Documentation Technique

Direct Prod

07 mai 2019

CFPT Informatique – Travail Pratique Individuelle

Romain Peretti – IDA-P4A

Maître d’apprentissage : Jossi Sébastien

Experts : Fontaine Philippe & Aegerter Fredy

Table des matières

[1. Introduction 3](#_Toc8917366)

[1.1 Généralités 3](#_Toc8917367)

[1.2 Pourquoi ce projet 3](#_Toc8917368)

[2. Cahier des charges 3](#_Toc8917369)

[2.1 Objectif 3](#_Toc8917370)

[2.2 Tâches & Fonctionnalités 3](#_Toc8917371)

[2.3 Matériels & logiciels 3](#_Toc8917372)

[3. Analyse fonctionnelle 4](#_Toc8917373)

[3.1 Barre de navigation 4](#_Toc8917374)

[3.1.1 Barre de navigation non connecté 4](#_Toc8917375)

[3.1.2 Barre de navigation connecté 4](#_Toc8917376)

[3.1.3 Barre de navigation de l’administrateur 4](#_Toc8917377)

[3.2 Créer un compte 5](#_Toc8917378)

[3.3 Connexion 6](#_Toc8917379)

[3.4 L’accueil 6](#_Toc8917380)

[3.4.1 L’accueil non connecté 6](#_Toc8917381)

[3.4.2 L’accueil connecté 7](#_Toc8917382)

[3.5 Détails d’une annonce 8](#_Toc8917383)

[3.5.1 Détails d’une annonce non connecté 8](#_Toc8917384)

[3.5.2 Détails d’une annonce connecté 9](#_Toc8917385)

[3.6 Création d’une annonce 10](#_Toc8917386)

[3.7 Modification d’une annonce 11](#_Toc8917387)

[3.8 Suppression d’une annonce 12](#_Toc8917388)

[3.9 Détails de notre profil 13](#_Toc8917389)

[3.10 Modification de notre profil 14](#_Toc8917390)

[3.11 Suppression de notre profil 16](#_Toc8917391)

[3.12 Admin 16](#_Toc8917392)

[4. Analyse organique 17](#_Toc8917393)

[4.1 Base de données 17](#_Toc8917394)

[4.1.1 Modèle conceptuel proposé 17](#_Toc8917395)

[4.1.2 Modèle conceptuel modifié 17](#_Toc8917396)

[4.1.3 Modèle physique 18](#_Toc8917397)

[4.1.4 Détails des tables 19](#_Toc8917398)

[4.1.4.1 Table « users » 19](#_Toc8917399)

[4.1.4.2 Table « advertisements » 19](#_Toc8917400)

[4.1.4.2 Table « rates » 19](#_Toc8917401)

[4.1.4.2 Table « pictures » 19](#_Toc8917402)

[4.1.4.2 Table « roles » 19](#_Toc8917403)

[4.2 Architecture du code 20](#_Toc8917404)

[4.2.1 Arborescence de fichier 20](#_Toc8917405)

[4.2.2 Les classes 20](#_Toc8917406)

[4.2.2.1 Classe « User » 20](#_Toc8917407)

[4.2.2.2 Classe « Advertisement » 20](#_Toc8917408)

[4.2.2.3 Classe « Rating » 21](#_Toc8917409)

[4.2.2.4 Classe « Picture » 21](#_Toc8917410)

[4.2.3 Les manageurs 21](#_Toc8917411)

[4.2.3.1 Classe « UserManager » 21](#_Toc8917412)

[4.2.3.2 Classe « AdvertisementManager » 22](#_Toc8917413)

[4.2.3.3 Classe « RatingManager » 23](#_Toc8917414)

[4.2.3.4 Classe « PictureManager » 24](#_Toc8917415)

[5. Tests 24](#_Toc8917416)

[6. Améliorations possibles 24](#_Toc8917417)

[7. Conclusion 24](#_Toc8917418)

[7.1 Difficultés rencontrées 24](#_Toc8917419)

[7.2 Ressentie 24](#_Toc8917420)

[8. Bibliographie 24](#_Toc8917421)

[9. Annexes 25](#_Toc8917422)

# 1. Introduction

## 1.1 Généralités

Arrivant à la fin de ma formation d’informaticien, je dois fournir un Travail Pratique Individuelle (TPI) pour pouvoir obtenir mon certificat d’informaticien. J’ai choisi de réaliser un projet web et l’on m’a proposé Direct Prod.

## 1.2 Pourquoi ce projet

Début 2019 on m’a demandé quel type de projet je voulais réaliser pour mon TPI, j’ai décidé de faire un site web car durant ma troisième j’ai principalement fait du web, laissant le C# de côté. Par la suite, n’ayant pas d’idée de projet précis à faire j’ai préféré laisser mon maître d’apprentissage choisir le site, sachant que ce serai sans doute un site de vente.

