

E-SHOP

Type :	PROJET
Formations :	Ynov Informatique
Promotions :	Bachelor 2
UF :	Technologies WEB & Base de données

1. CADRE DU PROJET

Ce projet permet l'évaluation des compétences acquises grâce aux modules des UF « Technologies Web » et « Base de données ». Pour ce faire, ce projet devra être réalisé en groupe de 2.

Vous avez le choix parmi une liste de projets proposés dans la partie « LISTE DES PROJETS AU CHOIX ».

Les ressources qui vous sont proposées pour réaliser ce projet sont sur le Framework Laravel et la base de données MySQL. Si vous prenez le choix d'utiliser d'autres technologies pour réaliser votre projet, il vous appartient d'effectuer les recherches par vous-même.

2. OBJECTIFS DE FORMATION VISÉS

Vous serez évalué sur les compétences suivantes : *UF Technologie web et base de données.*

FRAMEWORK WEB

- maîtriser les outils du Framework
- mettre en place la persistance de données via un ORM
- organiser son projet en respectant l'arborescence des fichiers du Framework
- traiter des formulaires web (CRUD)
- gérer le système d'authentification

CRÉER UN SITE WEB ERGONOMIQUE

- structurer visuellement le contenu d'un site avec une démarche centrée utilisateur

CRÉER UN SITE WEB RESPONSIVE

- mettre en œuvre les techniques du web adaptatif

POO (Programmation Orientée Objet)

- programmer en orienté objet pour manipuler les données du site

CRÉER DES ANIMATIONS SUR LE SITE

- animer l'aspect graphique d'une page web en manipulant son contenu

BASE DE DONNÉES

- créer une base de données et des tables pour stocker les données du site
- créer des requêtes SQL pour manipuler les données de votre base de données

ALGORITHME

- mettre en place des fonctionnalités mettant en avant vos compétences en algorithmie

SERVEUR WEB

- mettre en place un environnement de développement
- mettre en place un environnement de production

VERSIONNER SON CODE

- utiliser un outil de versionning de code

3. PREREQUIS

- FRAMEWORK WEB
- Base de données (MySQL, MongoDB...)
- POO (Programmation Orientée Objet)
- Serveur WEB (Apache, Nginx)
- HTML/CSS
- JavaScript
- Langage web (PHP, NodeJS, Python, Ruby...)
- GIT
- UML
- Gestionnaire de dépendances (composer, npm, pip...)
- ORM

4. LIVRABLES

- dépôt GIT de votre site à jour
- votre site web en ligne (Serveurs d'Ynov)
- un document de présentation de votre projet (architecture du site, technologies utilisées, structure de la base de données, fonctionnalités majeures, captures d'écran...)

5. MODALITÉS D'ÉVALUATION DU PROJET

Vous serez évalué sur l'ensemble des productions. L'évaluation prendra aussi la forme d'une présentation orale de synthèse d'environ 10 minutes accompagnée d'un support de présentation et d'une démonstration des fonctionnalités du site mises en place.

Des évaluations intermédiaires auront également lieu au cours du déroulement du Projet.

RÉCAPITULATIF DE LA GRILLE DE NOTATION

Selon le nombre de points de difficulté total mis en place sur votre projet, des points bonus seront accordés selon les paliers suivants :

<i>Points de difficulté : Entre 30 et 35</i>	<i>0 point bonus</i>
<i>Points de difficulté : Entre 35 et 40</i>	<i>1 point bonus</i>
<i>Points de difficulté : Plus de 40</i>	<i>2 points bonus</i>

Les projets proposés devront être réalisés dans leurs intégralité pour obtenir la totalité des points.

6. DESCRIPTIF DU PROJET

Vous êtes totalement libre quant à l'apparence de vos interfaces. L'utilisation de librairies (bootstrap, tailwind, materialize...) est autorisée et encouragée afin de gagner en rapidité de développement.

BONUS POUR TOUS LES PROJETS :

Vous pouvez implémenter une architecture de type API REST. Celle-ci constitue un bonus et n'est donc pas obligatoire.

