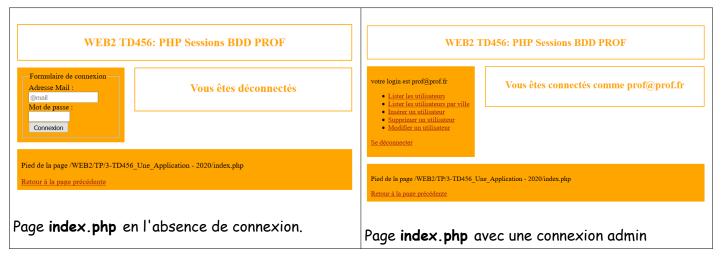
1<sup>ère</sup> Année **IUT LANNION** R209 - WEB2: Initiation au développement Web BUT Réseaux informatiques et Télécoms

TD45 (2\*1H45) + TP34 (2\*1H45) : Une application PHP

#### 1. Présentation de L'Application

### 1.1. Introduction

On se propose d'écrire une application qui va mettre en œuvre les sessions et le couplage avec une base de données. Cette application utilisera trois pages index.php, insertion.php modification.php



### 1.2. ETUDE DE LA MAQUETTE

L'application se décompose en quatre zones, le bandeau et le pied de
page sont statiques.
Relier les différentes zones de la maquette à l'IHM.
Le fichier de la maquette index.php et la feuille de style CSS sont
· · · · · · · · · · · · · · · · · · ·
fournis

# 1.3. Base de données SQLITE

Le fichier IUT.sqlite est fourni. Editer le avec le logiciel DB Browser for SQLite. Analyser le schéma de la BDD.

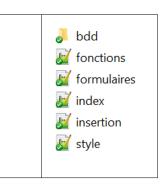
### 1.4. ORGANISATION DES FICHIERS

- 1 Placer dans un fichier
  - formulaires.php: toutes les fonctions qui proposent un formulaire.
  - fonctions.php: toutes les autres fonctions utilisées.

Inclure ces fichiers depuis les pages de votre application :

include 'fonctions.php'; include 'formulaires.php'

2 - Ranger votre BDD dans un dossier \bdd protégé par .htaccess



# 2. FORMULAIRE DE CONNEXION ET SESSION UTILISATEUR (TD)

Dans cette partie qui se concentre pour l'instant sur la page **index.php** on souhaite vérifier que l'utilisateur est inscrit dans la base de données et ensuite l'autoriser à utiliser l'application à l'aide des sessions PHP.



La fonction afficheFormulaireConnexion() est fournie dans formulaires.php. Citer les éléments importants du formulaire.

La fonction compteExiste(\$login,\$pass) est fournie dans fonctions.php (elle retourne un booléen).

- 1. Ecrire la fonction is Admin(\$login) qui renvoie un booléen.
- 2. Tester ces fonctions indépendamment de l'application en créant un script de test unitaire testCompte.php
- 3. Compléter ensuite le script index.php fourni afin de tester l'existence du compte, d'écrire des variables de session 'login' et 'statut' (membre ou admin ) puis d'afficher le menu avec la fonction afficheMenu(). Ce menu dépendra du statut de la personne connectée (cf capture).

# 3. PAGE INDEX.PHP : LISTER LES UTILISATEURS (TD)

Rappel : La fonction affiche Tableau (\$tab) fournie affiche sous forme d'une table HTML un tableau php de tableaux associatifs (2 dimensions) passé en paramètre quelle que soit la taille du tableau (avec les entêtes des colonnes).

## 3.1. FONCTION LISTECOMPTE()

- 1. Ecrire une fonction listeCompte dans fonctions.php qui retourne un tableau contenant les adresses mail, les statuts et les adresses de tous les utilisateurs présents dans la base de données.
- 2. Tester ces deux fonctions dans un script de test unitaire (testListeCompte.php)



### 3.2. LISTER LES COMPTES DANS L'APPLICATION

Compléter la zone **article** afin d'ajouter cette fonctionnalité dans l'application lorsque l'on clique sur le lien correspondant (en prenant soin de tester à la fois les variables de **sessions** et le contenu du tableau **\$\_GET**).

