

- Doc Séminaire
 - Cas d'utilisation
 - Inscription
 - Connexion
 - Voir le Programme
 - Voir les inscrits(admin)
 - Voir les inscrits a un événement choisi (admin)
 - Fonction Accès Données
 - Fonctions Principales
 - chargeJSONseminaire()
 - donnerIntituleSeminare()
 - donnerLesConferences(\$heure)
 - sauverDonneesInscription(\$nom, \$prenom, \$mail, \$ville, \$profession)
 - enregistre(\$lesChoix)
 - donnerToutesLesConferences()
 - Fonction Gestion
 - Fonctions principales
 - verifier(\$login, \$mdp)
 - estAdmin()
 - verifierDonneesInscription(\$nom, \$prenom, \$mail, \$ville)
 - ajouterErreur(\$msg)
 - donnerNbErreurs()
 - afficherErreurs()

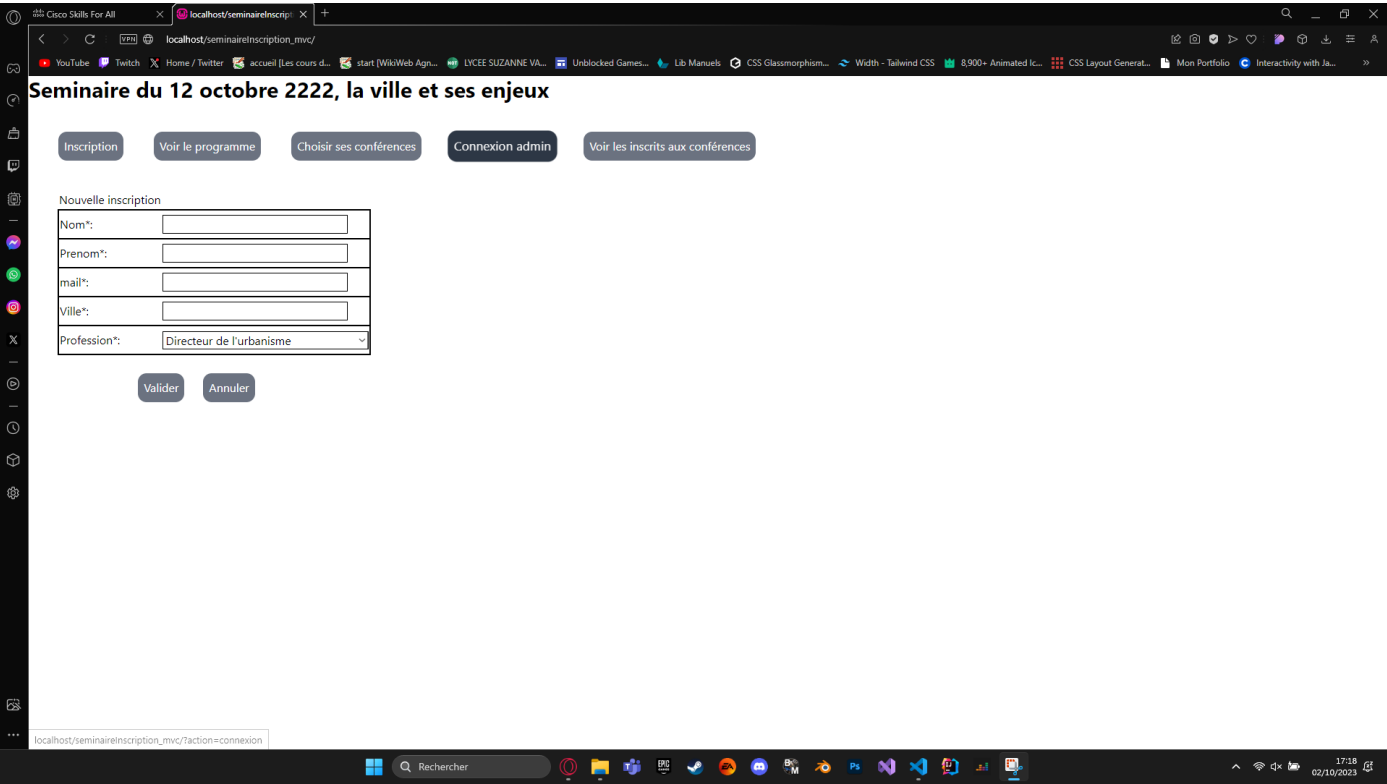
Doc Séminaire

Séminaire est une application web qui permet de gérer des séminaires avec des inscrits (inscription, consultations, etc.).

ATTENTION ! Cette application fonctionne avec des fichiers en .json pour la partie base de données.

