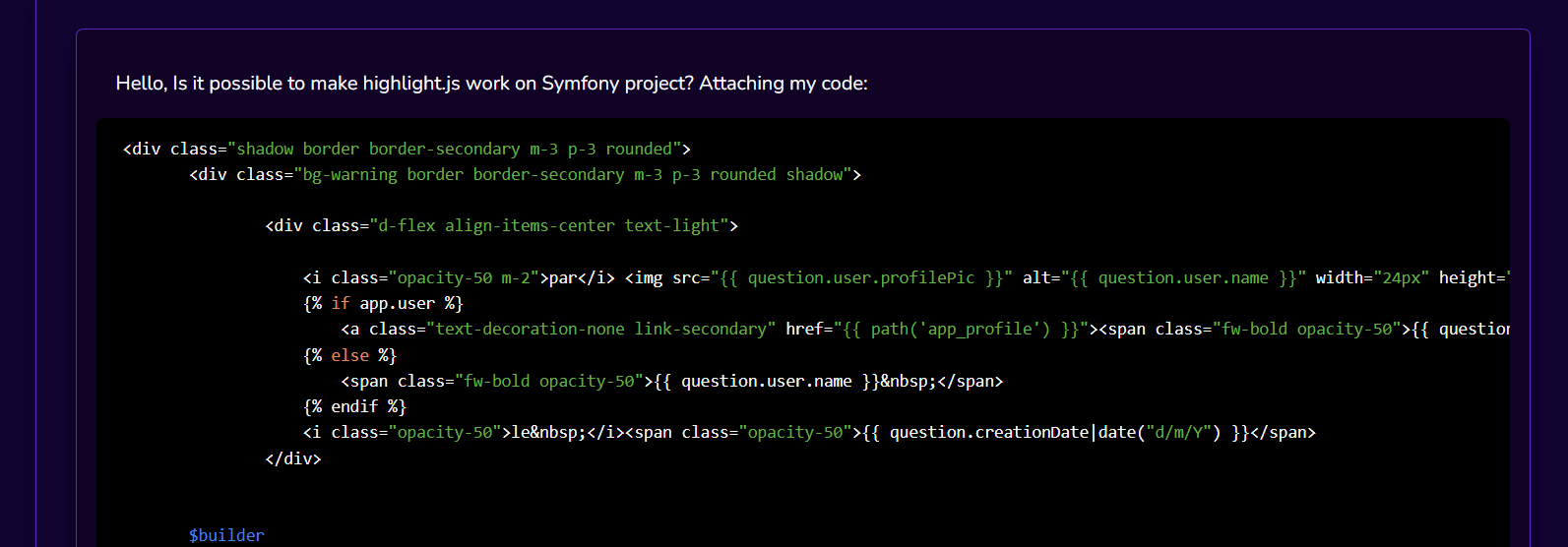
## Tests des Formulaires

J'ai testé les formulaires de questions, de catégories et de réponses en utilisant des jeux de données pour vérifier la soumission et l'affichage des données.

## 

## 

## Tests de Highlight.js

Lors de l'intégration de **Highlight.js** pour la mise en évidence de la syntaxe dans les contenus, j'ai effectué des tests en utilisant divers langages de programmation. Cette étape était essentielle pour garantir que **Highlight.js** interprétait correctement les différents langages et affichait la syntaxe de manière précise dans les questions et les réponses.

# **VULNÉRABILITÉS DE SÉCURITÉ ET VEILLE**

## SÉCURITÉ

Pour garantir la protection des données sensibles, j'ai mis en place plusieurs mesures de sécurité.

### Hachage des Mots de Passe

* J'ai utilisé des techniques de hachage pour stocker en toute sécurité les mots de passe des utilisateurs. Cela garantit que les informations sensibles restent confidentielles même en cas d'accès non autorisé à la base de données.