LISTE DES PROJETS AU CHOIX :

1^{er} PROJET : ECOMMERCE

Présentation

Il s'agit d'un site e-Commerce de vente de jeux vidéo. Les utilisateurs peuvent acheter des jeux vidéo. Lorsqu'un utilisateur en achète un, il recevra un email avec le code pour activer son jeu sur sa plateforme (PC, Xbox, PlayStation, ...) ainsi qu'une facture en PDF. Cette facture se retrouvera également dans son espace membre sur le site. Un membre ne peut acheter un jeu que s'il dispose d'un solde suffisant. Également, les jeux ont une quantité définie, il faudra vérifier le stock avant d'autoriser un achat.

Un espace administration permet de gérer les jeux en vente et les informations des membres.

Pages

Le site se composera obligatoirement des pages suivantes :

- page d'accueil (catalogue des jeux vidéo)
- page produit (description et avis)
- page du panier d'achat
- page d'espace membre
- page d'inscription / connexion

L'administration se composera obligatoirement des pages suivantes :

- page tableau de bord
- page des membres
- page des jeux
- page de connexion

Fonctionnalités

Partie membres

- la page d'accueil :
 - liste de tous les jeux disponibles avec un système de pagination
difficulté : 3
 - un système de recherche par le nom du jeu *difficulté : 2*
- les utilisateurs peuvent se créer un compte et se connecter. Ils pourront compléter et modifier leur profil *difficulté : 5*
 - chaque profil utilisateur possède :
 - un prénom
 - un nom
 - une adresse email
 - une date de naissance
 - un solde (factice)
- la page produit contient : *difficulté : 7*
 - l'image du jeu
 - le nom du jeu
 - le prix du jeu
 - la note moyenne
 - les avis des acheteurs
 - un bouton d'achat
 - un formulaire pour ajouter/modifier son avis (si le membre a déjà acheté le jeu)
- un email automatique est envoyé à l'acheteur après la vente avec : *difficulté : 5*
 - le code d'activation
 - la facture en PDF

Partie administrateur

- un espace administrateur est mis en place avec une authentification
difficulté : 3
- un tableau de bord est présent dans l'espace administrateur avec des informations
tels que : *difficulté : 2*
 - nombre de membres
 - nombre de ventes
 - nombre de nouvelles ventes sur les 7 derniers jours
 - les revenus du site totaux
 - les revenus du site sur les 7 derniers jours

- les administrateurs peuvent gérer les jeux avec un CRUD : *difficulté : 5*
 - le CRUD permet d'ajouter/modifier/supprimer un jeu avec :
 - un nom
 - une description
 - une photo
 - une quantité
 - un prix
 - un code d'activation
 - les administrateurs peuvent consulter :
 - la note moyenne des notes
 - les avis laissés par les acheteurs
- les administrateurs peuvent gérer les membres avec un CRUD : *difficulté : 4*
 - le CRUD permet de modifier :
 - un prénom
 - un nom
 - une adresse email
 - une date de naissance
 - les administrateurs peuvent consulter :
 - les achats de jeux
 - les factures en PDF

Degré de difficulté total : 36 points

2^{ème} PROJET : LIVRAISON DE REPAS

Présentation

Il s'agit d'un site de livraison de repas. Les membres connectés peuvent commander un plat chez un restaurant. Il devra ensuite valider la réception de celui-ci. Un client peut acheter un plat s'il dispose d'un solde suffisant.

Un espace restaurateur permettra aux restaurant d'ajouter, de modifier ou de supprimer leurs plats.

Un espace administrateur sera mis en place pour gérer le service après-vente et les restaurateurs.

