

Mission 1 : Agora Web (Architecture MVC)

Valentin, Yenis, Marco, Marvin

Projet réalisé : Développement du site d'administration de la MJC AGORA de Libreville

Dans le cadre de notre collaboration avec la MJC AGORA de Libreville, une association d'éducation populaire engagée dans la promotion de la culture, de l'éducation et du vivre-ensemble, nous avons été sollicités pour concevoir et développer leur site d'administration des données.

La MJC AGORA est un lieu dynamique où se croisent les expressions artistiques, les pratiques numériques, les activités sportives, culturelles et intergénérationnelles. Elle est affiliée à plusieurs fédérations nationales (FFMJC, FRMJC, FFE, FFRS) et dispose des agréments nécessaires pour mener des projets à l'échelle nationale et européenne (Service Volontaire Européen, Jeunesse et Sport).

Nous devons développer une application web pour permettre à la MJC de gérer plus efficacement ses données internes, plus précisément pour la gestion des jeux, marques, pegi et plateformes.

Organisation projet

Nous nous sommes partagés ainsi les tâches :

Structure (dossiers et index.php): **Marco, Valentin**

CRUD Jeux : **Valentin, Marvin**

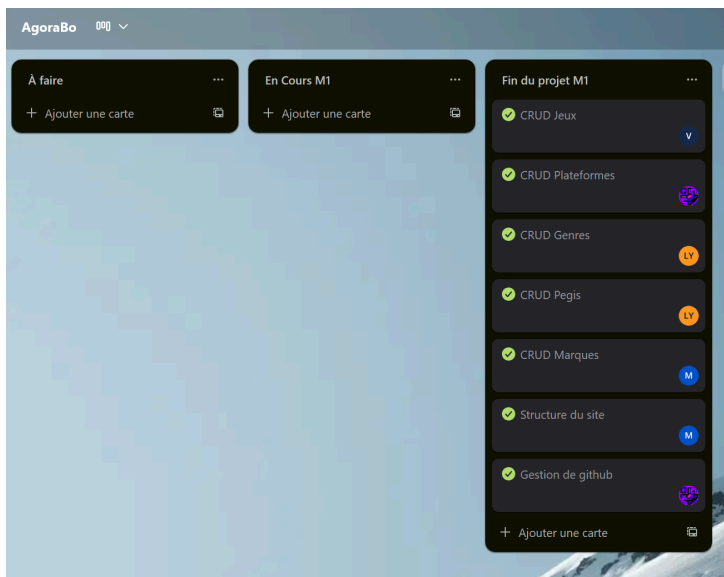
CRUD Marques : **Marco**

CRUD Pegis : **Yenis**

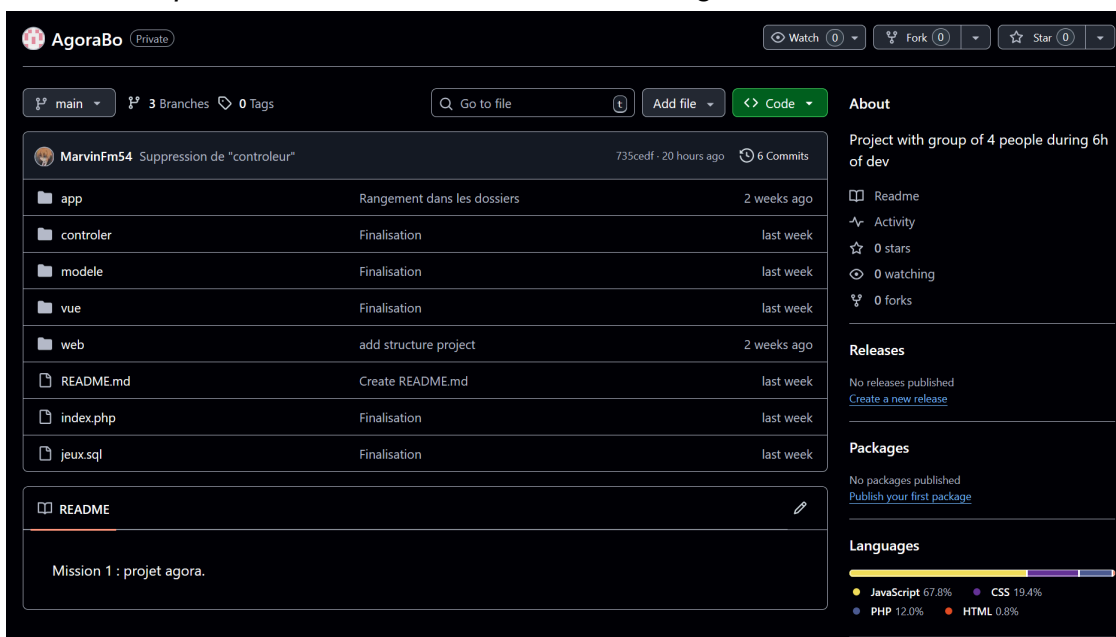
CRUD Genres : **Yenis**

CRUD Plateformes : **Marvin**

Voici une capture d'écran du Trello utilisé (une fois le projet terminé)



Voici une capture d'écran de la branche main du github



Technologies utilisées

Langage back-end : **PHP**

Langage front-end: **HTML, Bootstrap (CSS), JS**

Base de données : **MySQL**

Architecture : **Model - View - Controller (MVC)**

Outils : **Visual Studio Code, GitHub, Apache, Xampp, Wamp**

Fonctions principales développées

CRUD

- **Create**
- **Read**
- **Update**
- **Delete**

