

# Dialoguer avec les bases de données en PHP

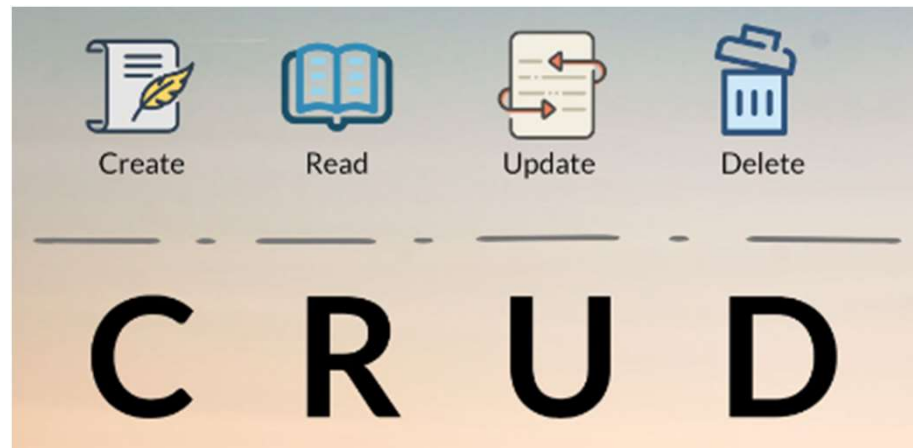
## CRUD APPLICATION MVC





## DÉFINITION DU CRUD

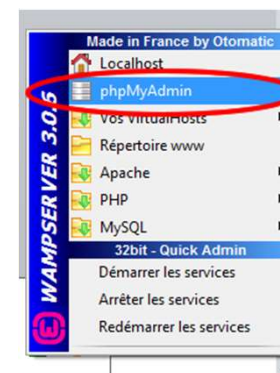
- CRUD est un acronyme pour : créer , lire , mettre à jour et supprimer (Create, Read, Update et Delete) .
- Les opérations CRUD sont des manipulations sur les données des tables d'une base de données





# LE SGBD MySQL

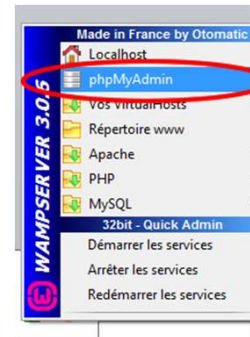
- Pour appliquer les opérations CRUD, PHP peut dialoguer avec plusieurs SGBD. Dans notre cours, nous allons choisir le SGBD MySQL pour les raisons suivantes:
  - Le SGBD MySQL qui est libre, gratuit et fiable
  - Le SGBD MySQL est inclus dans le logiciel WAMP qu'on a utilisé pour exécuter nos page PHP





# COMMENT ACCÉDER AU SGBD MySQL

- Exécuter votre logiciel Wamp puis cliquer sur l'option phpMyAdmin qui est le SGBD MySql



- Taper les coordonnées du super utilisateur sur cet écran, pour notre cas le login est root et pour le mot de passe: on met rien

phpMyAdmin

Bienvenue dans phpMyAdmin

Langue - Language

Français - French

Connexion

Utilisateur : root

Mot de passe :

Exécuter



# CRÉER LA BASE DE DONNÉES

- Une fois connecté, créer votre base de données

The screenshot shows the MySQL database management interface. At the top, there is a navigation bar with tabs: "Bases de données", "SQL", "État", "Comptes d'utilisateurs", and "Export". The "Bases de données" tab is selected and highlighted with a purple underline. Below this, the main heading "Bases de données" is displayed. Underneath, there is a section titled "Créer une base de données" with a green plus icon and a help icon. This section contains a text input field with the value "gest", a dropdown menu set to "Interclassement", and a "Créer" button. Below this form is a table listing existing databases. Three callout boxes with numbered instructions are overlaid on the image: 1. "1. Choisir cette option" points to the "Bases de données" tab. 2. "2. Donner un nom à votre base" points to the text input field containing "gest". 3. "3. valider la création" points to the "Créer" button.