### Utilisation du Bundle Symfony Security

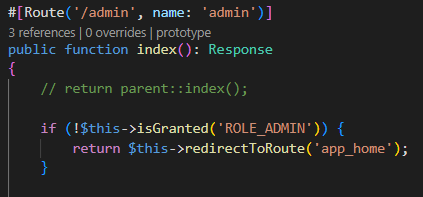
* J'ai intégré le bundle de sécurité Symfony, ce qui m'a permis de bénéficier de nombreuses fonctionnalités de sécurité intégrées, telles que l'authentification, l'autorisation, et la gestion des sessions.

### 

### Gestion des Inscriptions avec Symfony

* J'ai utilisé les fonctionnalités de gestion des inscriptions fournies par Symfony pour m'assurer que seuls les utilisateurs autorisés peuvent créer des comptes. Cela limite les risques de comptes non authentifiés.

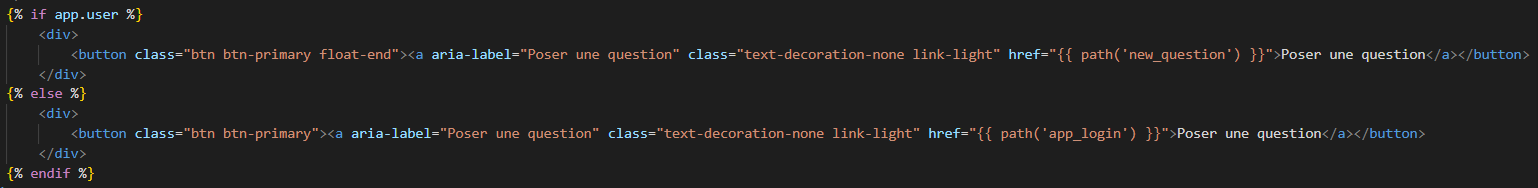
### **Attribution du Rôle "ROLE\_ADMIN"**

* Pour sécuriser l'accès au backend de l'application, j'ai attribué le rôle "ROLE\_ADMIN" aux utilisateurs autorisés. Cela leur donne un accès spécial avec des privilèges étendus, tandis que les utilisateurs standard ont un accès limité.

### 

### 

### **Vérifications avec {% if app.user %}**

* J'ai utilisé des conditions Twig, comme **{% if app.user %}**, dans certaines parties de l'interface utilisateur pour m'assurer que seuls les utilisateurs enregistrés peuvent effectuer des actions spécifiques, comme la publication de questions, de réponses ou la mise en favoris.

En effectuant cette veille sur les vulnérabilités de sécurité, j'ai pu renforcer la robustesse de l'application et garantir que les données des utilisateurs étaient protégées contre les menaces potentielles.

## VEILLE

Pour moi, la "veille informatique" est essentielle dans le domaine du développement web. C'est une pratique qui consiste à rester constamment informé des dernières avancées technologiques, des nouvelles tendances et des meilleures pratiques. Dans cet essai, je vais partager mes propres méthodes pour mener à bien cette veille informatique, en me basant sur ma propre expérience.

### Recherche Google et Stack Overflow

Lorsque je décide de mener des recherches, ma première étape consiste généralement à utiliser Google. J'opte pour l'anglais comme langue principale de recherche, car il offre une pléthore de contenus dans le domaine de la technologie. Google me dirige souvent vers des ressources précieuses, et l'une de mes destinations préférées est Stack Overflow. Cette plateforme est une véritable mine d'or pour obtenir des réponses à mes questions de codage et de programmation.

### Chaînes YouTube

Pour rester au sommet des actualités en développement web, je suis abonné à plusieurs chaînes YouTube spécialisées. Celles-ci m'aident à découvrir les dernières nouvelles, les tendances émergentes et les nouvelles technologies. J'apprécie particulièrement la manière dont les vidéos visuelles et les tutoriels de ces chaînes rendent l'apprentissage plus engageant et interactif. Parmi mes chaînes préférées, on compte **Fireship, Beyond Fireship**.

### Podcast Spotify - "Syntax"

Lorsque je suis en déplacement, que ce soit en voiture ou en train, ou lorsque je fais de l'exercice, j'aime écouter des podcasts pour rester informé. L'un de mes podcasts préférés est "Syntax" sur Spotify. En anglais, ce podcast couvre non seulement les actualités du développement web, mais aussi des faits intéressants et les sujets brûlants de l'industrie. C'est une manière idéale de rester connecté au monde du développement web, même lorsque je suis en déplacement.