# 2. Cahier des charges

## 2.1 Objectif

L’objectif de ce projet web est de réaliser une plateforme d’annonces pour des produits de consommation, où les utilisateurs pourront voir, évaluer et créer des annonces.

## 2.2 Tâches & Fonctionnalités

Pendant ce projet, j’ai plusieurs tâches et fonctionnalités à développer. Pour ce qui est des tâches les voici :

* Réalisation des maquettes du site
* Création de la base de données
* Réalisation des pages du site
* Élaboration d’un rapport de test
* Réalisation d’une documentation technique
* Réalisation d’un manuel utilisateur
* Conception d’un journal de bord
* Réalisation d’un résumé du travail

Ensuite, les fonctionnalités :

* Inscription, connexion et déconnexion d’un utilisateur
* Création d’annonce
* Modification d’une annonce
* Suppression d’une annonce
* Modification du profil
* Suppression du profil
* Évaluation d’une annonce
* Validation d’une annonce
* Recherche des annonces

## 2.3 Matériels & logiciels

Pour la réalisation de ce projet j’ai eu à ma disposition ce matériels :

* 1 ordinateur de l’école
* 2 écrans
* 1 journal de bord vierge

Puis, comme logiciel j’ai utilisé :

* Apache (v. 2.4.28)
* Php (v. 7.0)
* MySQL Workbench (v. 8.0)
* Visual Studio Code
* Pencil
* Bootstrap (v. 4.2.1)
* Microsoft Office 2016

# 3. Analyse fonctionnelle

## 3.1 Barre de navigation

Dans mon site il y aura plusieurs barres de navigation suivant le rôle de l’utilisateur.

### 3.1.1 Barre de navigation non connecté

C

C

A

B



Figure 1 : Barre de navigation d’un utilisateur non connecté

A : Ce lien redirige l’utilisateur vers la page d’accueil (voir Fig. 6).

B : Ce lien redirige l’utilisateur vers la page de connexion (voir Fig. 5).

C : Ce lien redirige l’utilisateur vers la page d’inscription (voir Fig. 4).

### 3.1.2 Barre de navigation connecté

C

C

B

A



Figure 2 : Barre de navigation d’un utilisateur connecté

A : Ce lien redirige l’utilisateur vers la page d’accueil (voir Fig. 7).

B : Ce lien redirige l’utilisateur vers la page de son profil (voir Fig. 13).

C : Ce lien permet la déconnexion de l’utilisateur.

### 3.1.3 Barre de navigation de l’administrateur

D

C

C

C

B

A



Figure 3 : Barre de navigation de l’administrateur

A : Ce lien redirige l’administrateur vers la page d’accueil (voir Fig. 7).

B : Ce lien redirige l’administrateur vers la page de son profil (voir Fig. 13).

C : Ce lien redirige l’administrateur vers la page d’administration (voir Fig. 16).

D : Ce lien permet la déconnexion de l’administrateur.

## 3.2 Créer un compte

Sur cette page l’utilisateur peut s’inscrire au site. Tous les champs sont obligatoires. Les seules règles à respecter pour pouvoir se créer un compte sont :

* L’email ne doit pas déjà être utilisé
* Les deux mots de passe doivent être égaux

I

H

G

F

E

D

C

B

A

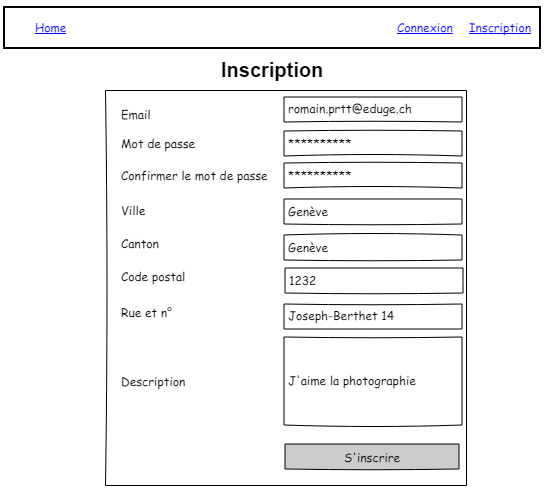


Figure 4 : Créer un compte

A : Email de l’utilisateur

B : Mot de passe de l’utilisateur

C : Confirmation du mot de passe

D : Ville de résidence de l’utilisateur

E : Canton de résidence de l’utilisateur

F : Code postal de l’utilisateur

G : Rue et numéro de résidence de l’utilisateur

H : Description de l’utilisateur

I : Bouton validant l’inscription de l’utilisateur.

## 3.3 Connexion

Cette page permet à un utilisateur de se connecter au site, pour cela il doit spécifier son email et son mot de passe.

