

**Livrable technique sur l’application ministage**



BTS SIO

Valentine Duprez – Quentin Chéruel – Romain Provoste

Table des matières

[**Présentation** 2](#_Toc64554674)

[**Architecture de l’application web** 3](#_Toc64554675)

[**Le modèle :** 4](#_Toc64554676)

[**Le contrôleur :** 12](#_Toc64554677)

[**La vue :** 16](#_Toc64554678)

[**La connexion :** 19](#_Toc64554679)

[**La base de données :** 20](#_Toc64554680)

[**Exportation du code :** 21](#_Toc64554681)

# **Présentation**

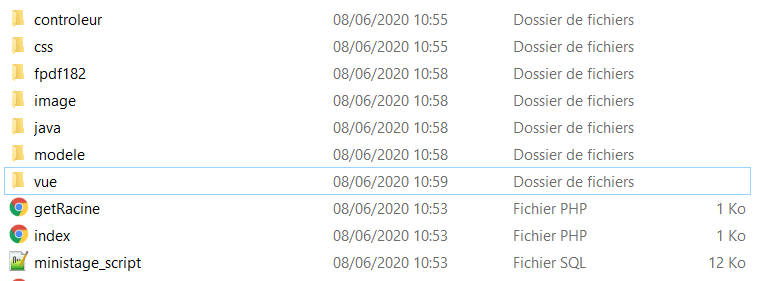
Nous sommes au sein du lycée Jean Rostand.

Le Directeur Délégué aux Formations Professionnelles et Technologiques en établissement (DDFPT) nous à demander de mettre en place une application qui sera simple d’utilisation et moderne qui lui permettra d’automatiser la gestion des mini stage afin que les établissements puissent inscrire leurs élèves au stage qui leur seront proposé.

L’objectif de ce projet est le gain de temps, la simplification de la gestion des mini stages et l’automatisation de la gestion.

# **Architecture de l’application web**

Voici l’architecture de l’application web :



La partie contrôleur permet de récupérer les informations de les traités en fonction des paramètres demandé par la vue et de renvoyer les données à la vue.

La partie modèle permet la récupération de toutes les données de la base de données et permet de crée des fonctions qui seront utilisé dans le contrôleur.

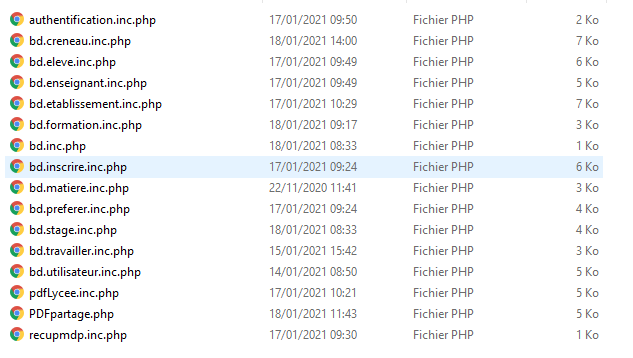
Le fpdf182 permet la création du PDF.

Le dossiers java permet l’autocomplète des formulaires.

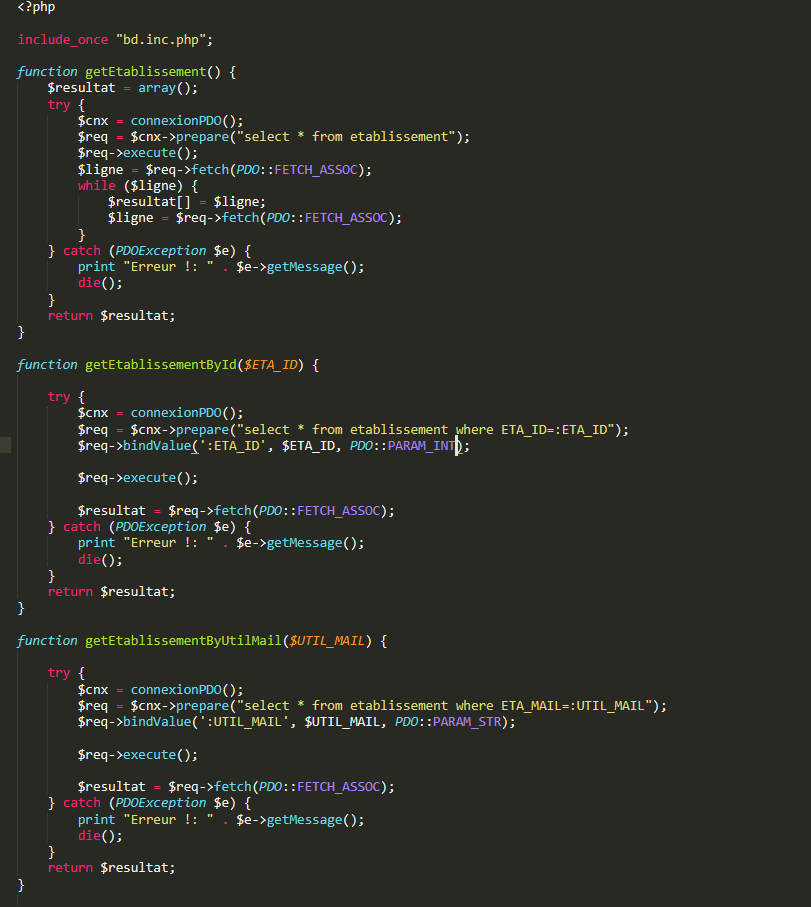
Le dossier css permet d’affiche le design des pages.