Pages

Le site se composera obligatoirement des pages suivantes :

- page d'inscription / connexion
- page d'accueil (liste des restaurants)
- page restaurant (affiche les plats du restaurant)

L'espace restaurateur se composera obligatoirement des pages suivantes :

- page d'inscription / connexion
- page tableau de bord (décrit ci-dessous)
- page plats (CRUD)
- page de profil (décrit ci-dessous)

L'administration se composera obligatoirement des pages suivantes :

- page de connexion
- page tableau de bord (décrit ci-dessous)
- page des membres (CRUD)
- page des restaurants (CRUD)
- page plats (CRUD)

Fonctionnalités

Partie membres

- les utilisateurs peuvent se créer un compte et se connecter. Ils pourront compléter et modifier leur profil *difficulté : 5*
 - chaque membre possède :
 - un prénom
 - un nom
 - une adresse email
 - une adresse postal
 - un solde (factice)
- les frais de livraison seront de 2,5€ et seront facturés en plus du prix du plat choisit
- une commande peut comporter X plats d'un même restaurant (**une commande ne peut pas avoir des plats qui proviennent de plusieurs restaurants**) : *difficulté : 3*
- un email automatique est envoyé au restaurant après la vente avec : *difficulté : 3*
 - le numéro de commande
 - le plat commandé
 - l'adresse du client
 - l'heure de livraison estimée (1h de plus que la date à laquelle la commande a été passé)
- les clients peuvent noter un plat après l'avoir bien reçu *difficulté : 3*

Partie restaurateurs

- les restaurateurs peuvent se créer un compte et se connecter. Ils pourront compléter et modifier leur profil *difficulté : 3*
 - chaque restaurant possède :
 - un nom
 - un logo
 - une adresse
 - une adresse email
- les restaurateurs peuvent ajouter/modifier/supprimer des plats : *difficulté : 4*
 - chaque plat possède :
 - une photo
 - un nom
 - un prix (à l'unité)

Partie administrateur

- un espace administrateur est mis en place avec une authentification
difficulté : 3
- un tableau de bord est présent dans l'espace administrateur avec des informations
telles que : *difficulté : 3*
 - nombre de restaurants
 - nombre de commandes passées
 - nombre de commandes en cours
 - les revenus total (la somme des 2,5€ de commission pour le site pour les commandes passées)
- les administrateurs peuvent gérer les restaurants avec un CRUD : *difficulté : 5*
 - le CRUD permet de modifier/supprimer un restaurant avec :
 - un nom
 - un logo
 - une adresse
 - une adresse email
 - les administrateurs peuvent consulter :
 - les commandes passées auprès du restaurant
 - les commandes en cours
 - les avis laissés par les clients
- les administrateurs peuvent gérer les clients avec un CRUD : *difficulté : 5*
 - le CRUD permet de modifier/supprimer :
 - un prénom
 - un nom
 - une adresse email
 - une adresse postal
 - un solde (factice)
 - les administrateurs peuvent consulter :
 - les commandes passées et en cours du client

Degré de difficulté total : 37 points

7. RESSOURCES COMPLEMENTAIRES

Environnement de dev : <https://laravel.com/docs/5.8/>

homestead GIT : <https://www.grafikart.fr/formations/git>

Composer : <https://getcomposer.org>

Documentation Laravel 5.8 : <https://laravel.com>

Commande Laravel : <https://laravel.com/docs/5.8/artisan#writing-commands>

ORM : <https://laravel.com/docs/5.8/eloquent>

MAILING : <https://laravel.com/docs/5.8/mail>

Integrer du CSS & JS : <https://laravel.com/docs/5.8/mix#working-with-scripts>

REACT : <https://reactjs.org>

VueJS : <https://vuejs.org>

SELECT2 JS: <https://select2.org>

Date Picker: <https://bootstrap-datepicker.readthedocs.io/en/latest/>

Outil de debug laravel : <https://laravel.com/docs/5.8/telescope>

Tuto Laravel : <https://morioh.com/p/9b8c8ef67bd5/laravel-5-8-tutorial-from-scratch-for-beginners>

8. BESOINS MATERIELS ET LOGICIELS

- un IDE (environnement de développement)
- un serveur web local (WAMP, MAMP, LAMP...)
- un serveur de production (Serveur Ynov)
- un langage de développement web
- GIT