B

A

C

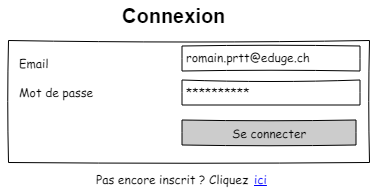


Figure 5 : Connexion

A : Email de l’utilisateur

B : Mot de passe de l’utilisateur

C : Bouton de connexion

D : Lien redirigeant sur la page de création d’un compte (voir Fig. 4).

## 3.4 L’accueil

### 3.4.1 L’accueil non connecté

L’accueil est la page principale de mon site, les annonces affichées sont uniquement celles validées par l’administrateur. On peut la consulter sans être connecté mais la fonctionnalité de création d’une annonce est désactivée.

B

A

C

C1



Figure 6 : Page d’accueil d’un utilisateur non connecté

A : Boite de saisie pour faire une recherche dans les annonces.

B : Lance la recherche.

C : Affichage d’une annonce avec ses détails importants.

C1 : Bouton redirigeant vers la page des détails de l’annonce (voir Fig. 8)

### 3.4.2 L’accueil connecté

Lorsqu’on est connecté le bouton pour créer une annonce apparait (Point A).

De plus, si une annonce nous appartient (Point B) deux boutons supplémentaires apparaissent sur l’annonce (Point B2, B3).

BV

A

C

D3

D2

D1

D

E



Figure 7 : Page d’accueil d’un utilisateur connecté

A : Boite de saisie pour faire une recherche dans les annonces.

B : Lance la recherche.

C : Bouton permettant de créer une annonce (voir Fig. 10).

D : Affichage d’une annonce m’appartenant, c’est pourquoi deux nouveaux boutons apparaissent.

D1 : Bouton permettant d’être redirigé vers la page des détails de l’annonce (voir Fig. 9)

D2 : Redirection vers la page de modification de l’annonce (voir Fig. 11).

D3 : Redirection vers la page de suppression (voir Fig. 12).

E : Affichage d’une annonce avec ses détails importants.

## 3.5 Détails d’une annonce

Cette page affiche toutes les informations ainsi que les images liées à l’annonce.

### 3.5.1 Détails d’une annonce non connecté

B

A1

A

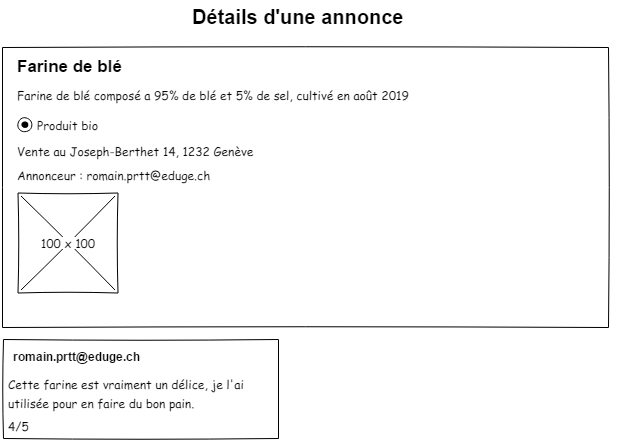


Figure 8 : Détails d’une annonce avec un utilisateur non connecté

A : Description de l’annonce avec affichages des images liées.

A1 : Affichage des images liés en miniature, si on clique sur l’image elle s’ouvre en grand.

B : Commentaire et note laisser par un utilisateur.

### 3.5.2 Détails d’une annonce connecté

A

B

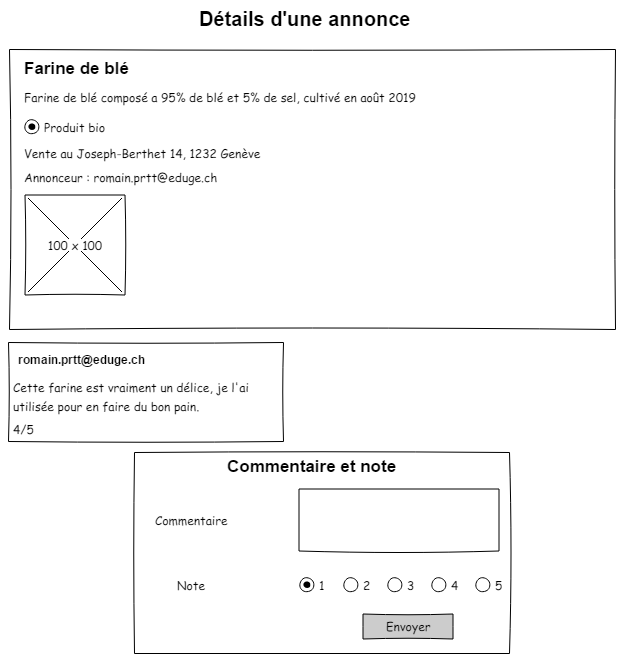


Figure 9 : Détails d’une annonce avec un utilisateur connecté

A : Formulaire permettant de laisser un commentaire et une note à l’annonce.

B : Confirme l’envoie du commentaire et de la note dans la base de données.