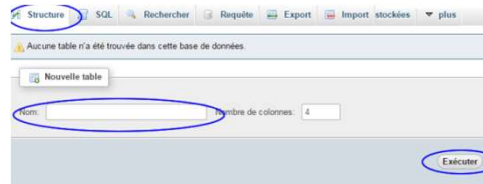
Base de données	Interclassement	Action
<input type="checkbox"/> information_schema	utf8_general_ci	<a href="#">Vérifier les privilèges</a>
<input type="checkbox"/> mysql	latin1_swedish_ci	<a href="#">Vérifier les privilèges</a>



# AJOUTER DES TABLES À LA BASE DE DONNÉES

- Après la création de la base, vous pouvez ajouter des tables avec plusieurs manières:

- Ajout manuel:



- Ajout avec Ecriture du script SQL



- Ajout avec importation du fichier de script résultat de conception de base de données ( méthode la plus recommandée)





## LES SCRIPTS ESSENTIELS POUR CONNECTER UNE APPLICATION PHP À UNE BASE DE DONNÉES ET GÉRER SES TABLES

- Pour connecter une application PHP à une base de données gérée par le SGBD MySQL, utiliser ce script:

```
$variableConnexion=new PDO('mysql:host=localhost;dbname=nomBase','root','');
```

- Pour manipuler ses différentes table, utiliser ce script:

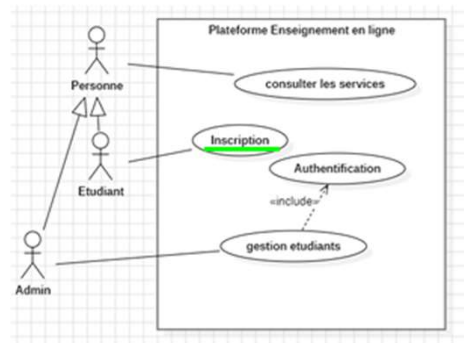
```
$nomrequete = $variableConnexion->prepare('commande SQL avec paramètres?(se  
répète n fois));
```

```
$nomrequete->execute(array($param1,..., $paramn)); // on remplace chaque ? Par un  
paramètre
```



# EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

- Implémenter les cas d'utilisation suivants en PHP en respectant le pattern MVC



## Explication

- Avec le cas d'utilisation « Inscription », l'internaute enregistre ses informations qui sont: matricule (entier qui s'écrit sur 4 chiffres) et un nom (commence par une lettre puis un ensemble de lettres et espaces) dans la table per crée dans la base de données gest gérée par le SGBD MySql. **Ecrire le script PHP nécessaire qui répond à ce cas d'utilisation tout en utilisant le pattern MVC**

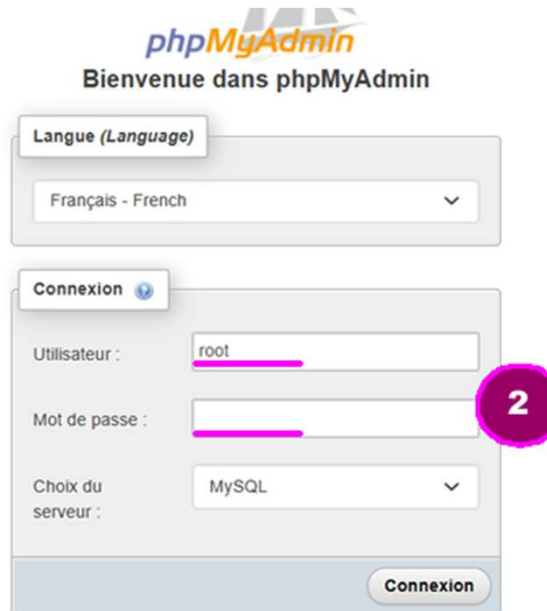
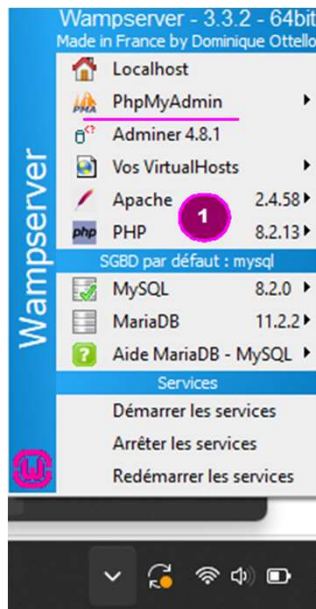




# EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

Implémentation de cas d'utilisation «Inscription »

- Accès au SGBD MySQL:





# EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

Implémentation de cas d'utilisation «Inscription »

- Création de la base de données gest et la table per:

Bases de données

The screenshot shows the MySQL Workbench interface. At the top, a window titled "Création d'une base de données" (Creation of a database) is open. It contains a text field with "gest", a dropdown menu showing "utf8mb4\_0900\_ai\_ci", and a "Créer" (Create) button. Below this, the "Structure" tab is selected, showing the SQL editor. The editor contains the following SQL code:

```
1 create table per(  
2     mat int,  
3     nom varchar(30),  
4     primary key(mat)  
5 );
```

Below the SQL editor, there is a "Délimiteur" (Delimiter) field set to ";" and a checkbox labeled "Afficher à nouveau la requête" (Show the query again). At the bottom, there is an "Exécuter" (Execute) button.

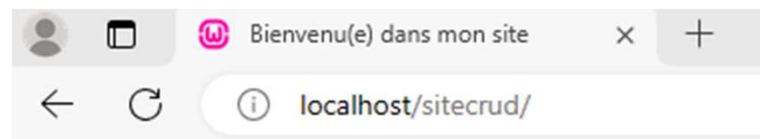


# EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

Implémentation de cas d'utilisation «Inscription »

- Page index.php → view : contact direct avec l'internaute:  
(Menu)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Bienvenu(e) dans mon site</title>
</head>
<body>
  <p><a href="inscr.php" target="_blank">Voulez vous s'inscrire!!!</a></p>
  <p><a href="aut.php" target="_blank">Voulez vous gérer les données d'un étudiant!!!</a></p>
</body>
</html>
```



[Voulez vous s'inscrire!!!](#)

[Voulez vous gérer les données d'un étudiant!!!](#)

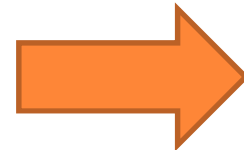


# EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

Implémentation de cas d'utilisation «Inscription »

- Page inscr.php → view : contact direct avec l'internaute (formulaire)

```
inscr.php > html > body > form > p
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Inscrivez vous</title>
7 </head>
8 <body>
9   <form action="traitins.php" method="post">
10     <p>
11       <label for="t1">Introduire votre matricule</label>
12       <input type="text" name="mat" id="t1">
13     </p>
14     <p>
15       <label for="t2">Introduire votre nom</label>
16       <input type="text" name="nm" id="t2">
17     </p>
18     <p>
19       <input type="submit" value="OK">
20       <input type="reset" value="NO">
21     </p>
22   </form>
23 </body>
24 </html>
```



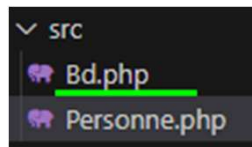
The browser window shows the page title "Inscrivez vous" and the URL "localhost/sitecrud/inscr.php". The form contains two text input fields: "Introduire votre matricule" and "Introduire votre nom". Below the inputs are two buttons: "OK" and "NO".



# EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

Implémentation de cas d'utilisation «Inscription »

- Page BD.php ➔ Model: une classe qui permet à l'application PHP la connexion à la base de données gest gérée par le SGBD MySQL



```
> Bd.php > Bd > connexion
<?php

class Bd {
    private static $con=null;
    public static function connexion() {
        try{
            self::$con=new PDO('mysql:host=localhost;dbname=gest','root','');
            return self::$con;
        }
        catch(PDOException $e){
            echo 'Erreur ' . $e->getMessage();
            die();
        }
    }
}