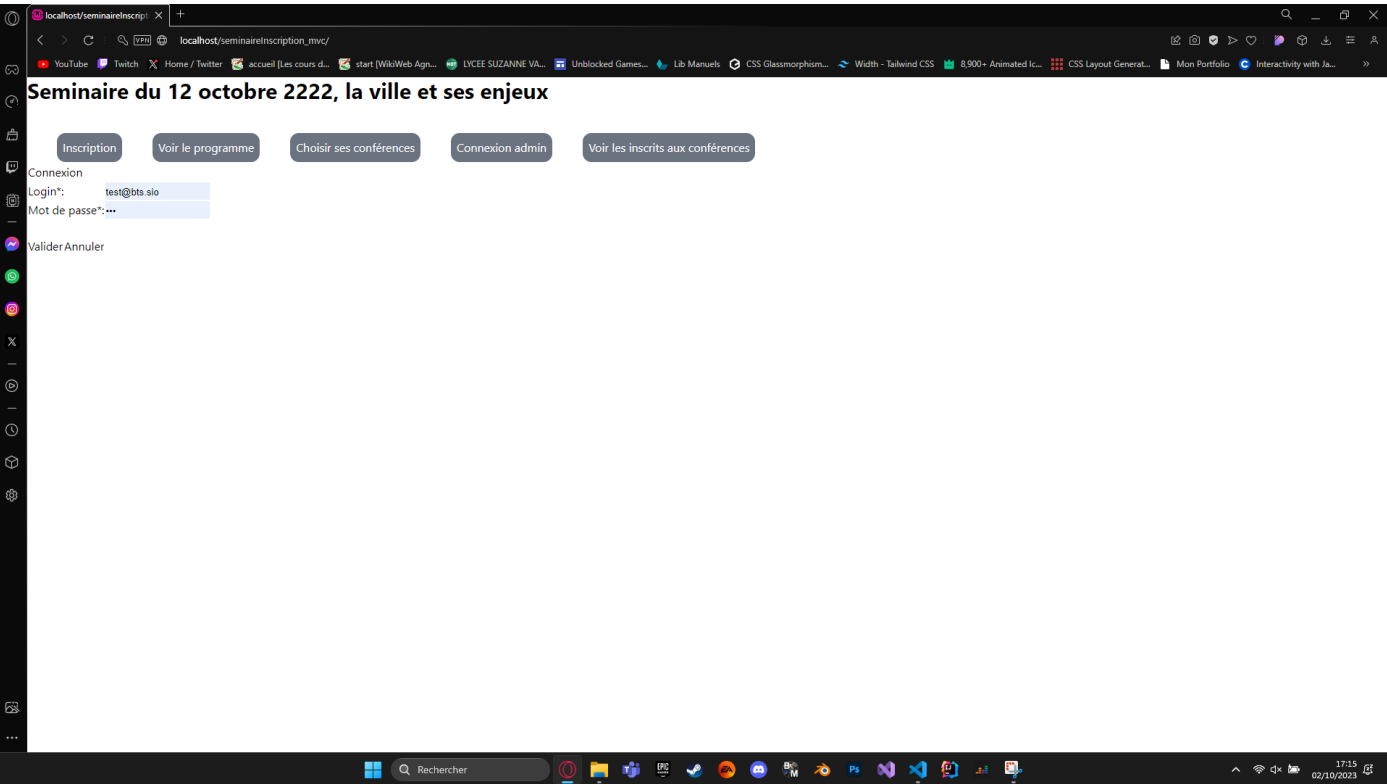
Cas d'utilisation

Inscription



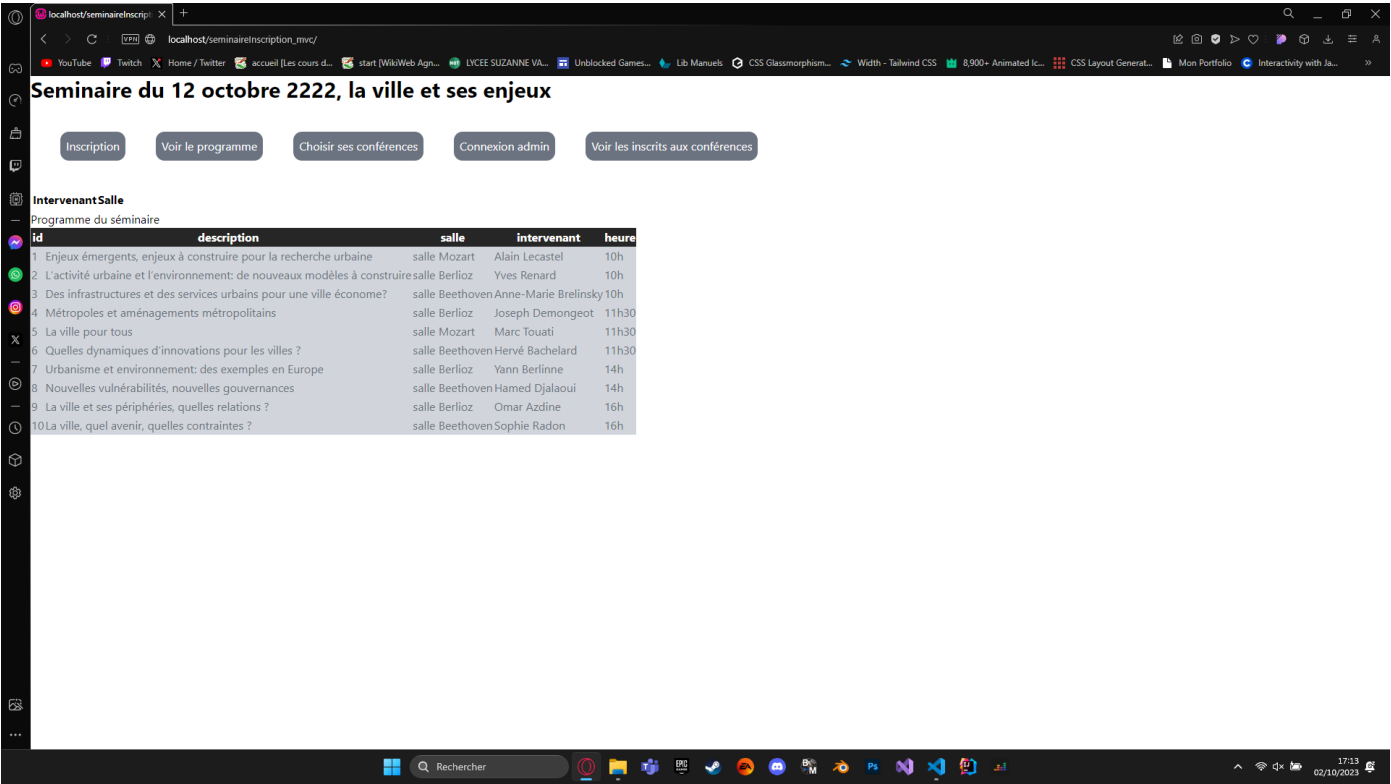
L'inscription permet a l'utilisateur de s'inscrire pour pouvoir acceder au