## 3.6 Création d’une annonce

Lors de l’appui sur le bouton de création d’une annonce, l’utilisateur est redirigé vers un formulaire. Les champs « Titre », « Description » et « Bio » sont obligatoires, mais il n’est pas nécessaire de mettre une image pour pouvoir créer une annonce.

D

C

B

A

E

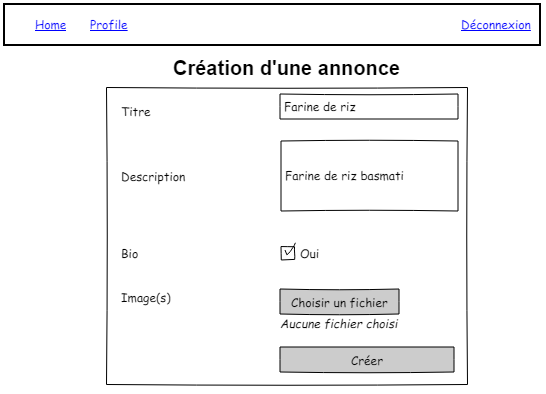


Figure 10 : Formulaire de création d’une annonce

A : Le titre de l’annonce

B : La description du produit

C : Case à cocher si le produit est bio

D : L’utilisateur peut choisir une ou plusieurs images pour son annonce

E : Bouton Confirmant la création de l’annonce

## 3.7 Modification d’une annonce

Lors de la modification d’une annonce tous les champs du formulaire sont remplis avec les informations de la base de données.

F

E

DA

C

B

A

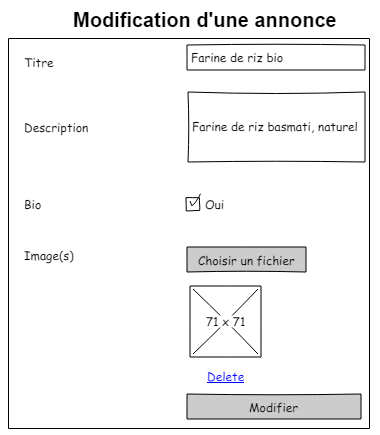


Figure 11 : Modification d’une annonce

A : Le titre de l’annonce

B : La description du produit

C : Case à cocher si le produit est bio

D : L’utilisateur peut choisir une ou plusieurs images pour son annonce

E : Les images anciennement choisies sont affichées avec un lien supprimant l’image.

F : Confirme la modification des informations dans la base de données.

## 3.8 Suppression d’une annonce

La page de suppression d’une annonce est surtout une confirmation de la part de l’utilisateur qu’il veut bel et bien supprimer cette annonce.

B

A

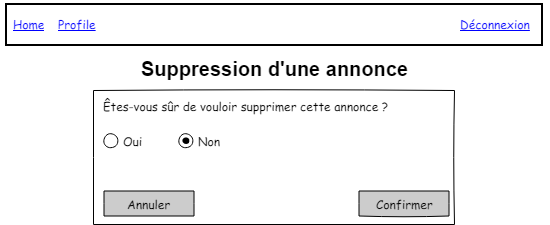


Figure 12 : Confirmation de la suppression d’une annonce

A : Redirige vers la page d’accueil.

B : Supprime votre compte de la base de données et redirige vers la page d’accueil.

## 3.9 Détails de notre profil

Cette page montre les détails de notre profil ainsi que toutes les annonces créées.

B3

B2

B1

A2

A1

B

A

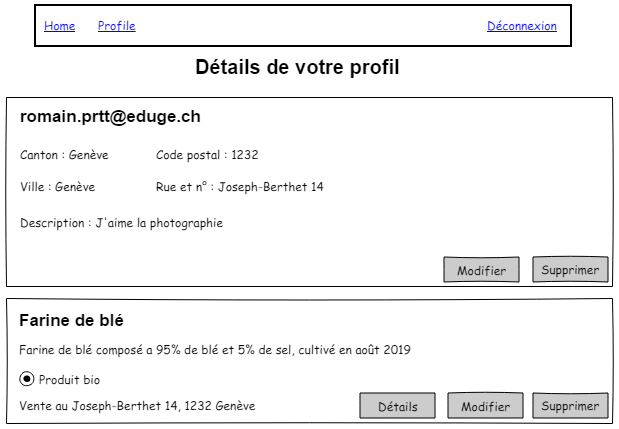


Figure 13 : Détails de notre profil

A : Description de l’utilisateur.

A1 : Redirige vers la page de modification de son profil (voir Fig. 14).

A2 : Redirige vers la page de suppression de son compte (voir Fig. 15).

B : Affichage des annonces nous appartenant.

B1 : Bouton permettant d’être redirigé vers la page des détails de l’annonce (voir Fig. 9)

B2 : Redirection vers la page de modification de l’annonce (voir Fig. 11).

B3 : Redirection vers la page de suppression (voir Fig. 12).

