

## Évaluation CCP 2: Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité

- 1) Télécharger l'archive eval.zip
- 2) Analysez les pages web et préparer la liste des tâches à réaliser
- 3) Concevoir le modèle conceptuel de donnée
- 4) Isoler les scripts javascript dans des fichiers .js (sauf ajax si symfony)
  - a) Attention un des scripts bloque les formulaires a vous de trouver lequel
- 5) Concevoir la partie Backend du site:

Le but de l'évaluation est de fournir un site de e-commerce complet  
(sans la partie paiement l'action payer validera automatiquement la commande il y aura cependant le choix du type de paiement: PayPal ou CB a prendre en compte)  
Pour cela vous devrez a partir du design existant créer les pages nécessaire a la gestion du site (partie admin devant avoir un design différent, vous pouvez prendre un template admin bootstrap et l'adapté a vos besoins)  
Vous aurez aussi besoin de gérer la partie front office

### Partie Admin:

- a) Gestion des genres
- b) Gestions des catégorie
- c) Gestion des images (galerie d'image)
- d) Gestion des articles
- e) Gestion des comptes clients
- f) Gestion des commandes
- g) Réponses au formulaire de contact

### Partie Client:

- a) Inscription
- b) Connexion
- c) Liste des genres
- d) Liste des catégories
- e) Liste des articles (vous pouvez vous aidez de site existant pour le design)
- f) Gestion du panier
- g) Gestion des commandes
- h) Gestion de son compte (modif email, mot de passe etc...)
- i) Formulaire de contact

Le contenu du site devra être éditable depuis l'interface admin (titre du site, images sections etc...)

La page CGU et CGV devra être ajoutée et éditable avec un éditeur wysiwyg (CKEditor ou FOSCKEditor pour symfony) Pour CKEditor voir la documentation en page 2.

Le choix des technologies à utiliser est à votre libre décision, symfony n'est pas requis pour cette évaluation cependant libre à vous de faire cet évaluation avec. L'utilisation de composer et de PSR-4 dans le cas où vous n'utilisez pas symfony est un plus. L'utilisation d'AJAX pour l'ajout d'un produit au panier est un point bonus. L'utilisation des supports de cours est autorisée cependant un copier collé depuis le code d'un autre stagiaire est éliminatoire.

Le résultat final de l'évaluation devra être posté sur github (en répertoire privé) avec comme membres «RomualdLewandoski» et «rudy62800»

Il devra au minimum y avoir un commit par jour afin d'attester du travail effectué la journée

Attention à bien respecter les normes de sécurité les failles de types XSS ET SQL doivent impérativement être bloquées.

Un utilisateur anonyme n'a pas accès à l'espace client, un client n'a pas accès au panel d'administration.

Le modèle conceptuel de données devra être présent dans le git sous forme d'image ou au format HTML (cas de draw.io)

#### DOCUMENTATION :

CKEDITOR Sans symfony:

Prérequis chargement de JQuery AVANT CKeditor  
CDN :

```
<script src="https://cdn.ckeditor.com/4.15.1/standard/ckeditor.js"></script>
```

Créez une textarea avec un id puis créer un script

```
<textarea name="editor1" id="editor1"></textarea>
```

```
<script> CKEDITOR.replace( 'editor1' ); </script>
```

ATTENTION changer le name/id du text area en fonction de vos besoins (et ne pas oublier de changer aussi le nom dans la fonction replace() en dessous.

Pour plus d'information cf documentation de CKEditor 4 (attention il s'agit de la version 4 et non de la version 5 )