

PHP

PROJET NOTÉ

Exercice de fin de parcours

PHP – PROJET NOTÉ

Créons notre version simplifiée de twitter

Projet par groupe de 2

Maintenant que nous avons appris à coder une base en PHP pour faciliter le développement de sites, nous allons mettre à l'épreuve nos capacités en créant un projet.

Le but de ce projet est de créer un « Clone » de twitter dans une version simplifiée. Vous trouverez ci dessous les détails des fonctionnalités à implémenter :

- **Système de gestion utilisateur** : L'utilisateur doit être en capacité de faire les choses suivantes :
 - Créer un compte
 - Se connecter avec ses identifiants
 - Avoir accès à un espace personnel
 - Mettre à jour ses informations personnelles
 - Disposer d'une page de profil publique où s'affichent ses récents tweets et retweets dans un ordre chronologique
 - Un utilisateur peut suivre un autre utilisateur pour voir son contenu dans sa timeline
- **Système de Tweets** : L'utilisateur doit pouvoir écrire des Tweets d'un nombre de caractère limité à 140.
 - Les tweets doivent être visible seulement par les personnes qui suivent la personne qui envoie le tweet
 - Une personne doit pouvoir retweeter un tweet
 - Une personne doit pouvoir liker le tweet
 - Seul la personne ayant créé le tweet peut le supprimer / modifier
- **Système de timeline** : L'utilisateur doit pouvoir voir les tweets écrits par les personnes qu'il suit
 - Sur la page d'accueil lorsque l'utilisateur est connecté, il doit voir les derniers tweets des personnes qu'il suit (Dans un ordre chronologique)
 - Il doit pouvoir remonter dans sa timeline pour voir des tweets plus anciens

Il est attendu dans ce sujet que vous utilisiez le petit framework que nous avons développé en cours.

Si vous vous sentez courageux vous pouvez utiliser d'autres frameworks, mais ce sera à vos risques et périls, voilà pourquoi :

PHP – PROJET NOTÉ

- Vous ne bénéficierez pas de mon aide : Si vous prenez un framework vous serez livrés à vous même
- Un framework peut faire appel à des notions non présentes dans les cours de cette année
- Un framework va peut être demander une installation plus complexe
- Le déploiement sur serveur de ces frameworks sera différent de celui vu en cours

Si vous choisissez cette voie, voilà les frameworks que je préconise :

- Slim : <http://www.slimframework.com/>
- Symfony : <https://symfony.com/>
- Laravel : <https://laravel.com/>

Néanmoins, si vous choisissez de **partir avec un autre framework**, vous aurez **deux points d'audace** accordés par défaut, **mais faut-il encore rendre quelque chose de fonctionnel...**

Vous serez évalués sur les critères suivants :

- **Schéma de base de données cohérent** [10% de la note]
- **Respect des notions vues en CMs et TPS** [30% de la note]
- **Développement des nouvelles fonctionnalités au format MVC** [30% de la note]
- **Sécuriser son code pour éviter les attaques de base** [30% de la note]
- **Bonus** [10% de la note]

Vous trouverez ci dessous les différents critères d'évaluation :

Schéma de base de données cohérent

Dans cette partie, vous allez être noté sur votre capacité à créer une structure de base de donnée permettant de faciliter le développement de votre App PHP.

N'hésitez donc pas à faire des MCD, Modèle logique et relationnel avant de vous lancer dans le développement !

Respect des notions vues en CMs et TPS

Les notions que nous avons vues dans les CMs et Tes sont à appliquer sur ce projets, je vous les ai regroupé les points ci dessous :


PHP - PROJET NOTÉ

- Bonnes pratiques PHP
- Comprendre les URI pour former ses routes de manière logique
- Utiliser les bons verbes pour les bonnes actions à réaliser
- Découper ses fonctions ou classes utilitaires en services dès que c'est possible
- Essayer au maximum de ne pas répéter son code
- Commenter votre code
- Indenter correctement son code afin qu'il soit lisible

Développement des nouvelles fonctionnalités au format MVC

Pour cette partie, sera examiné la façon dont vous avez développé les fonctionnalités demandées.

Il est attendu que vous utilisez le format MVC, afin de bien séparer la partie Logique, la partie données, ainsi que la partie affichage.

 **Attention**, comme dit en CM et répété en TP, je ne doit pas voir de code PHP servant à autre chose que de l'affichage dans la partie template / View de votre app.

Sécuriser son code pour éviter les attaques de base

Cette partie sera testée en premier sur votre Site, des attaques de type **XSS** et **injection SQL** seront réalisées sur votre site.

Si ces attaques sont réussies, je ne pourrais pas continuer à noter votre site car il ne sera plus fonctionnel.

Sans rien pour tester votre site, je serai dans l'obligation de noter sur ce que j'ai vu jusque là, et donc avoir une vision très limitée de votre app PHP.

Voilà quelques rappels concernant la sécurité (Je vous laisse aller voir le CM2 qui en parle pour plus d'informations)

- Vous ne devez jamais faire confiance à une donnée provenant de l'utilisateur, il faut toujours la valider avant de faire quoi que ce soit de notre côté
- Utilisez le binding lors que vos requêtes MYSQL avec PDO pour éviter les surprises
- Échappez toutes les chaînes de caractères affichée sur le template pour éviter les attaques XSS (Injection de javascript dans votre page)
- Les mots de passes des utilisateurs doivent être cryptés

PHP - PROJET NOTÉ

Bonus

Seront disponibles les bonus suivants :

- Mise en page CSS pour le site (inclure de la couleur et une fiche css ne suffit pas)
- Pouvoir envoyer des images en tweets
- Implémenter la fonctionnalité « Temps fort », permettant de voir les 10 tweets du jour précédent ayant été le plus retweeter ou le plus liké (Parmis ceux que l'utilisateur suit)
- Utilisation de composer pour l'autoloading

Un seul bonus complété est suffisant pour avoir tous les points de bonus

Rendus

Afin de me rendre votre projet, vous aurez section réservée sur l'ent.

Merci de me rendre une Archive contenant les éléments suivants :

- Une fiche de description README.txt contenant les éléments suivants :
 - Nom et prénom des participants
 - Les fonctionnalités réalisées
 - Bonus tentés
 - Url vers le repository GitHub où est hébergé votre code
 - Archive de votre projet (Si GitHub ne marche pas)
 - Url de votre site en ligne

Le projet est à rendre pour le mardi 18/06 23/59 maximum.

Tout rendu en retard sera pénalisé.

Attention à bien implémenter toutes les fonctionnalités, où vous n'aurez pas tous les points.

Bon courage à vous