## 3.10 Modification de notre profil

Lors de la modification de notre profil tous les champs du formulaire sont remplis avec les informations de la base de données sauf pour les mots de passe.

L’email n’est pas modifiable car il sert d’identifiant pour l’utilisateur. Les champs « Nouveau mot de passe » et « Confirmer le nouveau mot de passe » ne sont pas obligatoire, si l’on ne veut pas changer de mot de passe. Pour confirmer le changement des informations l’utilisateur doit taper son mot de passe actuel dans le champ à cet effet et cliquer sur le bouton « Modifier ».

I

H

GA

E

F

D

C

B

A

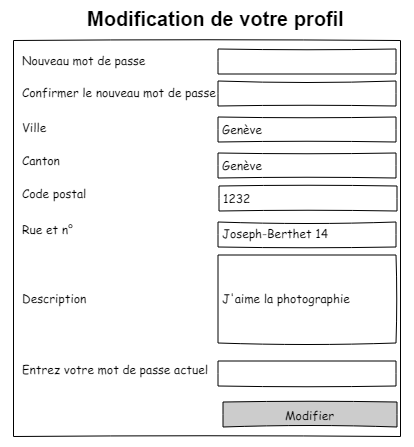


Figure 14 : Modification de notre profil

A : Le nouveau mot de passe de l’utilisateur

B : La confirmation du nouveau mot de passe de l’utilisateur

C : Ville de résidence de l’utilisateur

D : Canton de résidence de l’utilisateur

E : Code postal de l’utilisateur

F : Rue et numéro de résidence de l’utilisateur

G : Description de l’utilisateur

H : Le mot de passe de l’utilisateur

I : Bouton validant la modification des informations de l’utilisateur.

## 3.11 Suppression de notre profil

Pour supprimer son profil, on est redirigé vers une page de confirmation de suppression. Pour confirmer la suppression de son profil il faut cocher le radio bouton « Oui » et appuyer sur le bouton.

B

A

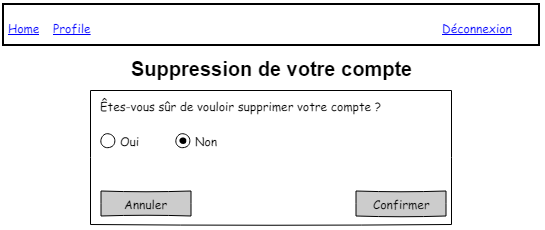


Figure 15 : Confirmation de la suppression de votre compte

A : Redirige vers la page de votre compte.

B : Supprime votre compte de la base de données et redirige vers la page d’accueil.

## 3.12 Admin

Cette page est réservée à l’admin personne d’autre n’y a accès. Elle permet à l’admin de valider ou de refuser les nouvelles annonces que les utilisateurs on créer.

D

C

B

A

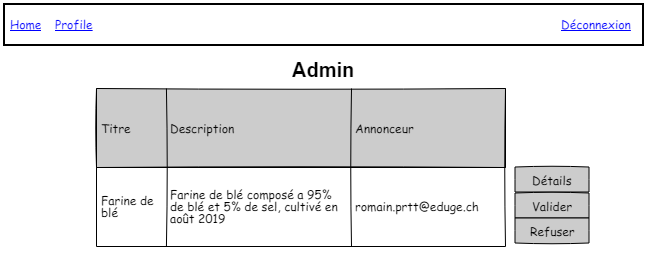


Figure 16 : Administration des nouvelles annonces

A : Informations importantes de l’annonce

B : Redirige vers la page des détails (voir Fig. 9)

C : Valide la nouvelle annonce.

D : Supprime l’annonce de la base de données.

