Foto’Gal

Rapport TPI et documentation technique

**Travail personnel individuel (TPI)**

Elias Zaiem — Mai 2022

Maître d’apprentissage : Stéphane Garchery

Table des matières

[Introduction 2](#_Toc103160998)

[Rappel d’énoncé 2](#_Toc103160999)

[**Organisation** 2](#_Toc103161000)

[**Livrables** 2](#_Toc103161001)

[**Matériel et logiciels à disposition** 3](#_Toc103161002)

[**Outils et technologies utilisées** 3](#_Toc103161003)

[**1.** **Html** 3](#_Toc103161004)

[**2.** **Css** 3](#_Toc103161005)

[**3.** **Php** 3](#_Toc103161006)

[**4.** **SQL** 3](#_Toc103161007)

[**Description de l’application** 3](#_Toc103161008)

[Méthodologie 4](#_Toc103161009)

[**Méthode en 6 étapes** 4](#_Toc103161010)

[Planification 5](#_Toc103161011)

[**Diagramme de Gantt (prévisionnel)** 5](#_Toc103161012)

[**Diagramme de Gantt (effectif)** 6](#_Toc103161013)

[Maquettage 7](#_Toc103161014)

[Généralités concernant l’implémentation 15](#_Toc103161015)

[**Base de données** 15](#_Toc103161016)

[**Classes (PHP)** 15](#_Toc103161017)

[**Contrôleurs (PHP)** 15](#_Toc103161018)

[Librairie et outils externes 16](#_Toc103161019)

[Scénarios de tests 16](#_Toc103161020)

[Conclusion 16](#_Toc103161021)

## Introduction

Ce document est un rapport présentant différents aspects de la conception du projet Foto’Gal. Ce projet a été réalisé dans le cadre du Travail pratique individuel (TPI) durant la session de mai 2022. Il a pour but de valider mes compétences acquises pendant la formation « Informaticien CFC » dispensée par l’école informatique du Centre de Formation Professionnelle et Technique (CFPT) au Petit-Lancy.

**Foto’Gal** est un site web qui permet à l’utilisateur de *voir*, *poster* et *trier* des productions disponibles sur le site, il peut modifier les productions que lui-même à poster. Une fonctionnalité de Like/Unlike sera disponible. L’utilisateur peut modifier son compte à tout moment, il peut aussi changer son mot de passe. Un système de récupération de mot de passe est aussi disponible. Un *administrateur* est présent sur le site, celui-ci peut *ajouter*/*modifier* ou *supprimer* une catégorie, il peut aussi gérer les droits utilisateurs (rendre un utilisateur admin ou non). Une page d’aide permet aux utilisateurs de s’y retrouver si jamais ils n’arrivent pas à faire quelque chose.

## Rappel d’énoncé

### **Organisation**

|  |
| --- |
| Élève |
| Elias Zaiem  <Elias.zm@eduge.ch> |

|  |
| --- |
| Maître d’apprentissage |
| Stephane Garchery  <edu-garcherys@eduge.ch> |

|  |
| --- |
| Expert 1 |
| Nicolas Terrond  <[nicolas.terrond@sig-ge.ch](mailto:nicolas.terrond@sig-ge.ch)> |

|  |
| --- |
| Expert 2 |
| Mickaël Strazzeri  <mickael.strazzeri@git-it.ch> |

### **Livrables**

|  |  |
| --- | --- |
| Pour les experts et le formateur par e-mail : | Pour le formateur uniquement : |
| * Planning détaillé du projet * Rapport du projet contenant le code source du projet au format PDF * Journal de bord * Résumé du TPI (1 page A4) | * Accès au repository distant du projet avec les droits de « colonage » * Un README expliquant l’installation du projet sur un poste de travail local * Un dump de la base de données contenant un set de données |