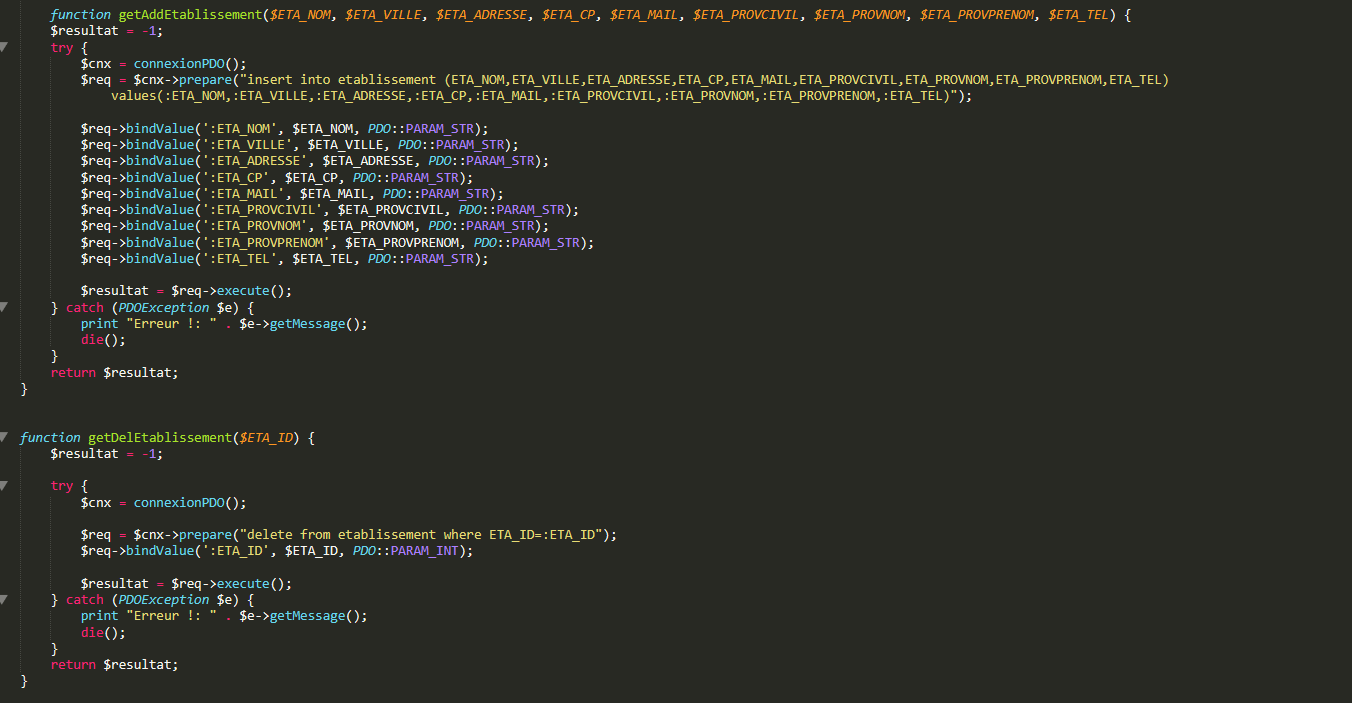
# **Le modèle :**

Voici les fichiers présents dans le modèle :



Voici le contenu de « bd.etablissement.inc.php » :







La fonction getEtablissement permet de récupérer tous les élément présent dans la table établissement pour réussir cela nous avons dû faire recours à une requête SQL « select \* from etablissement" qui permet de tout sélection dans la BDD, on peut voir dans cette fonction que le résultat qui nous sera retourné sera sous forme de tableau grâce à cette ligne « $resultat = array(); ».

