

Projet développement web

Projet de fin de module : 2ème session

- Thème: Magasin en ligne
- Date de fin du projet: **mercredi 2 septembre 2020**
- Défense orale (individuelle): **aucune (cause : COVID-19)**

Procédure à suivre pour la remise du projet:

L'entièreté du code doit être dans un répertoire github (accessible par l'utilisateur FBruynbroeck)

Un dump complet de la base de données mysql doit se trouver dans le répertoire github.

Analyses, diagrammes, documentations du projet seront également présents dans le répertoire github.

Un fichier README à la racine du projet contiendra la procédure à suivre afin d'installer le projet.

Exemple de README:

Le projet fonctionne avec la version 5.6 de php.
Editer les variables \$db_url, \$db_login, \$db_pass dans le fichier config.php.
\$db_url: chemin vers votre db MySQL
\$db_login: utilisateur ayant accès en lecture/écriture à la db.
\$db_pass: mot de passe de l'utilisateur utilisé pour la connexion à la db (\$db_login).
Créer une base de données MySQL ayant pour nom "projet" et procéder au restore du dump du fichier projet.sql accessible à la racine du répertoire.

Envoyer l'url de votre projet + le thème (exemple: Magasin de jeux vidéo) à l'adresse francois.bruynbroeck@hotmail.com avant le **2 septembre** (=dès que possible)

Attention, le sujet du mail doit commencer par: [PROJET IPAM] + Nom + Prénom.

Exemple: [PROJET IPAM] Bruynbroeck François

La récupération de votre projet depuis github se fera le **2 septembre à minuit**. Il ne sera donc plus possible de modifier votre projet après cette date.

Pensez à bien push votre code (régulièrement) et vérifiez que votre répertoire github possède bien l'entièreté du projet (accédez à votre répertoire depuis l'interface web github si vous avez un doute).

Limitations

Le site web permettra aux visiteurs de:

- Consulter la liste des articles
- Voir le détail des articles depuis une url dédiée (exemple : [monsite.be/monarticle](#))
- De s'inscrire sur le site
- De s'authentifier

Le site web permettra (en plus) aux clients (personnes authentifiées) de:

- Ajouter des articles dans son panier
- Valider une commande (confirmer son panier)
- Consulter la liste de ses commandes
- Modifier ses données personnelles (exemple : nom/prénom, adresse)

Le site web permettra aux administrateurs de:

- Ajouter des articles
- Editer des articles
- Supprimer des articles
- Lister les commandes des clients
- Statistique des commandes
- Ajouter d'autres administrateurs

SEO: La structure des URLs doit être soignée.

Les URLs comporteront toujours un mot ou une phrase identifiant la nature de la page.

(pour une phrase, les mots seront séparés par des tirets).

Exemple: l'utilisateur souhaitant accéder au détail d'une chaise de jardin accédera à l'url:
monprojet.be/articles-jardin/chaise

Libre à vous de structurer les URLs comme vous le souhaitez.

Il est également demandé d'utiliser à un moment le langage javascript dans votre projet.

Quelques idées d'utilisation:

- Un formulaire pour rajouter des articles via un fichier json avec une prévisualisation (preview via javascript)
- Drag and drop des articles dans le panier
- Prévisualisation du rendu final d'un article en temps réel quand on complète un formulaire de création
- Message d'avertissement quand on encode un article déjà créé (avant même de confirmer le formulaire)
- ...

Ce sont des exemples ! Vous ne devez pas implémenter l'entièreté de la liste !

Concernant le détail des articles, le minimum à afficher est:

- Un titre
- Un prix
- Une description
- Une image représentant l'article

La gestion d'un stock n'est donc pas obligatoire (on suppose que le stock est illimité).

Concernant les commandes, le minimum à afficher est:

- Le nom du client qui passe la commande
- Les articles commandés
- Le prix par article
- Le prix total de la commande
- Le statut de la commande (exemple : en attente, validée, annulée)

Concernant les statistiques des commandes, l'administrateur doit (en un clin d'oeil) pouvoir visualiser les articles les plus vendus grâce à un graphique.

Exemple : top 5 des articles sous forme d'un diagramme circulaire (« camembert »).

L'utilisation d'une librairie javascript est grandement conseillé afin de ne pas devoir refaire la roue.

Conseil : utilisez ajax pour rechercher les données de manière asynchrone.

Le site sera développé en PHP (minimum v5.6) ET JavaScript sans aucun framework.

Les données sont stockées dans une base de données MySQL.