### Avantages de la Veille Informatique

Pour moi, la veille informatique est plus qu'une simple habitude. C'est un moyen de rester compétitif dans mon domaine. Grâce à cette pratique, je suis en mesure de suivre le rythme des technologies en constante évolution, d'appliquer les meilleures pratiques et de rester pertinent en tant que développeur web.

### Défis et Tendances Futures

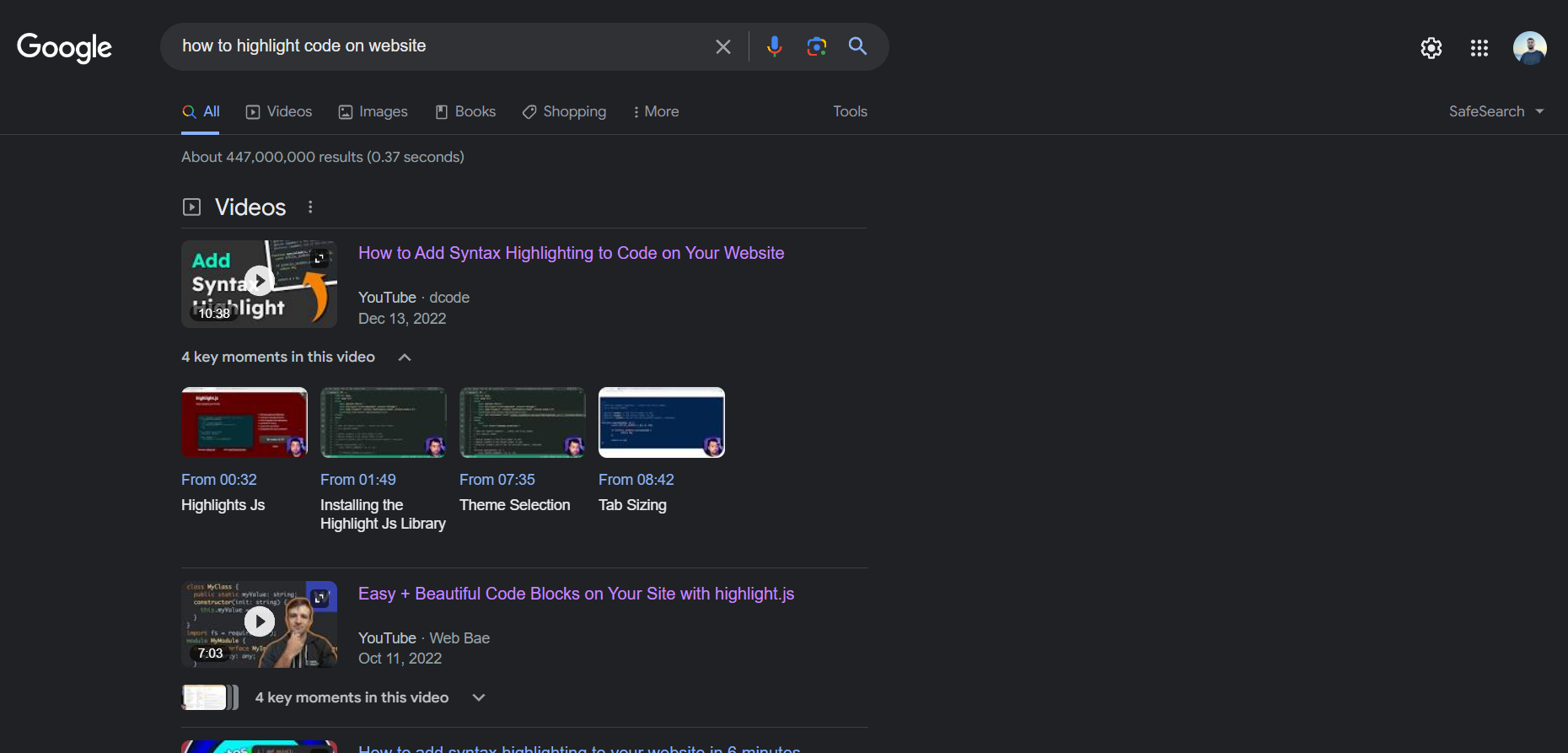
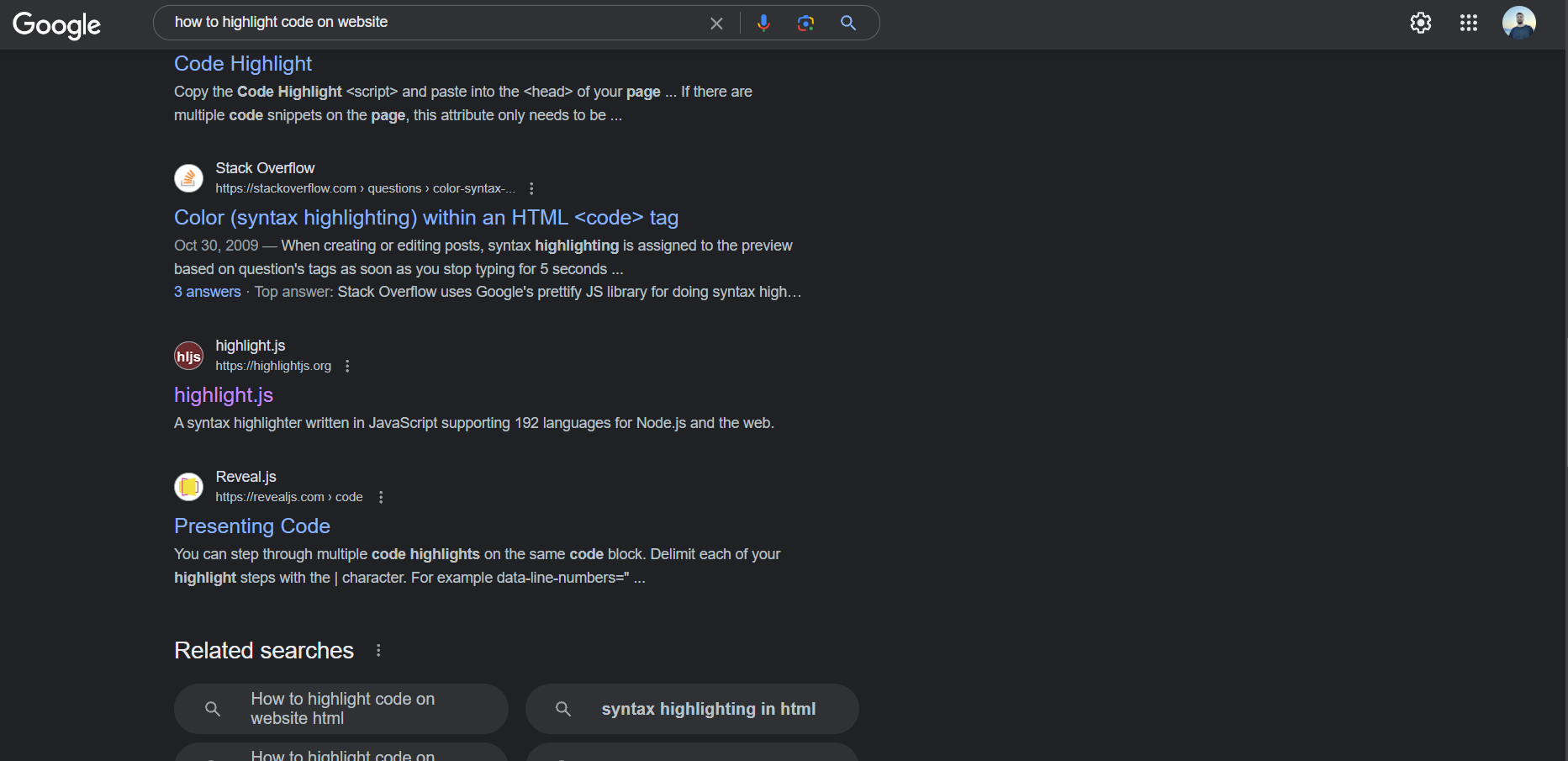
Cependant, je reconnais que la veille informatique peut être un défi, car l'industrie évolue rapidement. Dans l'avenir, je m'attends à ce que de nouvelles plateformes et outils émergent, ce qui pourrait nécessiter une adaptation constante de mes méthodes de veille.