## 4. LISTER LES UTILISATEURS PAR VILLE (TD)

Les questions suivantes vont permettre d'aboutir à ceci :



## 4.1. FORMULAIRE AFFICHEFORMULAIREUTILISATEURPARVILLE()

1. Créer une fonction afficheFormulaireUtilisateurParVille() dans formulaires.php proposant un formulaire dynamique (utilisant la méthode HTTP POST). Ce formulaire proposera à l'utilisateur une liste déroulante pour choisir, parmi les villes de résidence des utilisateurs enregistrés, une ville afin d'obtenir ensuite la liste des utilisateurs de cette ville. Le formulaire rappellera la même page. La balise <option> transmettra la valeur de son attribut value que l'on remplira avec le code INSEE plutôt que la ville (doublons de villes possibles). Au lieu d'afficher la ville dans la liste déroulante, nous afficherons le code postal + la ville pour faciliter l'utilisation.

## 4.2. Fonction ListeUtilisateurParVille(\$insee)

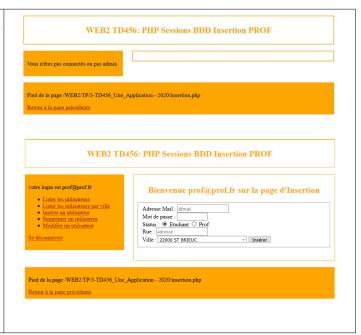
- Créer une fonction listeUtilisateurParVille(\$insee) dans fonctions.php qui retourne une tableau associatif contenant les informations de tous les Utilisateurs du code INSEE de la ville passée en paramètre.
- 2. Tester indépendamment de l'application ces fonctions dans un script testListeUtilisateurParVille.php.
- 3. Ajouter cette fonctionnalité dans l'application en complétant la zone «article».

#### (1) A finir et faire valider par l'enseignant de TP

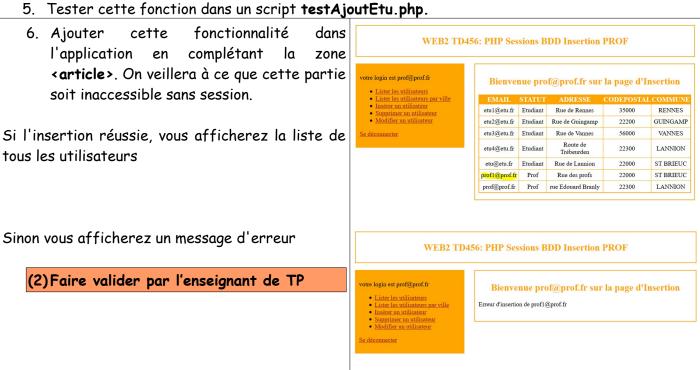
# INSERER DES UTILISATEURS (TP)

A partir du menu principal, on accède à une seconde page insertion.php.

- 1. Créer et protéger la page insertion.php à l'aide de variables de sessions, afficher le menu avant de mettre en place la fonctionnalité d'insertion. Si on n'est pas connecté et si on n'est pas admin, il faut rediriger l'utilisateur vers index.php après 2 secondes.
- 2. Tester cette protection en appelant la page en mode connecté et déconnecté.



- 3. Compléter la fonction afficheFormulaireAjoutUtilisateur() dans formulaires.php proposant un formulaire dynamique (utilisant la méthode HTTP POST) qui proposera à l'utilisateur une liste déroulante de toutes les villes afin de permettre l'enregistrement d'un nouvel utilisateur. Le formulaire rappellera la même page. On utilisera le même principe que pour la fonction afficheFormulaireUtilisateurParVille().
- 4. Créer une fonction ajoutUtilisateur(\$mail,\$pass,\$rue,\$insee,\$statut) dans fonctions.php afin de permettre l'insertion d'un utilisateur dans la table utilisateurs. Elle retournera 0 en cas d'échec ou 1 si l'insertion réussie.