Aucune limitation au niveau des librairies que vous souhaitez utiliser (Ex: jQuery).

Attention cependant à utiliser des librairies récentes (vous êtes pro, et donc, vous devez utiliser des outils à jour).

Côté skin (CSS), le site doit être agréable à visiter.

Vous devez donc structurer le rendu et respecter les balises (<h1..2..3>, <div>, <label>, <table>, <th>, etc)

Pensez également à la résolution des écrans pouvant varier.

Conseil: utilisez des librairies comme Bootstrap (<https://getbootstrap.com>) afin de limiter les risques.

Structure

Le développeur doit respecter l'architecture MVC.

Avec le framework codeigniter3, cette structure est déjà implémentée.

Sans framework, il est important de séparer vos fichiers en (minimum) 3 dossiers:

- models
- views
- controllers

Un fichier "index.php" à la racine du projet fera office de routeur.

Le développeur aura le choix de représenter les modèles sous forme d'objet PHP qui fera alors office de mapper (ORM) OU de travailler avec des fonctions.

N'oubliez pas de sécuriser vos dossiers et de configurer apache (via .htaccess) afin de toujours faire pointer les chemins vers votre routeur.

Attention: il est possible que votre client git n'ajoute pas les fichiers cachés (exemple: .htaccess).

Vérifiez donc cela (configurez votre client pour qu'il le fasse ou ajoutez les fichiers depuis l'interface web de github)

Par où commencer ?

Avant de foncer dans le code, vous devez reprendre les différentes demandes du client sous forme de point.

Ensuite, préparez votre analyse.

Pour faciliter votre développement, vous pouvez représenter les différents cas d'utilisation (via un diagramme des cas d'utilisation).

Exemple:

En tant que visiteur, je peux:

- Voir la liste des jeux vidéo
- Voir le détail d'un jeu vidéo
- M'identifier
- Créer un nouveau compte utilisateur

En tant qu'administrateur, je peux:

- Ajouter un jeu vidéo
- Importer des jeux vidéo
- Editer un jeu vidéo
- Supprimer un jeu vidéo
- Editer mon profil
- Voir le listing des commandes
- Ajouter un nouvel administrateur
- Voir les statistiques
- Me déconnecter

En tant que simple utilisateur connecté, je peux:

- Ajouter des articles dans un panier
- Voir mes commandes
- Editer mon profil
- Me déconnecter

Grâce à ce genre de diagramme, vous avez une vision plus claire du projet. Il sera alors plus facile de développer sans oublier quelque chose.

Ensuite concevez votre base de données en fonction de votre projet.

Là aussi, une analyse est nécessaire afin de ne pas oublier une table, une colonne, une relation.

Préparez un diagramme (via UML par exemple) afin d'avoir un visuel de votre base de données.

Il sera alors plus facile de créer votre base de données sur base de votre diagramme.

Une fois que la base de données est prête, commencez par coder les différentes pages qui seront utilisées (aucune logique, juste un titre).

Attention: Ne pas oublier de bien respecter la structure MVC !

Ensuite, implémentez votre page de listing qui est l'étape la plus simple.

Sans le savoir, en procédant de cette façon, vous travaillez avec une méthode "agile".

Le principe ici est de développer par étape en ayant un rendu visuel "rapide".

De cette façon, vous pouvez revenir vers votre client en lui montrant l'état d'avancement du projet.

Quelques avantages:

Vous voyez un avancement rapide.

Le client voit que son projet avance.

Le client peut revenir sur des détails que l'on peut alors adapter/corriger rapidement (ce qui serait plus difficile une fois le projet terminé).

Continuez l'implémentation des autres pages.

N'oubliez pas que des outils/plugins/librairies existent ! Il ne faut pas refaire la roue.

Extra

Les gens ne savent pas ce qu'ils veulent avant que vous le leur montriez - Steve Jobs

C'est la même chose pour votre client.

Soyez imaginatif. Pensez logique et facilité.

Pourquoi ne pas lui proposer un formulaire d'inscription à une newsletter pour ses visiteurs ? De cette façon, on pourrait envoyer de la publicité.

Mais aussi, un formulaire pour éditer le profil, rajouter un avatar aux utilisateurs, permettre l'ajout d'images pour les articles, rajouter une gestion du stock, etc.

Aucune obligation pour cette partie. Il s'agit d'Extra.

Mais gardez en tête qu'un client satisfait est un client fidèle.

Si vous allez plus loin que sa demande, il vous sera reconnaissant.