programme et s'inscrire aux conférences de son choix.

Connexion



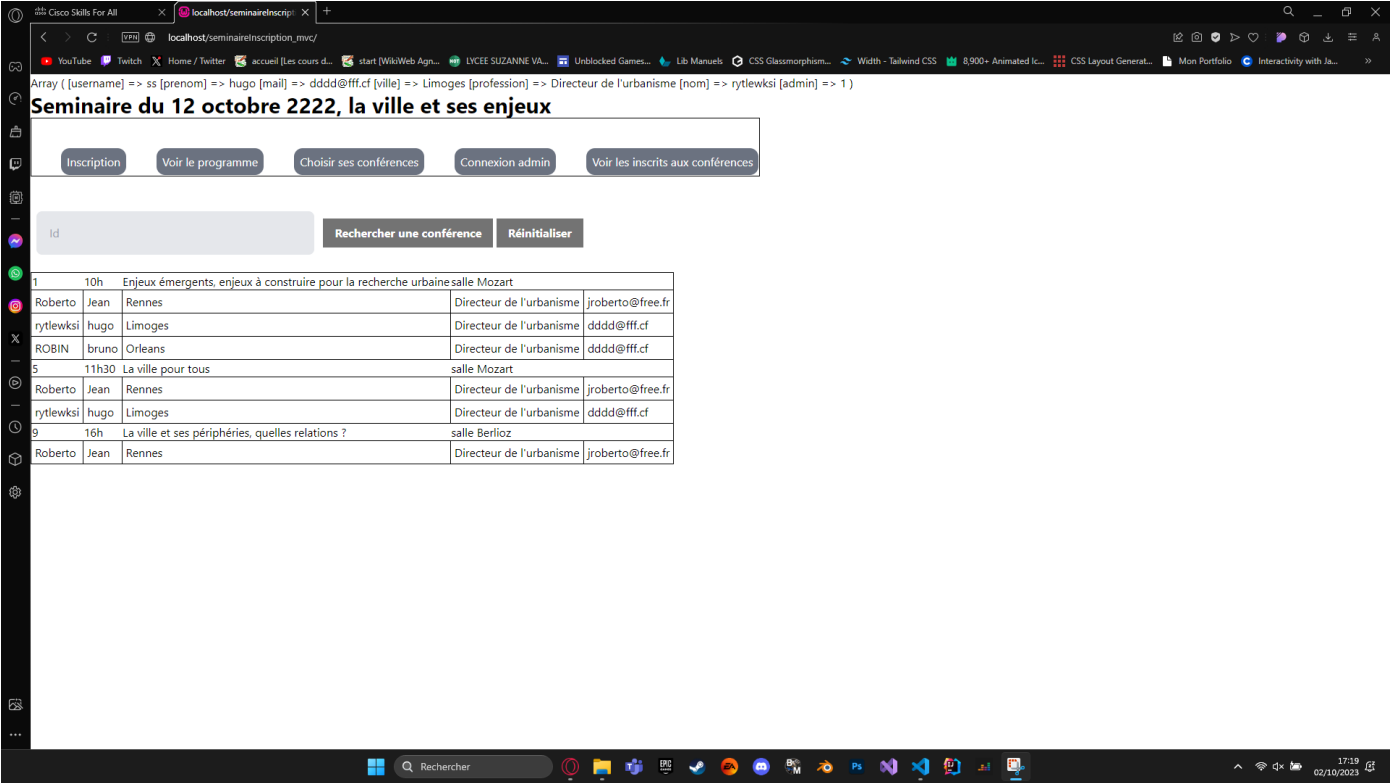
La connexion permet a l'utilisateur de se connecter pour acceder a la partie admin.

