

Créez mon premier blog en PHP

Projet 3 - Parcours développeur d'application PHP - OpenClassrooms

Installation sur serveur

Une fois le site déployé, le dossier racine doit être /public. Faire pointer le *virtual host* sur celui-ci.

Connexion à la BDD

Le site utilise une base de données afin d'enregistrer les articles, commentaires, comptes utilisateur etc.

- Utilisez le fichier *blog_kz.sql* pour la création de la BDD et son remplissage.
- Entrez vos paramètres de connexion à la Bdd dans le fichier 'MVC/models/Model.php', dans la méthode setBdd.

Configurer l'envoi de mail

- Afin de rendre fonctionnels le formulaire de contact, veuillez renseigner le mail du SuperAdmin, dans le fichier 'controllers/ControllerPost.php' dans la méthode contactMessage

Vous pouvez désormais utiliser le blog.

Thème Bootstrap :

<https://startbootstrap.com/theme/clean-blog>

Pour consulter mon blog:

<https://github.com/knouz>

Description du besoin

Le projet est de développer votre blog professionnel. Ce site web se décompose en deux grands groupes de pages :

- les pages utiles à tous les visiteurs ;
- les pages permettant d'administrer votre blog.

Voici la liste des pages qui devront être accessibles depuis votre site web :

- la page d'accueil ;
- la page listant l'ensemble des blog posts ;
- la page affichant un blog post ;
- la page permettant d'ajouter un blog post ;
- la page permettant de modifier un blog post ;
- les pages permettant de modifier/supprimer un blog post ;
- les pages de connexion/enregistrement des utilisateurs.

Vous développerez une partie administration qui devra être accessible uniquement aux utilisateurs inscrits et validés.

Les pages d'administration seront donc accessibles sur conditions et vous veillerez à la sécurité de la partie administration.

Les pages utiles à tous les internautes.

Sur la page d'accueil, il faudra présenter les informations suivantes :

- votre nom et votre prénom ;
- une photo et/ou un logo ;
- une phrase d'accroche qui vous ressemble (exemple : "Martin Durand, le développeur qu'il vous faut !") ;
- un menu permettant de naviguer parmi l'ensemble des pages de votre site web ;
- un formulaire de contact (à la soumission de ce formulaire, un e-mail avec toutes ces informations vous sera envoyé) avec les champs suivants :
 - nom/prénom,
 - e-mail de contact,
 - message,
- un lien vers votre CV au format PDF ;
- l'ensemble des liens vers les réseaux sociaux où l'on peut vous suivre (GitHub, LinkedIn, Twitter...).

Sur la page listant tous les blogs posts (du plus récent au plus ancien), il faut afficher les informations suivantes pour chaque blog post :

- le titre ;
- la date de dernière modification ;
- le châpo ;
- et un lien vers le blog post.

Sur la page présentant le détail d'un blog post, il faut afficher les informations suivantes :

- le titre ;
- le chapô ;
- le contenu ;
- l'auteur ;
- la date de dernière mise à jour ;
- le formulaire permettant d'ajouter un commentaire (soumis pour validation) ;
- les listes des commentaires validés et publiés.

Sur la page permettant de modifier un blog post, l'utilisateur a la possibilité de modifier les champs titre, chapô, auteur et contenu.

Dans le footer menu, il doit figurer un lien pour accéder à l'administration du blog.

Contraintes

Nous n'utiliserons pas WordPress. Tout sera développé par vos soins. Les seules lignes de code qui pourront provenir d'ailleurs seront celles du thème Bootstrap, que vous prendrez grand soin de choisir. La présentation, ça compte ! Il est également autorisé d'utiliser une ou plusieurs librairies externes à condition qu'elles soient intégrées grâce à [Composer](#).

Attention, votre blog doit être navigable aisément sur un mobile (téléphone mobile, phablette, tablette...). C'est indispensable !

Nous vous conseillons vivement d'utiliser un moteur de templating tel que Twig, mais ce n'est pas obligatoire.

Sur la partie administration, vous veillerez à ce que seules les personnes ayant le droit "administrateur" aient l'accès ; les autres utilisateurs pourront uniquement commenter les articles (avec validation avant publication).

Important : Vous vous assurerez qu'il n'y a pas de failles de sécurité (XSS, CSRF, SQL Injection, session hijacking, upload possible de script PHP...).

Votre projet doit être poussé et disponible sur GitHub. Je vous conseille de travailler avec des pull requests. Dans la mesure où la majorité des communications concernant les projets sur GitHub se font en anglais, il faut que vos commits soient en anglais.

Vous devrez créer l'ensemble des issues (tickets) correspondant aux tâches que vous aurez à effectuer pour mener à bien le projet.

Veillez à bien valider vos tickets pour vous assurer que ceux-ci couvrent bien toutes les demandes du projet. Donnez une estimation indicative en temps ou en points d'effort (si la méthodologie agile vous est familière) et tentez de tenir cette estimation.

L'écriture de ces tickets vous permettra de vous accorder sur un vocabulaire commun. Il est fortement apprécié qu'ils soient écrits en anglais !