### **Matériel et logiciels à disposition**

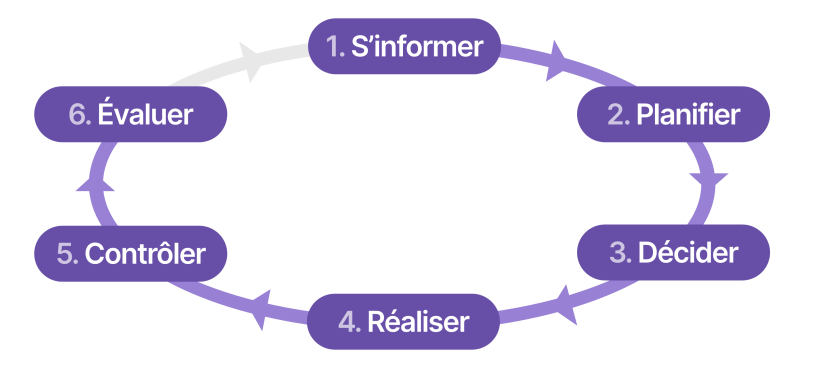
* Un PC standard école avec Windows 10, 2 écrans
* Serveur Web et SGBD à choix (EasyPHP, Wamp, Laragon, Adminr, phpMyAdmin, autre)
* IDE à choix (NetBeans, Visual Studio Code, PHPStorm, autre)
* Logiciel de création de schémas (Visio, Gliffy, autre)
* Outil de versionnage de code (Git, avec dépôt distant sur Github / Bitbucket / GitLab)
* Navigateur web (Mozilla Firefox/Google Chrome)
* Logiciel de création de maquettes d’interfaces utilisateur (Sketch, Adobe XD, Proto.io, autre)
* Outil bureautique à choix pour les documents (Google Docs, MSOffice, OpenOffice)

### **Description de l’application**

L’application permet aux visiteurs de voir les productions graphiques des utilisateurs. Les visiteurs doivent s’inscrire afin de pouvoir publier et/ou voter pour des productions. Ils ne peuvent voter (like / unlike) qu’une seule fois.

## Méthodologie

### **Méthode en 6 étapes**

****Pour planifier mon projet de TPI, je me suis basé sur la méthode en 6 étapes qui nous a été enseignée au CFPT.

**S’informer :**

* La toute première étape de mon projet était la lecture en profondeur de mon énoncé pour comprendre toutes les fonctionnalités qu’il était nécessaire d’implémenter. J’ai également demandé à mon formateur des clarifications sur des détails des spécifications lorsque c’était nécessaire.

**Planifier :**

* Dès le début du projet, j’ai préparé un planning de travail pour savoir ce que j’avais besoin de faire et quand. Afin de pouvoir créer ce planning, j’ai eu besoin de découper mon travail en sous-tâches : j’ai donc divisé les divers points de l’énoncé sous forme de petites tâches. J’ai ensuite créé un diagramme de Gantt, outil qui m’a permis de visualiser au quotidien ma progression dans le travail ainsi que les différences entre ma planification et mon avancement effectif.

**Décider :**

* Au cours de l’avancement de mon travail, j’ai dû prendre de nombreuses décisions sur la manière de le réaliser. Lorsque je faisais des choix que je jugeais importants pour le projet, j’en ai parlé dans mon journal de bord en expliquant les raisons qui m’ont poussé à choisir ces options d’implémentation.

**Réaliser :**

1. Une fois les bonnes décisions prises, j’ai pu continuer sur l’implémentation de la fonctionnalité dans le code (ou la rédaction pour tout ce qui concerne la documentation).