Chaque élément (jeux, marques, pegi, plateforme) a ses propres actions CRUD, pour permettre une gestion totale du projet.

# Ref Jeu	Plateforme	PEGI	Genre	Marque	Nom	Prix	Date de Parution	Actions
8F6763098765	Playstation 3	12 Ans minimum	Sport	Bandai Namco Entertainment	FIFA 18 - Edition essentielles	59.99 €	2017-09-29	[Edit] [Delete]
US276454756T	Playstation 3	12 Ans minimum	Sport	Sony Interactive Ent.	Gran Turismo Sport	21.50 €	2013-06-12	[Edit] [Delete]
ET868874315	Xbox 360	18 Ans minimum	Action	Warner Bros	La terre de milieu - L'Ombre de la Guerre	59.90 €	2017-10-10	[Edit] [Delete]
Y1E548788	Nintendo 3DS	3 Ans minimum	Aventure	Nintendo	Mario Kart 7	39.90 €	2012-11-28	[Edit] [Delete]
8827535987	Nintendo 3DS	12 Ans minimum	Aventure	Sony Interactive Ent.	Minecraft Story Mode - L'aventure Complète	39.99 €	2016-12-16	[Edit] [Delete]
TF98653108	Nintendo Wii U	12 Ans minimum	Aventure	Sony Interactive Ent.	Minecraft Story Mode - L'aventure Complète	39.99 €	2016-12-16	[Edit] [Delete]
RT495867312	Nintendo Wii	7 Ans minimum	Aventure	Nintendo	New Super Mario Bros	19.90 €	2016-04-16	[Edit] [Delete]
CF475483776	Nintendo 3DS	3 Ans minimum	Stratégie	Tecliland	Paddington - escapades à Londres	18.30 €	2015-06-19	[Edit] [Delete]
EG7635475989F	Nintendo 3DS	7 Ans minimum	Jeu de rôle	Nintendo	Pokemon X	39.90 €	2013-10-12	[Edit] [Delete]
ER49324478	Xbox One	18 Ans minimum	Action	Square Enix	Rise of the Tomb Raider	19.90 €	2015-11-13	[Edit] [Delete]
1234567890123456	Oculus quest	12 Ans minimum	Jeux réalité virtuelle	Bandai Namco Entertainment	Super hot	25.00 €	2016-02-25	[Edit] [Delete]
ES47562098754	Nintendo Wii	7 Ans minimum	Aventure	Nintendo	The Legend of Zelda - The Wind Waker HD	29.80 €	2016-04-15	[Edit] [Delete]

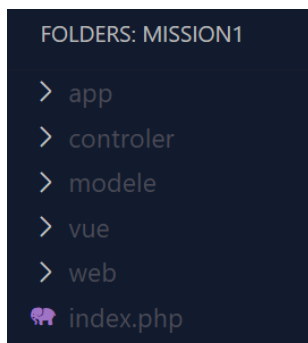
Capture d'écran de la page de gestion des jeux, on voit bien les fonctionnalités CRUD.