La fonction getEtablissementById($ETA\_ID) permet de récupérer toutes les données selon l’id de l’établissement sélectionner dans un formulaire.

« $req->bindValue(':ETA\_ID', $ETA\_ID, PDO::PARAM\_INT); » cette ligne permet de définir la variable ETA\_ID qui sera saisie comme étant de type « integer ».

La fonction getEtablissementByUtilMail($UTIL\_MAIL) fonctionne de la même manière que la précédente sauf que là on saisit le mail de l’établissement qui est égale à celui d’un des utilisateurs. Cette fonction sera utilisée pour la connexion.

La fonction suivante permet l’ajout d’un établissement « function getAddEtablissement($ETA\_NOM, $ETA\_VILLE, $ETA\_ADRESSE, $ETA\_CP, $ETA\_MAIL, $ETA\_PROVCIVIL, $ETA\_PROVNOM, $ETA\_PROVPRENOM, $ETA\_TEL) », elle prend en paramètre tous les variables qui devront être saisie par l’utilisateur.

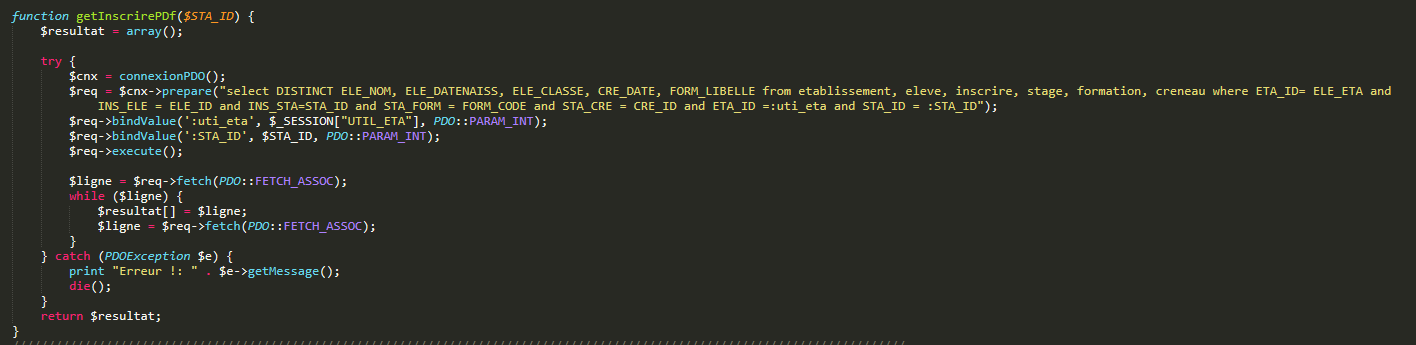
Ensuite, on exécute la requête sql « insert into etablissement (ETA\_NOM,ETA\_VILLE,ETA\_ADRESSE,ETA\_CP,ETA\_MAIL,ETA\_PROVCIVIL,ETA\_PROVNOM,ETA\_PROVPRENOM,ETA\_TEL) values(:ETA\_NOM,:ETA\_VILLE,:ETA\_ADRESSE,:ETA\_CP,:ETA\_MAIL,:ETA\_PROVCIVIL,:ETA\_PROVNOM,:ETA\_PROVPRENOM,:ETA\_TEL) » qui permet l’ajout dans la BDD.

La fonction de suppression « getDelEtablissement($ETA\_ID) » va prendre en paramètre la clé primaire et va exécuter la requête SQL permettant la suppression.