**Contrôler :**

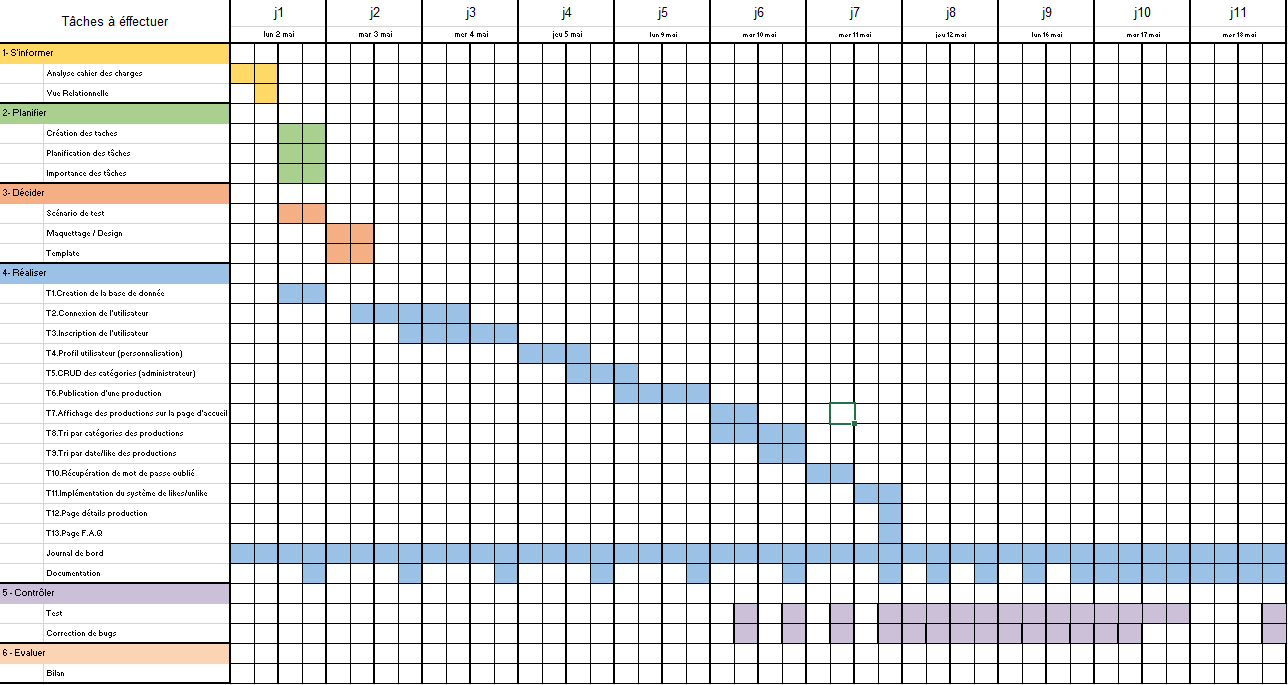
* À chaque fois que je terminais une fonctionnalité, je l’ai testée dans différents cas d’usage pour être sûr qu’elle fonctionne comme prévu. J’ai documenté mes tests dans mon rapport TPI pour m’assurer qu’ils soient reproductibles par une personne externe au besoin. J’ai aussi effectué à nouveau tous mes tests à la fin de chaque journée de travail pour éviter les régressions. J’ai ajouté à mon rapport TPI un tableau qui permet de visualiser l’avancement de l’état des tests au cours du projet. Enfin, j’ai testé de fond en comble le programme à la fin de son implémentation sur d’autres plateformes (navigateurs et systèmes d’exploitation) pour m’assurer qu’il puisse bien fonctionner chez tout le monde.

**Évaluer :**

* La dernière étape de la méthodologie en 6 étapes est l’évaluation, pour faire une rétrospective de ce que j’ai fait et avoir un regard critique sur ce qui pourrait être amélioré. Pour ce faire, j’ai écrit des bilans journaliers à chaque fin de journée de travail dans mon journal de bord. Pour avoir une auto-évaluation du travail en entier, j’ai également rédigé la partie Conclusion du rapport TPI.

## Planification

### **Diagramme de Gantt (prévisionnel)**

****Voici mon planning prévisionnel, il à été réalisé dans le but de me guider durant mon travail.

### **Diagramme de Gantt (effectif)**

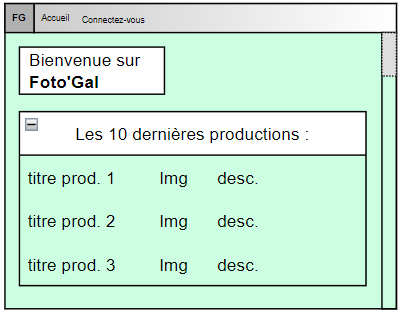
Voici maintenant mon planning effectif, celui-ci est représentatif de l’avancer de mon travail comparé à celle que j’avais prévu au début de mon travail.