?>
```

Nom de la base

coordonnées d'accès

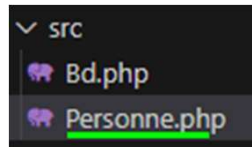
une classe qui existe sous PHP qui permet la connexion aux SGBDs



# EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

Implémentation de cas d'utilisation «Inscription »

- Page Personne.php → Model: une classe qui permet de vérifier la saisie et insérer les données dans la table per de la base gest



```
<?php
class Personne{
    private $mat;
    private $nom;

    /**
     *
     */
    public function __construct($mat,$nom)
    {
        $this->mat=intval($mat);
        $this->nom=$nom;
    }

    /**
     *
     */
    public static function verif($mat,$nom){
        $patmat='/^[0-9]{4}$/';
        $patNom='/^[a-z][a-z]*$/i';
        if(preg_match($patmat,$mat) && preg_match($patNom,$nom))return true;
        else return false;
    }

    /**
     *
     */

    public function inserer($con){
        $req=$con->prepare('insert into per(mat,nom) values (?,?)');
        $req->execute(array($this->mat, $this->nom));
        $con=null;
    }
}
?>
```

Instanciation de l'objet

Vérification de la saisie

Insertion des données dans la table per



## EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

Implémentation de cas d'utilisation «Inscription »

- Page traitins.php ➔ Controller qui reçoit les informations de views et désigne les models qui vont faire le traitement et envoie le résultat vers d'autres views pour informer l'utilisateur
- Voir l'exemple:



# EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

Implémentation de cas d'utilisation «Inscription »

- Page traitins.php:

```
1 <?php
2 spl_autoload_register(function($name){
3     require_once('src/'.$name.'.php');});
4 >?
5
6 <!DOCTYPE html>
7 <html lang="en">
8 <head>
9     <meta charset="UTF-8">
10    <meta name="viewport" content="width=device-width, initial-scale=1.0">
11    <title>Document</title>
12 </head>
13 <body>
14 <?php
15 if($_SERVER['REQUEST_METHOD'] === 'GET'){
16     echo 'SVP, passer par le formulaire!!!';
17     include('inscr.php');
18 }
19 else{
20     $mat=$_POST['mat'];
21     $nom=$_POST['nm'];
22     if(Personne::verif($mat,$nom)){
23         $con=bd::connexion();
24         $a=new Personne($mat,$nom);
25         $a->inserer($con);
26         $con=null;
27         echo 'insertion réussie!!!';
28         include('index.php');
29     }
30     else{
31         echo 'SVP, Saisie convenablement les info!!!';
32         include('inscr.php');
33     }
34 }
35
36 >?
37
38 </body>
39 </html>
```

Importation des classes

Interdire l'accès direct à la page

Récupération des données du formulaire

Vérification de la validité de saisie

Insertion des données dans la table de la base

Traitement dans le cas où la saisie est invalide

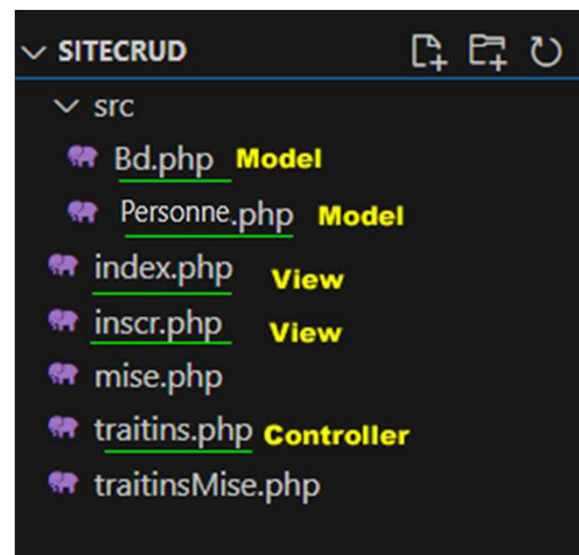




# EXEMPLE 1: APPLICATION CRUD PHP (INSERTION)

Résumons:

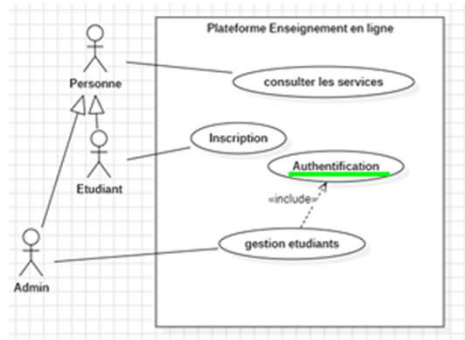
- Voici les caractéristiques des pages de l'application qui représente le cas utilisation « Inscription » tout en respectant le modèle MVC:
- :





## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Implémentation de cas d'utilisation « Authentification », en respectant MVC (Model/ View/ Controller)



### Explication

- Avec le cas d'utilisation « Authentification », l'administrateur doit s'authentifier: fournir un login et un mot de passe qui existent dans la table aut crée dans la base de données gest gérée par le SGBD MySql. **Ecrire le script PHP nécessaire qui répond à ce cas d'utilisation tout en utilisant le pattern MVC**



## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Implémentation de cas d'utilisation «Authentification»:

- D'abord, on va créer une table nommée aut dans la base gest:

Exécuter une ou des requêtes SQL sur la base de données « gest »:

```
1 create table aut(  
2   login varchar(30),  
3   motpasse varchar(30),  
4   primary key(login));
```

Délimiteur :  ☐ Afficher à nouveau la requête après exécution

**Exécuter**

- Puis on va insérer des tuples dans cette tables:

Parcourir Structure SQL Rechercher **Insérer** Exporter Importer Privileges

Colonne	Type	Fonction	Null	Valeur
login	varchar(30)	<input type="text"/>	<input type="text"/>	essai
motpasse	varchar(30)	<input type="text"/>	<input type="checkbox"/>	123

**Exécuter**



login	motpasse
essai	123
stat	a1

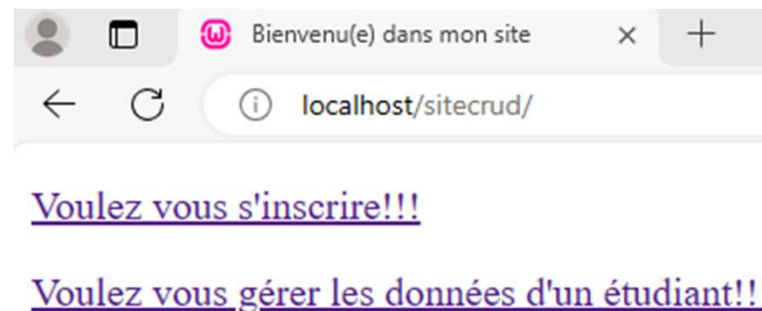


## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Implémentation de cas d'utilisation «Authentification»:

- Page index.php → view : contact direct avec l'internaute:  
(Menu)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Bienvenu(e) dans mon site</title>
</head>
<body>
  <p><a href="inscr.php" target="_blank">Voulez vous s'inscrire!!!</a></p>
  <p><a href="aut.php" target="_blank">Voulez vous gérer les données d'un étudiant!!!</a></p>
</body>
</html>
```





## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Implémentation de cas d'utilisation «Authentification»:

- Page aut.php → view : contact direct avec l'internaute:  
(formulaire)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <form action="traitcons.php" method="post">
    <p>
      <label for="t1">Introduire votre login</label>
      <input type="text" name="lg" id="t1">
    </p>
    <p>
      <label for="t2">Introduire votre mot de passe</label>
      <input type="password" name="mp" id="t2">
    </p>
    <p>
      <input type="submit" value="OK">
      <input type="reset" value="NO">
    </p>
  </form>
</body>
</html>
```

Bienvenu(e) dans mon site

localhost/sitecrud/aut.php

Introduire votre login

Introduire votre mot de passe

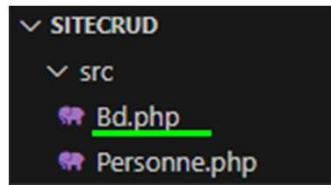
OK NO



## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Implémentation de cas d'utilisation «Authentification»:

- Page Bd.php → Model: une classe qui permet à l'application PHP la connexion à la base de données gest gérée par le SGBD MySQL