# 4. Analyse organique

## 4.1 Base de données

### 4.1.1 Modèle conceptuel proposé

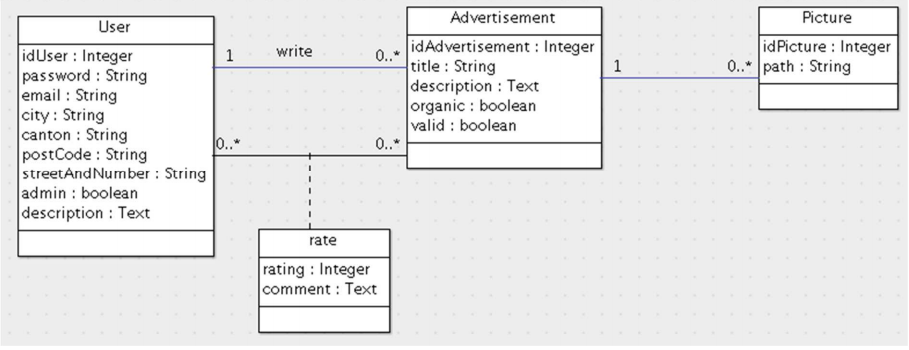


Figure … : Modèle conceptuel proposé

### 4.1.2 Modèle conceptuel modifié

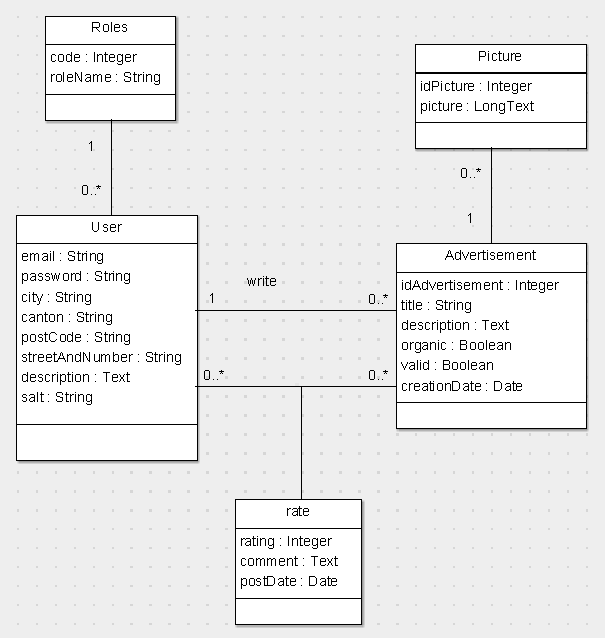


Figure … : Modèle conceptuel modifié

### 4.1.3 Modèle physique

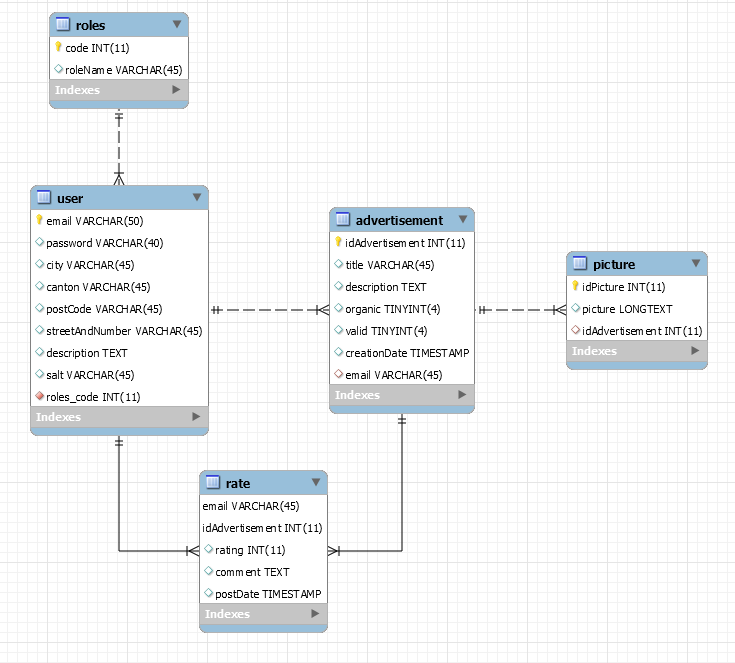


Figure … : Modèle physique

### 4.1.4 Détails des tables

|  |  |
| --- | --- |
| Abréviation | Signification |
| PK | Clé primaire |
| NN | Non nul (Empêche la valeur NULL) |
| AI | Auto incrément |
| CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP |  |

Tableau 1 : Légende des propriétés des attributs

#### 4.1.4.1 Table « users »

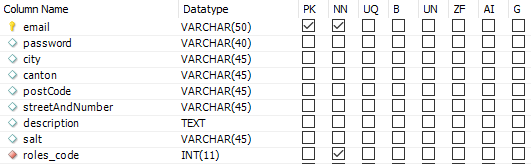


Figure … : Table « users »

#### 4.1.4.2 Table « advertisements »

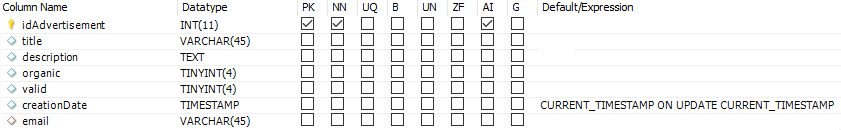


Figure … : Table « advertisements »

#### 4.1.4.2 Table « rates »

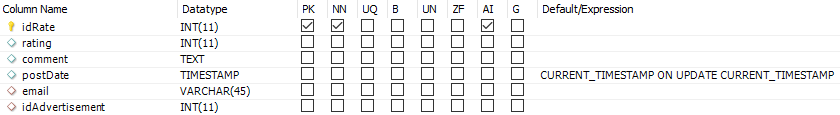


Figure … : Table « rates »

#### 4.1.4.2 Table « pictures »

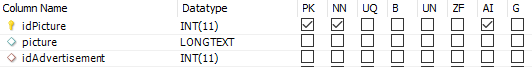


Figure … : Table « pictures »