**ScreenShot du plan effectif\***

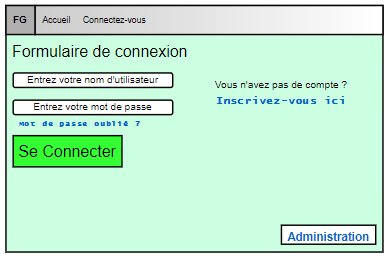
## Maquettage

**Vue d’un utilisateur non connecté :**

Cette maquette représente la page d’accueil de mon site web lorsque l’utilisateur ne se connecte pas. L’utilisateur **ne peut pas** liker/unliker les production, ni en ajouter ou modifier.

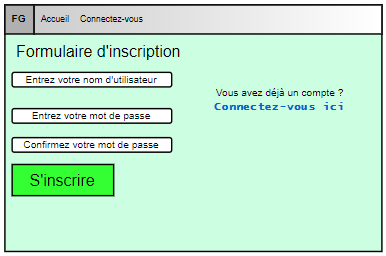


Il y a un bouton dans la barre de navigation qui propose à l’utilisateur de se **connecter**, s’il clique dessus, il sera redirigé vers la page de connexion.

****

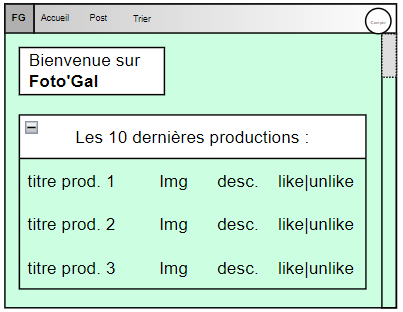
L’utilisateur peut maintenant se connecter au site web et utiliser toutes les fonctionnalités.

Si l’utilisateur n’a pas de compte, il peut en **créer** un en cliquant sur le texte \*Inscrivez-vous ici\*.



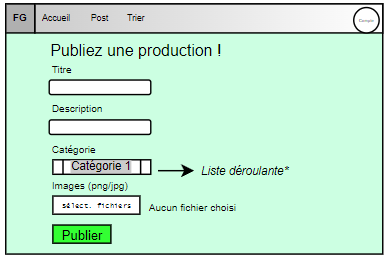
Une fois le compte créer, il peut se connecter et accéder à la page d’accueil. Il peut désormais utiliser toutes les fonctionnalités disponibles.

**Vue d’un utilisateur connecté :**

Voici la page d’accueil une fois l’utilisateur connecté au site, il peut **liker**, **trier**, **publier** et **modifier** ses productions. Il peut aussi modifier son profil en cliquant en haut à droite.

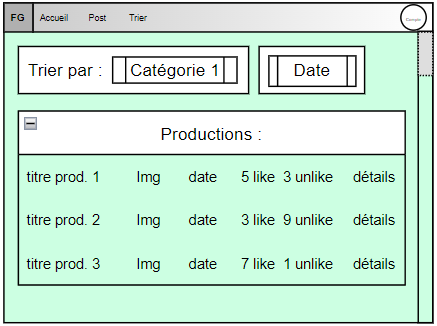
**Vue d’un utilisateur qui souhaite poster une production :**

Lorsque l’utilisateur clique sur le bouton post, il est redirigé vers cette page.

****Elle lui permet de publier une production, il doit compléter les champs ci-dessous pour pouvoir la publier. Le champ concernant la catégorie est sous forme de liste déroulante. Seules les images au format PNG et JPG ne dépassant pas 5Mo sont autorisés.

Une fois la production publiée, elle s’affichera sur la page d’accueil et sur le profil de l’utilisateur. Il peut la modifier ou la supprimer à tout moment depuis la page profil > mes productions

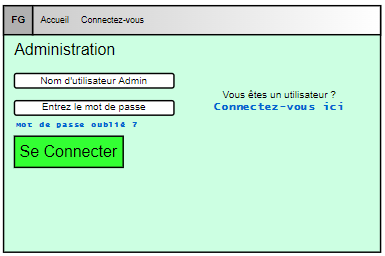
L’utilisateur peut trier les productions, il doit premièrement se rendre sur la page de tri.