## SUPPRIMER DES UTILISATEURS (TP)

A partir du menu principal, on accède à une troisième page modification.php. Cette page permettra soit de supprimer un membre, soit par la suite de modifier un membre.

- 1. Créer et protéger la page modification.php à l'aide de variables de sessions avant de mettre le place la fonctionnalité de suppression (même chose que précédemment avec la page insertion.php).
- 2. Tester cette protection en appelant la page en mode connecté et déconnecté.
- 3. Vous allez créer fonction une WEB2 TD456: PHP Sessions BDD Update PROF afficheFormulaireChoixUtilisateur(\$choix) formulaires.php proposant otre login est prof@prof.fr Bienvenue prof@prof.fr sur la page de Suppression formulaire dynamique (utilisant la méthode prof1@prof.fr Supprimer HTTP POST) qui proposera à l'utilisateur la liste des membres présents dans la bdd. Le formulaire rappellera la même page. Pied de la page /WEB2/TP/3-TD456\_Une\_Application - 2020/modification.php prendre \$choix pourra « Supprimer » ou « Modifier » (pour l'exercice suivant) et modifier la valeur du bouton en conséquence. WEB2 TD456: PHP Sessions BDD Update PROF 4. Tester cette fonction avec les deux valeurs Bienvenue prof@prof.fr sur la page de Modification etu1@etu Modifier

5. Créer une fonction supprimerUtilisateur(\$mail) dans fonctions.php afin de permettre la suppression d'un utilisateur dans la table utilisateurs. On ré-affichera la liste des Utilisateurs présent dans la BDD et un message indiquant la réussite ou non de l'opération.

(3) Faire valider par l'enseignant de TP

## 7. Modifier des Utilisateurs (TP)

A partir du menu principal, on accède à la troisième page modification.php avec le lien correspondant, vous devez afficher le formulaire afficheFormulaireChoixUtilisateur(\$choix) pour sélectionner le membre à modifier.



1. Créer la fonction afficheFormulaireModificationUtilisateur(\$mail) dans formulaires.php proposant un formulaire dynamique (utilisant la méthode HTTP POST) qui proposera à l'utilisateur les valeurs pré-remplies de l'utilisateur passé en paramètre. Le formulaire rappellera la même page. La valeur de l'adresse mail ne doit pas pouvoir être changée au niveau du formulaire.

#### WEB2 TD456: PHP Sessions BDD Update PROF

#### votre login est prof@prof.fr

- Lister les utilisateurs
- Lister les utilisateurs par ville
- Insérer un utilisateur
- Supprimer un utilisateur
- Modifier un utilisateur

Se déconnecter

Mot de passe	: ••••		
Statut : O Et	udiant   Prof		
Rue: Rue des	profs		
Ville : 22000	ST BRIEUC	∨ Valider	

2. Créer une fonction modifierUtilisateur(\$mail,\$pass,\$rue,\$insee,\$status) dans fonctions.php afin de permettre la modification d'un utilisateur dans la table utilisateurs. On ré-affichera la liste des Utilisateurs présent dans la BDD et un message indiquant la réussite ou non de l'opération.

#### WEB2 TD456: PHP Sessions BDD Update PROF

### votre login est prof@prof.fr

- <u>Lister les utilisateurs</u>
- <u>Lister les utilisateurs par ville</u>
- Insérer un utilisateur
- Supprimer un utilisateur
- Modifier un utilisateur

Se déconnecter

EMAIL	STATUT	ADRESSE	CODEPOSTAL	COMMUNI
etu1@etu.fr	Etudiant	Rue de Rennes	35000	RENNES
etu2@etu.fr	Etudiant	Rue de Guingamp	22200	GUINGAME
etu3@etu.fr	Etudiant	Rue de Vannes	56000	VANNES
etu4@etu.fr	Etudiant	Route de Trébeurden	22300	LANNION
etu@etu.fr	Etudiant	Rue de Lannion	22000	ST BRIEUC
prof1@prof.fr	Prof	Rue des profs	22600	ST BARNABE
prof@prof.fr	Prof	rue Edouard Branly	22300	LANNION

#### (4) Faire valider par l'enseignant de TP