Pourquoi utiliser une structure en MVC ?

Le modèle MVC repose sur une séparation claire des responsabilités :

- **Model** : gère la logique métier et les interactions avec la base de données (requêtes SQL, récupération, insertion, suppression, mise à jour).
- **View** : s'occupe uniquement de l'affichage (HTML, CSS, JS) et de la restitution des données aux utilisateurs.
- **Controller** : agit comme intermédiaire entre Model et View. Il reçoit les actions de l'utilisateur (clics, formulaires envoyés), appelle les méthodes du Model nécessaires, puis transmet les résultats à la View pour affichage.

Cette structure facilite la maintenance, permet de travailler en parallèle (un développeur sur la BDD, un autre sur l'UI), et rend l'application plus claire et évolutive.

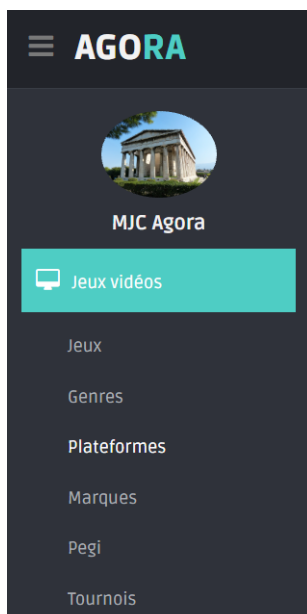


Capture d'écran du dossier contenant le projet, on voit ici l'architecture en MVC.

Cas d'utilisation

Afficher les plateformes :

L'utilisateur (admin) sélectionne « **Plateformes** » dans le menu.



Capture d'écran du menu permettant l'accès aux différentes pages.

Le fichier **index.php** détermine le bon contrôleur à appeler, en fonction de la variable `$_GET` (qui indique la page et l'action).

```
case 'gererPlateformes': {
    $menuActif = 'Plateformes';
    require 'vue/v_menu.php';
    require 'contrôleur/c_gererPlateformes.php';
    break;
}
```

Capture d'écran de l'appelle du contrôleur des plateformes.

Le **Contrôleur** (`c_gererPlateformes.php`) reçoit la requête et appelle la méthode (`getLesPlateformes()`) correspondante du **Model** (`class.PdoJeux.inc.php`).

Le **Modèle** exécute une requête SQL de type `SELECT * FROM plateforme`.

Le **Modèle** renvoie au **Controller** la liste des plateformes sous forme de tableau associatif.

Le **Controller** transmet ces données à la **View** (`v_lesPlateformes.php`).

```
$tbPlateformes = $db->getLesPlateformes();
require 'vue/v_lesPlateformes.php';
```

Capture d'écran de la création tableau des plateformes et de l'appelle de la vue.

La **View** affiche la liste des plateformes dans un tableau HTML (nom, éditeur, date, etc.). *Captures d'écran du code et de l'affichage de la vue.*

```
<?php
foreach ($tbPlateformes as $plateforme) {
    ?>
    <tr>
        <form action="index.php?uc=gererPlatef
        <td><?php echo $plateforme->identifiant
        <td><?php
            if ($plateforme->identifiant != $id
                echo $plateforme->libelle;
                ?>
            </td><td>
                <?php if ($notification !=
                    echo '...<button class="bt
                } ?>
                <button class="btn btn-prim
                <button class="btn btn-dang
            </td>
        </td>
    <?php
    }
    else {
        ?><input type="text" id="txtLib
    </td>
    <td>
```

Identifiant	Libellé
Nouveau	Libellé
9	Nintendo 2DS
3	Nintendo 3DS
11	Nintendo DS
13	Nintendo Switch
4	Nintendo wii
15	Nintendo wii U
18	Oculus quest
5	PC
2	PlayStation 3
1	PlayStation 4
17	PlayStation Vita
6	Sony PSP
7	Xbox 360
8	Xbox One