```
Bd.php > Bd > connexion
<?php

class Bd {
    private static $con=null;
    public static function connexion() {
        try{
            self::$con=new PDO('mysql:host=localhost;dbname=gest','root','');
            return self::$con;
        }
        catch(PDOException $e){
            echo 'Erreur ' . $e->getMessage();
            die();
        }
    }
}
```

Nom de la base

coordonnées d'accès

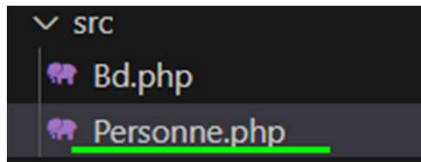
une classe qui existe sous PHP qui permet la connexion aux SGBDs



## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Implémentation de cas d'utilisation «Authentification»:

- `Personne.php` → Model: une classe qui permet de vérifier le login et le mot de l'administrateur



```
class Personne{
    /**
     *
     */
    public function __construct($mat,$nom)
    {
        $this->mat=intval($mat);
        $this->nom=$nom;
    }

    /**
     *
     */
    public static function verif($mat,$nom){
        $patmat='/^[0-9]{4}$/' ;
        $patNom='/^[a-z][a-z]*$/i';
        if(preg_match($patmat,$mat) && preg_match($patNom,$nom))return true;
        else return false;
    }

    public static function verif1($login,$mpasse){

        if($login===' ' || $mpasse===' '){return false;
        else return true;
        }

    }

    /**
     *
     */
    public static function consulter($con,$login, $mpasse) {
        $req = $con->prepare('select * from aut where login=? and motpasse=?');
        $req->execute(array($login, $mpasse));
        $res = $req->fetch(); // Corrected typo from 'fecth()' to 'fetch()'
        $con = null; // Moved closing connection outside the if-else block
        if (empty($res)) {
            return false;
        } else {
            return true;
        }
    }
}
```

Methode qui permet de vérifier la saisie

Methode qui vérifie le login et mot de passe qui existent dans la table aut

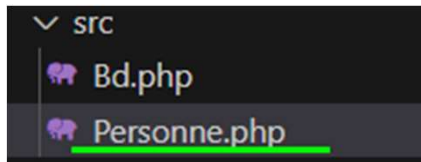




## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Implémentation de cas d'utilisation «Authentification»:

- `Personne.php` → Model: une classe qui permet de vérifier le login et le mot de l'administrateur



```
class Personne{
    /**
     *
     */
    public function __construct($mat,$nom)
    {
        $this->mat=intval($mat);
        $this->nom=$nom;
    }

    /**
     *
     */
    public static function verif($mat,$nom){
        $patmat='/^[0-9]{4}$';
        $patNom='/^[a-z][a-z]*$/i';
        if(preg_match($patmat,$mat) && preg_match($patNom,$nom))return true;
        else return false;
    }

    public static function verif1($login,$mpasse){

        if($login===' ' || $mpasse===' '){return false;
        else return true;
        }

    }

    /**
     *
     */
    public static function consulter($con,$login, $mpasse) {
        $req = $con->prepare('select * from aut where login=? and motpasse=?');
        $req->execute(array($login, $mpasse));
        $res = $req->fetch(); // Corrected typo from 'fecth()' to 'fetch()'
        $con = null; // Moved closing connection outside the if-else block
        if (empty($res)) {
            return false;
        } else {
            return true;
        }
    }
}
```

Methode qui permet de vérifier la saisie

Methode qui vérifie le login et mot de passe qui existent dans la table aut





## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Implémentation de cas d'utilisation «Authentification»:

- `traitcon.php` → Controller: une page qui reçoit les informations de views et désigne les models qui vont faire le traitement et envoie le résultat vers des autres views pour informer l'utilisateur
- Voir l'exemple:



## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Implémentation de cas d'utilisation «Authentification»:

- traitcon.php:

```
traitcon.php > html > body
1  <?php
2  spl_autoload_register(function($name){
3      require_once('src/'.$name.'.php');});
4  ?>
5
6  <!DOCTYPE html>
7  <html lang="en">
8  <head>
9      <meta charset="UTF-8">
10     <meta name="viewport" content="width=device-width, initial-scale=1.0">
11     <title>Document</title>
12 </head>
13 <body>
14     <?php
15     if($_SERVER['REQUEST_METHOD'] === 'GET'){
16         echo 'SVP, passer par le formulaire!!!';
17         include('aut.php');
18     }
19     else{
20         $lg=$_POST['lg'];
21         $mp=$_POST['mp'];
22         if(Personne::verifl($lg,$mp)){
23             $con=Bd::connexion();
24             if(Personne::consulter($con,$lg,$mp)){
25                 echo 'Vous êtes connectées ';
26                 include('mise.php');
27             }
28         }
29         else{
30             echo 'SVP, login/ mot de passe erronés !!!';
31             include('aut.php');
32         }
33     }
34     else{
35         echo 'SVP, Saisie convenablement les info!!!';
36         include('aut.php');
37     }
38 }
39 ?>
40 </body>
```

Importation des classes

Interdire l'accès direct à la page

Récupération des données saisies

si la saisie de login et mot passe est non vide, on se connecte à la base et on cherche leur existence dans la table aut

traitement dans le cas d'existence de login et mot passe

traitement dans le cas de non existence de login et mot passe

traitement dans le cas où la saisie est invalide



## EXEMPLE 2: APPLICATION CRUD PHP (CONSULTATION)

Résumons:

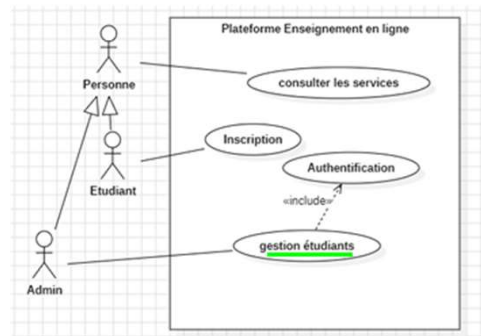
- Voici les caractéristiques des pages de l'application qui représente le cas utilisation « Authentification » tout en respectant le modèle MVC:

src	
<u>Bd.php</u>	Model
<u>Personne.php</u>	Model
<u>aut.php</u>	view
<u>index.php</u>	view
<u>inscr.php</u>	
<u>mise.php</u>	view
<u>traitcons.php</u>	Controller
<u>traitins.php</u>	
<u>traitinsMise.php</u>	



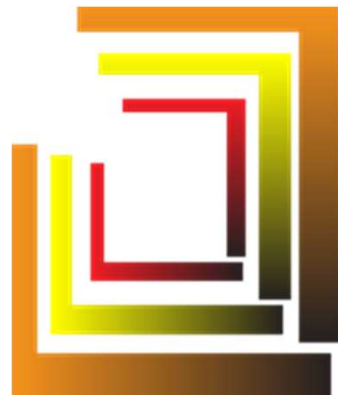
## EXEMPLE 3: APPLICATION CRUD PHP (MISE À JOUR)

- Implémenter le cas d'utilisation « gestion étudiants » en PHP tout en respectant le modèle MVC:



### Explication :

- Après avoir fait l'authentification, l'administrateur peut modifier le nom (doit commencer par une lettre puis un ensemble de lettres et espaces) de l'étudiant à partir de sa matricule (entier de 4 chiffres). La modification se fait dans la table qui existe dans la base de données gest. **Ecrire le script PHP nécessaire qui répond à ce cas d'utilisation tout en respectant le pattern MVC**



***FIN***  
***Mansour Sihem***