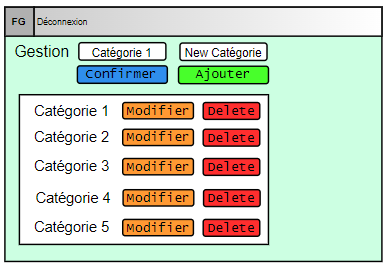
Il peut trier par catégorie, et à l’intérieur de cette catégorie il peut trier par date ou par le nombre de likes.

**Administrateur (connecté depuis le login admin)**

Voici la page de connexion pour l’administrateur.



Une fois connecté, l’administrateur arrive sur la page de gestion (CRUD) des catégories, il peut en ajouter, modifier ou supprimer.

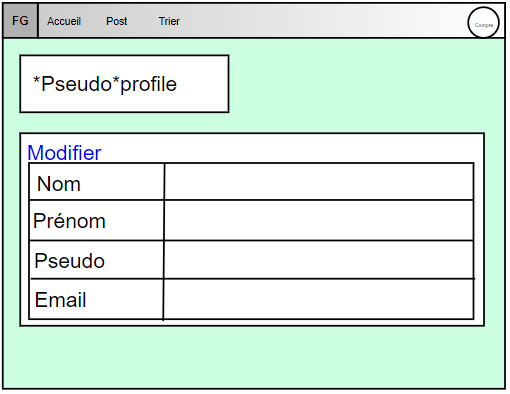


Pour ajouter une catégorie il suffit de rentrer le nom de cette nouvelle catégorie dans le champ NewCatégorie, puis de cliquer sur **Ajouter**.

Pour les modifier, il clique sur le bouton Modifier à côté de la Catégorie qu’il souhaite modifier, puis le nom de celle-ci s’affichera dans le champ Catégorie 1. Il lui reste plus qu’à modifier le nom dans le champ et cliquer sur **Confirmer**. Pour Supprimer une Catégorie, il faut juste cliquer sur **Delete** à coté de la Catégorie souhaitée.

**Vue du profil de l’utilisateur (connexion obligatoire pour y accéder)**

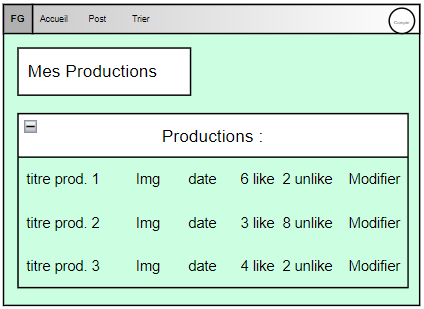
En haut à gauche de l’écran sera afficher le pseudonyme de l’utilisateur. Ses informations qui seront affichées sur la droite du tableau. Il peut cliquer sur le bouton **Modifier** pour changer ses informations personnelles.

****

Une image contenant table

Description générée automatiquement

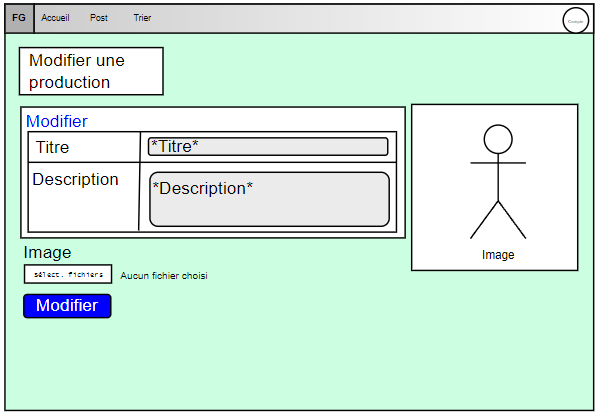
Sur votre page profil vous aurez aussi accès à vos propres productions, elles pourront être modifiées ou supprimées depuis cette page :



Sur la droite du tableau, il y aura un bouton **Modifier** et un bouton **Supprimer**.