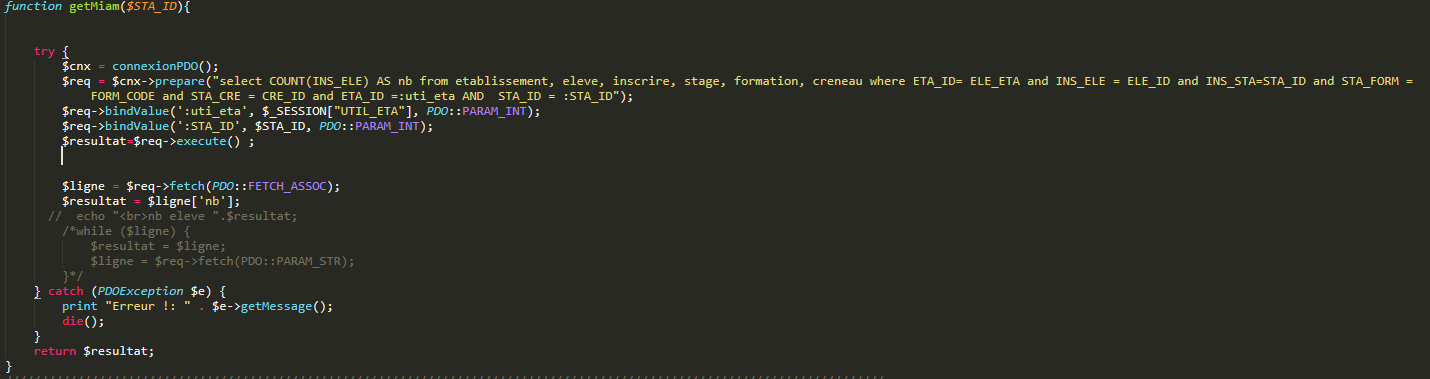
Pour finir la dernière requête « getUpdateEtablissement($ETA\_NOM,$ETA\_VILLE,$ETA\_ADRESSE,$ETA\_CP,$ETA\_MAIL,$ETA\_PROVCIVIL,$ETA\_PROVNOM,$ETA\_PROVPRENOM,$ETA\_TEL,$ETA\_ID) » qui va permettre de modifier les données selon la clé primaire sélectionné.

**Pour les pages non-similaires :**

Pour pdfLycee.inc.php on utilise les mêmes fonctions.



Voici la première qui permet de récupérer les données de la BDD et de les affichés dans le pdf.



Voici la seconde fonction qui permet de compter le nombre d’élève et d’afficher le résultat ce qui permettra de commande le nombre de repas correspondant et de les facturé au lycée ou collège correspondant.

On déclare une nouvelle class PDF qui est une extension de FPDF.



A l’intérieur on déclare la fonction Header() qui sera l’entête du PDF ensuite on déclare la fonction Footer() qui sera le pied de page du PDF.

Ensuite nous avons les fonction Table et viewTable qui permette la création d’un tableau avec les données de la b ase de données, comme on peut le dans viewTable on utilise une structure Itérative avec for qui permet d’afficher toute les données présente demande.

Pour l’authentification :



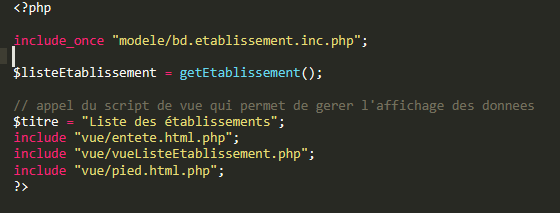


Le premier screen nous montre les fonctions qui vont nous permettre de nous connecter selon la personne qui se connectera on ferra appelle à différente fonctions qui vont détermine notre type d’utilisateur c’est-à-dire admin, établissement, enseignant.

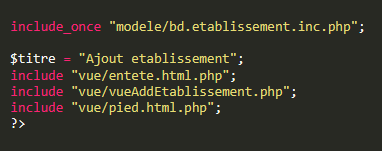
Le second screen nous montre les fonction permettant l’ajout d’utilisateurs.

# **Le contrôleur :**

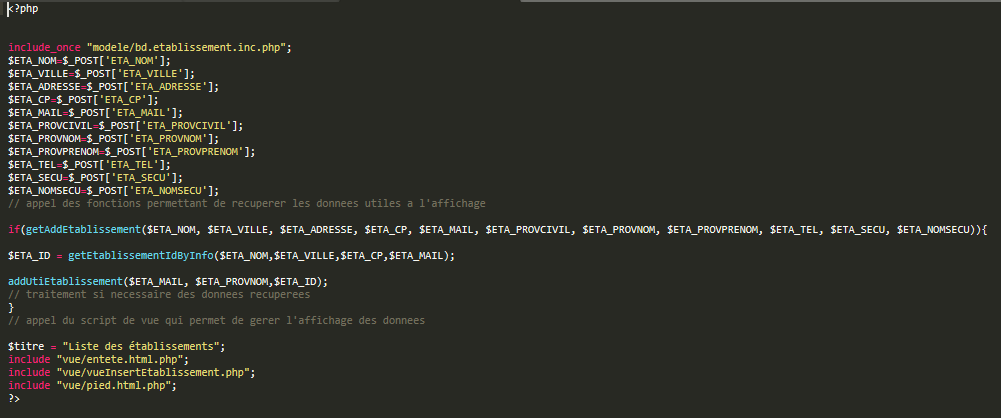
Voici le script du contrôleur permettant l’obtention de la liste des établissements.