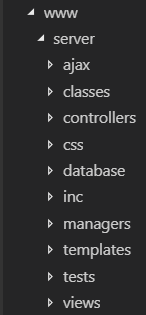
#### 4.1.4.2 Table « roles »



Figure … : Table « roles »

## 4.2 Architecture du code

### 4.2.1 Arborescence de fichier



…

### 4.2.2 Les classes

#### 4.2.2.1 Classe « User »

Cette classe contient les informations sur un utilisateur

|  |  |  |
| --- | --- | --- |
| Nom de la variable | Type | Description |
| **email** | STRING | L’email de l’utilisateur |
| **password** | STRING | Le mot de passe de l'utilisateur |
| **city** | STRING | La ville de l'utilisateur |
| **canton** | STRING | Le canton de l'utilisateur |
| **postCode** | INT | Le code postal de l'utilisateur |
| **streetAndNumber** | STRING | Le numéro et le nom de la rue de l'utilisateur |
| **description** | STRING | La description de l'utilisateur |
| **salt** | STRING | Le salt du mot de passe de l'utilisateur |
| **roles\_code** | INT | Le code du rôle de l'utilisateur |

Tableau 2 : Variables de la classe « User »

#### 4.2.2.2 Classe « Advertisement »

Cette classe contient les informations sur une annonce

|  |  |  |
| --- | --- | --- |
| Nom de la variable | Type | Description |
| **idAdvertisement** | INT | L'id de l'annonce |
| **title** | STRING | Le titre de l’annonce |
| **description** | STRING | La description de l’annonce |
| **organic** | BOOLEAN | Un bool définissant si l'annonce est organique ou non |
| **valid** | BOOLEAN | Un bool définissant si l'annonce est validé par l'admin |
| **creationDate** | STRING | La date de création de l'annonce |
| **userEmail** | STRING | L'email de l'utilisateur auquel l'annonce est assigné |

Tableau 3 : Variables de la classe « Advertisement »

#### 4.2.2.3 Classe « Rating »

|  |  |  |
| --- | --- | --- |
| Nom de la variable | Type | Description |
| **idRate** | INT | L'id de l'évaluation |
| **rating** | INT | La note donné par l'utilisateur |
| **comment** | STRING | Le commentaire |
| **postDate** | STRING | La date de création de l'évaluation |
| **userEmail** | STRING | L'email de l'utilisateur ayant mis un commentaire |
| **idAdvertisement** | INT | L'id de l'annonce auquel l’évaluation est assignée |

Tableau 4 : Variables de la classe « Rating »

#### 4.2.2.4 Classe « Picture »

|  |  |  |
| --- | --- | --- |
| Nom de la variable | Type | Description |
| **idPicture** | INT | L’id de l’image |
| **Picture** | STRING | L’image en base 64 |
| **idAdvertisement** | INT | L’id de l’annonce auquel l’image est assignée |

Tableau 5 : Variables de la classe « Picture »

### 4.2.3 Les manageurs

#### 4.2.3.1 Classe « UserManager »

Les actions que peuvent exécuter les utilisateurs sont gérés par les requêtes SQL ci-dessous.

|  |  |  |
| --- | --- | --- |
| Nom de la fonction | Paramètre(s) | Description |
| **GetUserByEmail** | $email | @brief Retourne un utilisateur en fonction de son email  @param string email L'email de l'utilisateur qu'on recherche.  @return   * [User] L'utilisateur de type User. * False Une erreur est survenue. * Null L'utilisateur n'est pas trouvé. |
| **CreateUser** | $u | @brief Ajoute un utilisateur  @param [User] L'objet User que l'on veut insérer.  @return boolean   * True Ok. * False Une erreur est survenue. * Null L'utilisateur existe déjà. |
| **UpdateUser** | $u | @brief Modifie un utilisateur  @param [User] L'objet User que l'on veut modifier.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **DeleteUser** | $email | @brief Supprime un utilisateur en fonction de son email  @param string email L'email de l'utilisateur qu'on veut supprimer.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **UserExist** | $email | @brief Vérifie si un utilisateur existe déjà avec l'email spécifié  @param string email L'email de l'utilisateur qu'on recherche.  @return boolean   * True L'utilisateur existe. * False Une erreur est survenue. * Null L'utilisateur n'existe pas. |
| **Login** | $email, $password | @brief Essaie de connecter l'utilisateur  @param string email L'email de l'utilisateur.  @param string password Le mot de passe de l'utilisateur.  @return boolean   * True La connexion a réussie. * False Une erreur est survenue. * Null L'utilisateur n'existe pas |

Tableau 6 : Fonctions de la classe « UserManager »