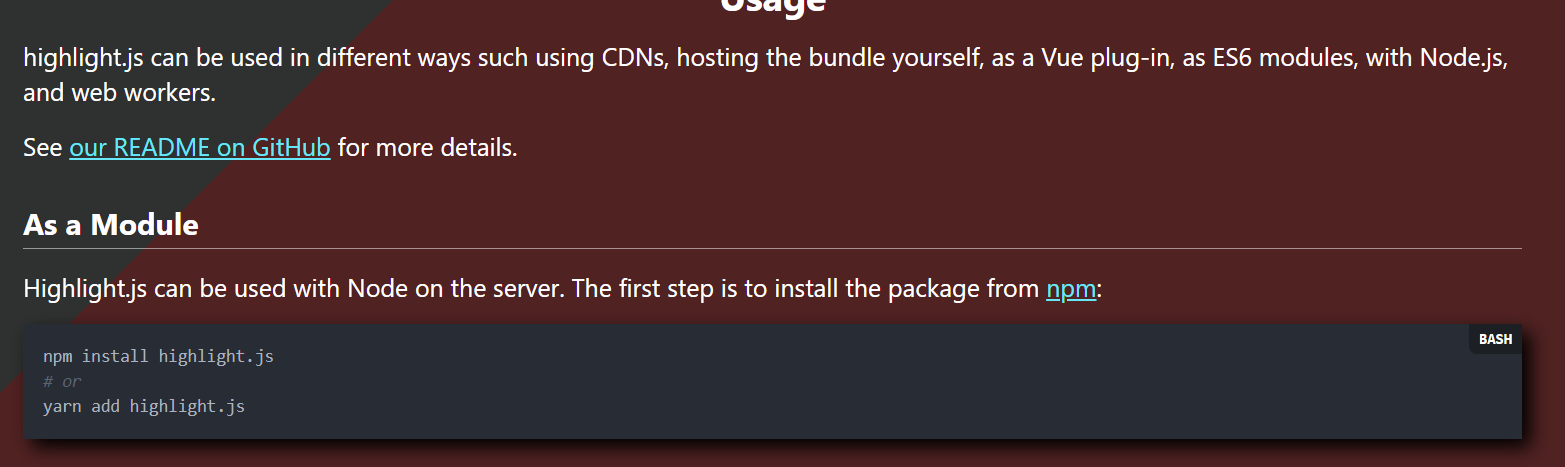
### 

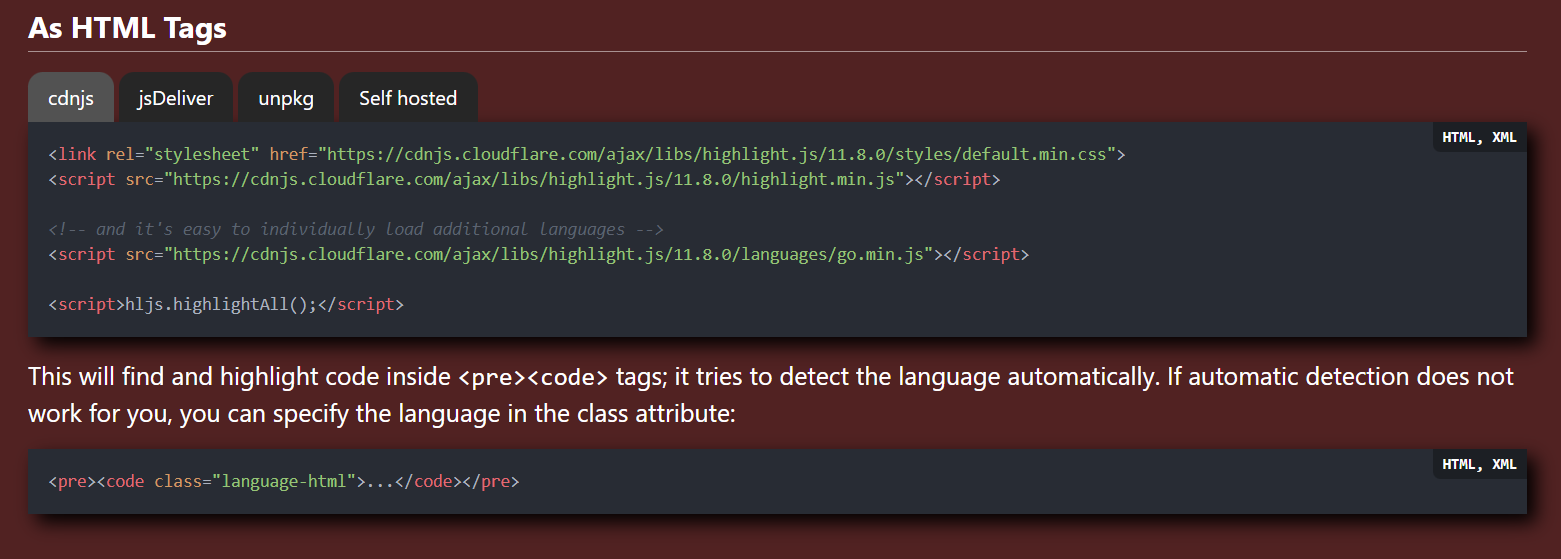
En conclusion, la veille informatique est cruciale pour moi en tant que développeur web. Elle me permet de rester à jour dans un domaine en constante évolution et d'améliorer continuellement mes compétences. J'encourage vivement d'autres professionnels de l'informatique à adopter des stratégies similaires pour rester informés et compétitifs dans notre domaine en perpétuelle mutation.

# **SITUATION DE TRAVAIL AYANT NÉCESSITÉ UNE RECHERCHE + EXTRAIT DU SITE**

En général, lorsque je fais des recherches, je vais sur Google, je saisis ma requête en anglais, puis je regarde les options proposées et j'expérimente différentes possibilités.

Un exemple concret: récemment, j'avais besoin de savoir comment "how to highlight code on website". C'est ainsi que j'ai découvert **highlight.js**. J'ai regardé quelques vidéos et lu la documentation officielle, ce qui m'a donné des idées pour l'intégrer dans mon projet.



Ceci est une capture d'écran de la page d'accueil de **highlight.js**, qui nous indique comment l'installer.highlight.js peut être utilisé de différentes manières, notamment en utilisant des CDNs, en hébergeant vous-même le bundle, en tant que plug-in Vue, en tant que modules ES6, avec Node.js et avec des web workers.