Voici AddEtablissemment  :



Voici insertEtablissement :



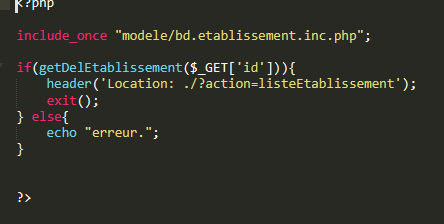
Voici modifEtablissement :



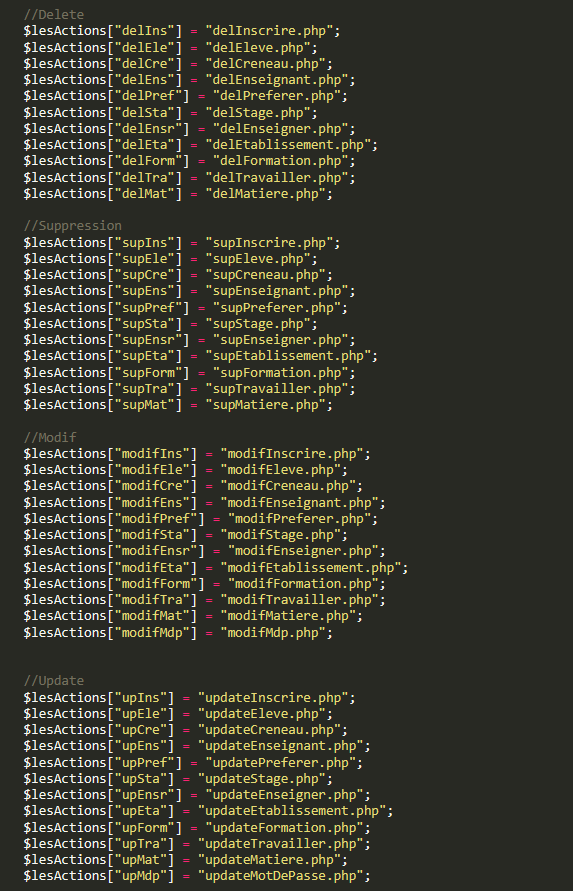
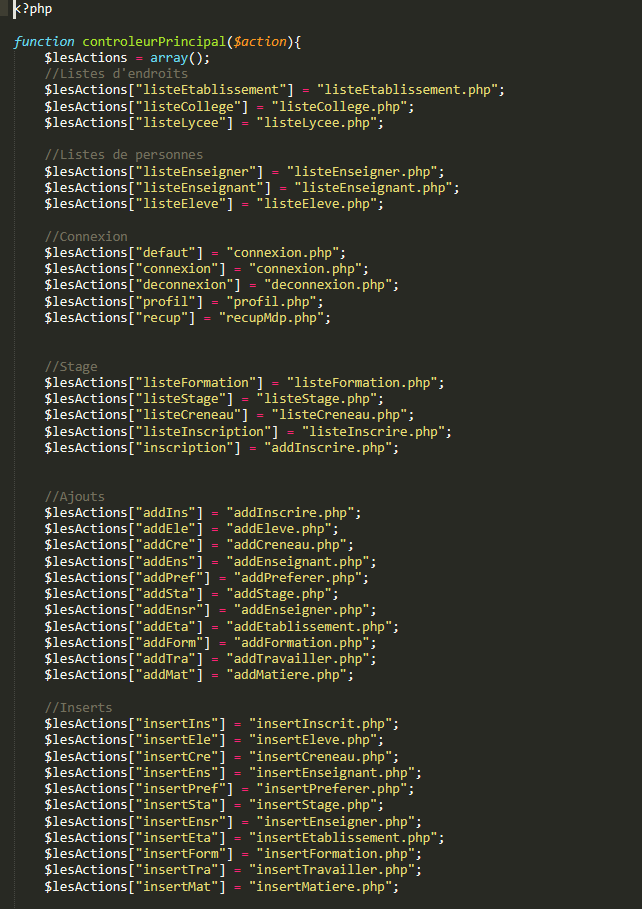
Voici updateEtablissement:



Voici delEtablissement:



Voici ce à quoi ressemble le contrôleur Principal :

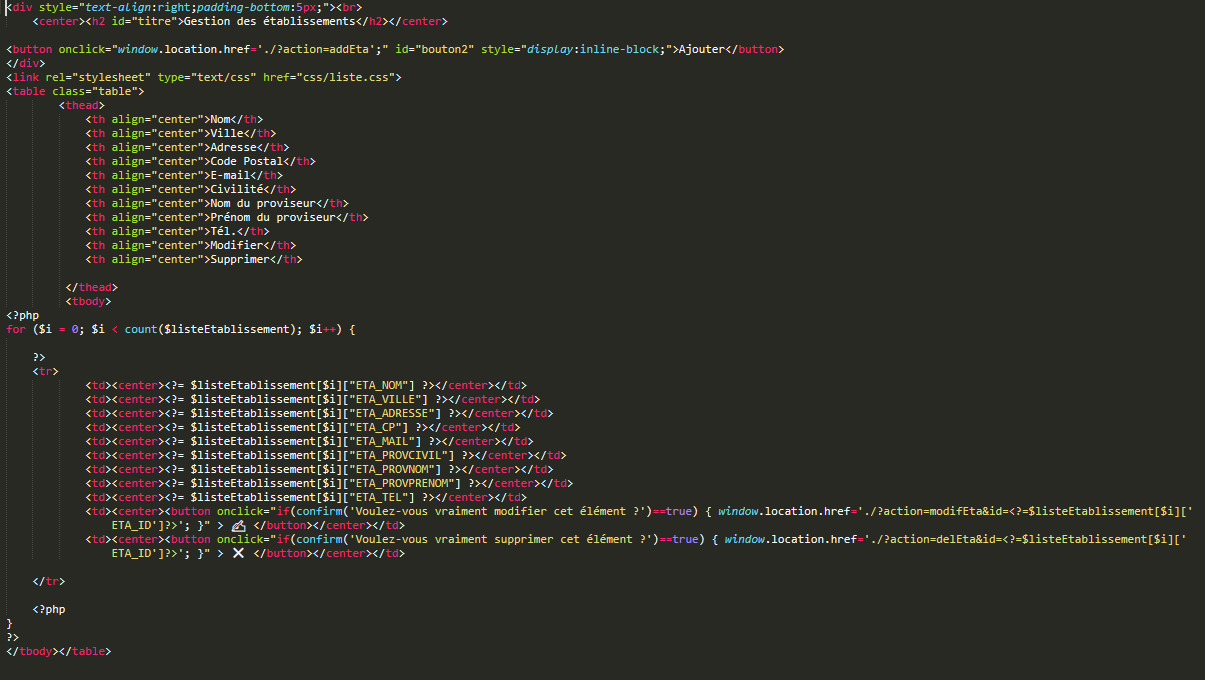


Voici le contrôleur permettant de la connexion à un compte :



# **La vue :**

Voici vueListeEtablissement:



Voici le formulaire de modification:

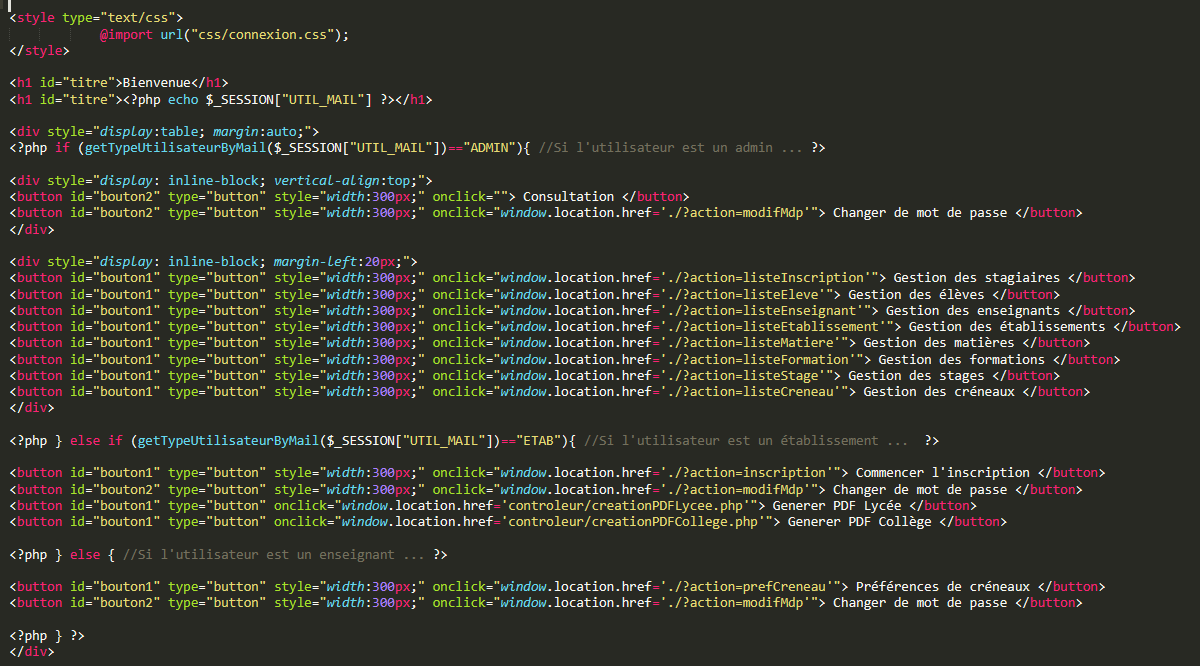


Voici le formulaire d’ajout :



Toutes les autres vues sont similaires sauf pour les pdf et la vue du profil.

Voici pour vueProfil.php :



L’administrateur aura accès à la gestion des élèves, enseignant, établissements, matières, stage, formation et créneau.

Les établissements auront accès au changement de mot de passe et à l’inscription des élèves ainsi qu’à la génération du PDF.

Les enseignants pourront choisir les créneaux qu’il préfère et pourront changer leur mot de passe.

Voici une partie du script du PDF :



« $pdf = new PDF(); » est l’instanciement de la classe dérive de FPDF

AliasNbPages est une fonction qui permettra d’afficher le nombre de page dans le footer.

utf8\_decode sert à affiché les accent des mots sans les remplacé par d’autres caractère spéciaux.

setFront permet de définir la police telle que « $pdf->SetFont('Arial','BU',10); » veut dire que la police sera Arial, de type en gras et souligné en taille 10.

MultiCell permet de crée plusieurs cellules d’écriture qui s’adapteront à la taille de la page.

# **La connexion :**

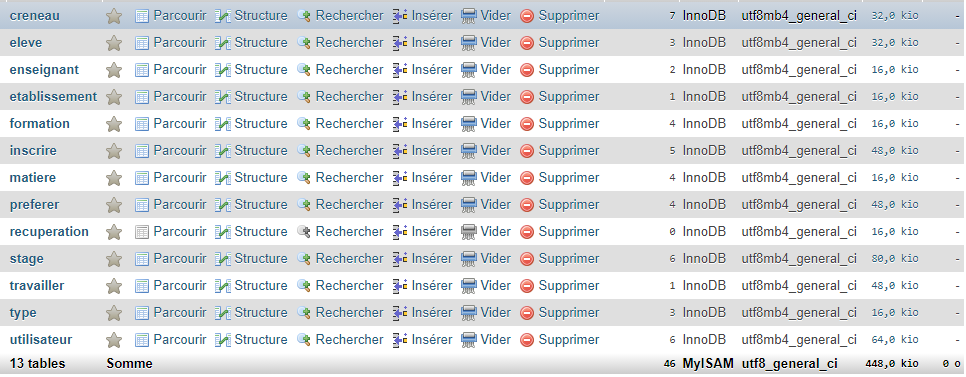
Lors de la création d’un compte enseignant le mot de passe sera la date de naissance de l’enseignant version américaine avec des tirets, exemple :

Identifiant : [test@enseignant.fr](mailto:test@enseignant.fr)