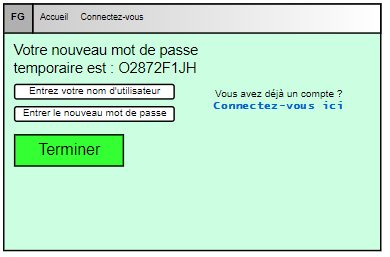
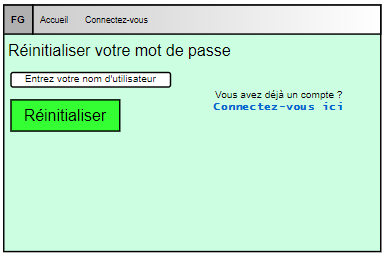
*(Le bouton supprimer n’apparaît pas sur la maquette)*

La page de modification sera similaire à cela :



Une fois les informations modifiées, il doit cliquer sur un bouton **Confirmer** (n’apparaît pas sur les maquettes). Puis il est redirigé vers la page précédente.

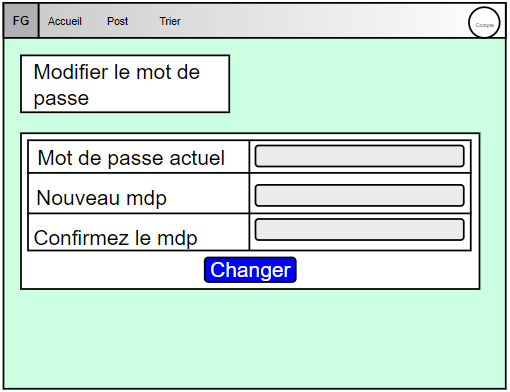
**Vue de la page mot de passe oublié :**

Si vous avez oublié votre mot de passe, ce n’est pas grave ! Une page dédiée à la récupération de votre mot de passe a été mise en place. Pour y accéder rendez-vous sur la page de connexion et cliquer sur **Mot de passe oublié.**

Entrez votre nom d’utilisateur puis cliquer sur **Réinitialiser**. Ensuite vous serez redirigé vers la page de récupération qui vous donnera un mot de passe généré aléatoirement. Copier-coller le et vous aurez à nouveau accès à votre compte. Mais un mot de passe si complexe ne vous convient pas ? Pas de soucis, rendez-vous sur la page profil > changer de mot de passe

Une fois ici, vous pourrez changer votre mot de passe. Pour le modifier, il vous suffit de mettre votre mot de passe actuel et d’ensuite entrer le nouveau puis le confirmer.

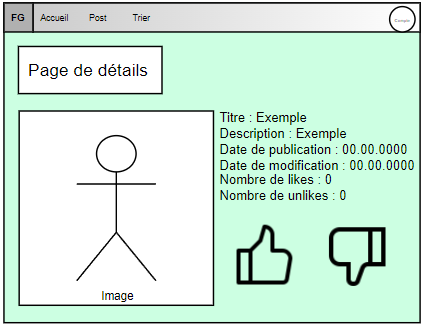
Cliquez sur **Changer** pour appliquer les modifications.



**Vue de la page détaillée d’une production :**

Vous pouvez accéder à une page détails pour une production, pour y accéder il faut tout d’abord aller sur la page de tri, puis cliquer sur **Détails.**

Une fois cela fait, vous serez redirigé sur cette page :



Vous aurez le titre, la description, la date de publication, la date de modification, le nombre de likes/unlikes et l’image en plein format. Vous pourrez liker ou unliker la production depuis cette page aussi.

## Généralités concernant l’implémentation

**Librairie et outils utilisés**

### **Html**

****

Pour ce projet, je vais utiliser l’HTML 5 pour la base visuelle du site web. Je vais le combiner avec Bootstrap pour rendre mon site responsive.

### **Css**



La Css sera utilisées pour quelques modifications visuelles. Je vais aussi utiliser les Css de Bootstrap pour qu’il fonctionne correctement.

### **Php**



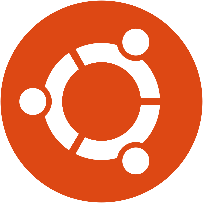
Le Php sera le cœur de mon site web, je vais l’utiliser pour l’entièreté de mes fonctionnalités.

### **SQL**



Pour tout ce qui est au niveau de la base de données et des requêtes, je vais utiliser du SQL, c’est le langage le plus répandu et c’est celui que j’ai appris à utiliser à l’école.