#### 4.2.3.2 Classe « AdvertisementManager »

|  |  |  |
| --- | --- | --- |
| Nom de la fonction | Paramètre(s) | Description |
| **GetValidatedAds** | / | @brief Retourne toutes les annonces validées par l'administrateur  @return array   * Un tableau contenant toutes les annonces de type "Advertisement". * False Une erreur est survenue. |
| **GetInvalidAds** | / | @brief Retourne toutes les annonces non validées par l'administrateur  @return array   * Un tableau contenant toutes les annonces de type "Advertisement". * False Une erreur est survenue. |
| **GetAdById** | $idAd | @brief Retourne l'annonce que l'on recherche par son identifiant  @param int idAd L'id de l'annonce que l'on recherche.  @return [Advertisement]   * L'annonce de type "Advertisement". * False Une erreur est survenue. * Null L'annonce n'est pas trouvée. |
| **GetAdsByUserEmail** | $email | @brief Retourne l'annonce que l'on recherche par l'identifiant de son annonceur  @param string email L'email de l'annonceur.  @return array   * Un tableau contenant toutes les annonces de type "Advertisement". * False Une erreur est survenue. |
| **CreateAd** | $a | @brief Ajout d'une annonce  @param [Advertisement] L'objet "Advertisement" qu'on veut insérer dans la base.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **UpdateAd** | $a | @brief Modification d'une annonce  @param [Advertisement] L'objet "Advertisement" qu'on veut modifier dans la base.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **UpdateAdToValid** | $idAd | @brief Valide une annonce  @param int idAd L'id de l'annonce que l'on veut validée.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **DeleteAd** | $idAd | @brief Suppression d'une annonce  @param int idAd L'id de l'annonce qu'on veut supprimer.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **DeleteAdsOfUser** | $email | @brief Suppression de toutes les annonces d'un utilisateur  @param string email L'email de l'utilisateur.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **Research** | $searchContent, $organic | @brief Retourne les annonces en rapport avec la recherche  @param string searchContent La recherche de l'utilisateur.  @param boolean organic Si l'utilisateur recherche que des annonces bio.  @return array   * Un tableau contenant toutes les annonces de type "Advertisement". * False Une erreur est survenue. |

Tableau 7 : Fonctions de la classe « AdvertisementManager »

#### 4.2.3.3 Classe « RatingManager »

|  |  |  |
| --- | --- | --- |
| Nom de la fonction | Paramètre(s) | Description |
| **GetRatingsOfAnAd** | $idAd | @brief Retourne toutes les commentaires et notes en rapport avec l'annonce spécifié  @param int idAd L'id de l'annonce qu'on recherche  @return array   * Un tableau contenant toutes les commentaires et notes de type "Rating". * False Une erreur est survenue. |
| **GetScoreOfAnAd** | $idAd | @brief Retourne le score donné par tous les commentaires des utilisateurs  @param int idAd L'id de l'annonce qu'on recherche  @return float   * Un nombre avec un chiffre derrière la virgule étant la moyenne de tous les commentaires. * False Une erreur est survenue. |
| **CreateRating** | $r | @brief Ajout d'un commentaire et d'une note  @param [Rating] Le commentaire et la note de type "Rating"  @return boolean   * True Ok. * False Une erreur est survenue. |
| **DeleteRatingsOfAnAd** | $idAd | @brief Suppression de toutes les évaluations d'une annonce  @param int idAd L'id de l'annonce.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **DeleteRatingsOfUser** | $email | @brief Suppression de toutes les évaluations d'un utilisateur  @param string email L'email de l'utilisateur.  @return boolean   * True Ok. * False Une erreur est survenue. |

Tableau 8 : Fonctions de la classe « RatingManager »

#### 4.2.3.4 Classe « PictureManager »

|  |  |  |
| --- | --- | --- |
| Nom de la fonction | Paramètre(s) | Description |
| **GetPictureById** | $idPicture | @brief Retourne l'annonce que l'on recherche par l'identifiant de son annonceur  @param int idPicture L'id de l'image qu'on recherche.  @return [Picture]   * L'image de type "Picture". * False Une erreur est survenue. * Null Aucune image. |
| **GetPicturesByAdId** | $idAd | @brief Retourne les images en rapport avec l'annonce spécifié  @param int idPicture L'id de l'annonce qu'on recherche.  @return array   * Un tableau contenant toutes les images. * False Une erreur est survenue. |
| **CreatePicture** | $p | @brief Ajout d'une image  @param int idAd L'id de l'annonce auquel est lié l'image.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **DeletePicture** | $idPicture | @brief Suppression d'une image  @param int idPicture L'id de l'image à supprimer.  @return boolean   * True Ok. * False Une erreur est survenue. |
| **DeletePicturesOfAnAd** | $idAd | @brief Suppression des images lié à une annonce  @param int idAd L'id de l'annonce.  @return boolean   * True Ok. * False Une erreur est survenue. |

Tableau 9 : Fonctions de la classe « PictureManager »

# 5. Tests

# 6. Améliorations possibles

# 7. Conclusion

## 7.1 Difficultés rencontrées

## 7.2 Ressentie

# 8. Bibliographie

<https://www.w3schools.com/sql>

# 9. Annexes