Cette capture d'écran nous indique comment installer highlight.js en utilisant **cdnjs** comme balise **HTML**

Inclure la feuille de style

<link rel="stylesheet"href="<https://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.8.0/styles/default.min.css>">

Inclure la bibliothèque JavaScript

<script src="<https://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.8.0/highlight.min.js>"></script>

Appeler la fonction pour mettre en “highlight” le code

<script>hljs.highlightAll();</script>

**CONCLUSION**

Ce projet a été une excellente occasion pour améliorer mes compétences techniques, apprendre à travailler en équipe avec l'Académie WS, et aider à faire grandir leur plateforme de formation en ligne. Je suis reconnaissant d'avoir eu cette expérience et je pense que les utilisateurs de l'Académie WS tireront de grands avantages de l'ajout de ce forum à leur site.

Je tiens également à exprimer ma gratitude envers notre formateur, M. Samih Habbani, dont les conseils et l'expertise ont été essentiels tout au long de ce projet. Son soutien a été inestimable.

Je souhaite également remercier chaleureusement l'ensemble de mes collègues, qui ont travaillé en étroite collaboration pour atteindre nos objectifs. Leur dévouement et leur engagement ont été remarquables.

Enfin, je voudrais exprimer ma reconnaissance envers la région et l'AFPA pour cette opportunité qui nous a permis de participer à la formation DWWM.