1. **Ubuntu 20.04**

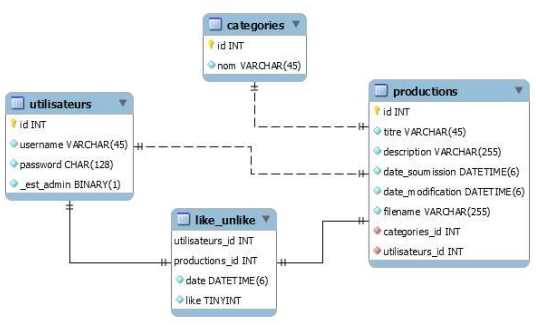
Pour avoir un serveur en local j’ai dû installer Ubuntu 20.04 Windows version, puis installer tout un tas de package nécessaires au bon fonctionnement de l’environnement linux. Ensuite j’ai installé MySQL puis connecté le tout à une base de données sur dBeaver.

### **Base de données**

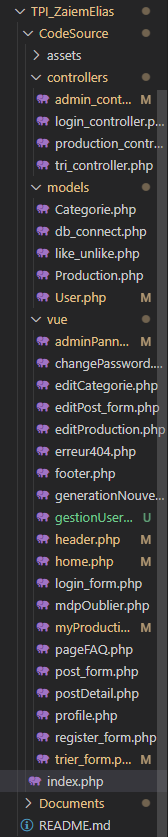
La base de données est composée de 4 tables, une pour les *utilisateurs*, une pour les *catégories*, une pour les *like/unlike* et une pour les *productions*.

Cette base de données sera réalisée sur **dBeaver**.

Chaque production sera attribuée à une catégorie dans la table *catégorie*.

Voici le modèle conceptuel de la base de données :

**Structure du projet**

Ci-contre est présentée l’arborescence de mon projet. Comme il était indiqué dans mon énoncé, je devais suivre un découpage d’application réalisé selon un des 4 « design patterns » proposés. J’ai décidé d’opter pour le « MVC » (Model-View-Controller). La totalité du code source se trouve dans le répertoire /CodeSource/. Les dossiers employés sont les suivants :

* **/controllers/** : Contient les fichiers sources des contrôleurs de mes classes PHP. Au départ j’ai voulu tout mettre dans 1 seul contrôleur, finalement par manque d’organisation j'ai changé et créer plusieurs contrôleurs pour éviter de surcharger 1 contrôleur.
* **/assets /** : Contient les dossier Css/Fonts/Js/Medias. C’est aussi l’endroit où je stock les images publiées par les utilisateurs.
* **/model/** : Contient les classes qui gère l’ajout / la modification et la suppression des catégories. Il y a aussi la classe qui s’occupe des likes et unlikes. La classe User qui elle gère tout ce qui est au niveau de l’utilisateur et pour finir, la classe « **db\_connect.php** » qui gère la connexion à la base de données.
* **/vue/**: Contient tous les fichiers qui seront utilisés pour le visuel de pages. Ces fichiers seront ensuite appelés dans des *includes***.**
* Le fichier **index.php** est le cœur du site, c’est lui qui s’occupe de connecter tout les fichiers ensembles (avec des *require* et des *includes*)

**Classes (PHP)**

Mettre les classes importantes\*

### **Contrôleurs (PHP)**

Mettre les contrôleurs\*

## Librairie et outils externes

**GitHub**

Je vous mets à disposition le lien de mon GitHub ci-dessous :

<https://github.com/ZeiShoot/TPI_ZaiemElias>

**Drive (système de sauvegarde)**

J’ai décidé de sauvegarder mes fichiers sur google drive et sur une clé USB afin d’éviter une perte de données à la suite d’un bug ou autre.

Voici le lien de celui-ci :

<https://drive.google.com/drive/folders/17FCClQ5hmOzqo8URW41YZ6C-5nsCHW8k?usp=sharing>

**Bootstrap**

Pour ce projet je vais utiliser Bootstrap 5, c’est une **Framework responsive** très polyvalent.

Voici le lien si vous désirez utiliser ce **Framework** :

<https://getbootstrap.com/docs/5.0/getting-started/download/>

## Scénarios de tests

## Conclusion

Faire une conclusion à propos du tpi et des problèmes que j’ai rencontrés durant celui-ci\*

! CONVERTIR LE CODE SOURCE EN PDF PUIS LE METTRE ICI !