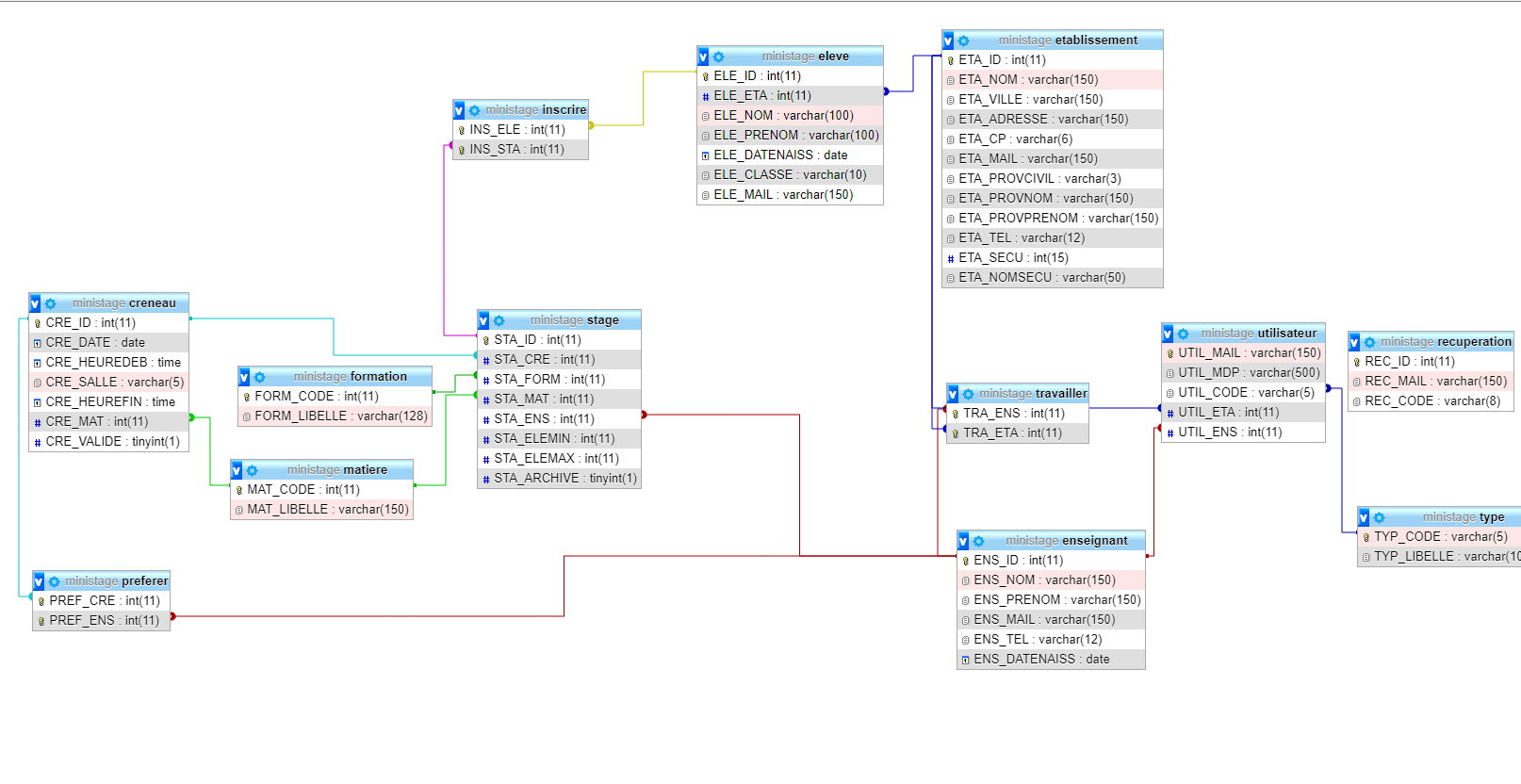
Mot de passe : 2000-12-21

# **La base de données :**

Voici les tables de la base de données :



Voici une modélisation de la base de données avec les tailles des caractères et le type de données qui seront accepté :



# **Exportation du code :**

Si l’application web vient à être exporté dans d’autres établissement, il conviendra de changer le logo depuis le dossiers image en donnant exactement le même nom et la même extension à l’image, ensuite pour la création de compte, vous devrait vous connecter avec le compte admin par défaut, ensuite crée votre établissement, aller voir l’id de votre établissement dans la bdd et ensuite changer dans la fonction « getAddTravailler($TRA\_ENS) » la ligne « $req->bindValue(':uti\_eta', 8, PDO::PARAM\_INT); » remplacer le 8 par l’id de votre établissement.