Voir le Programme



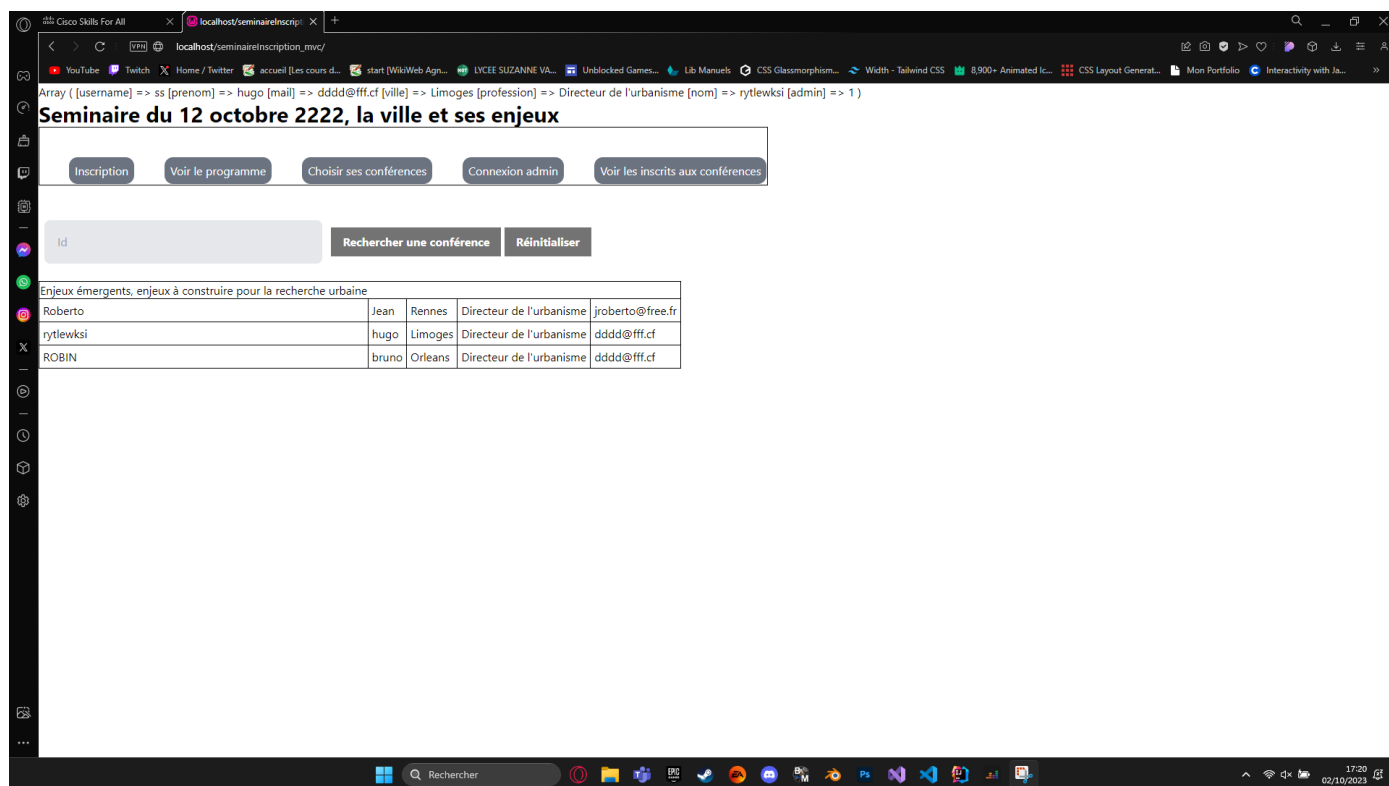
La partie programme permet a l'utilisateur de voir le programme du séminaire et de s'inscrire aux conférences de son choix.

Voir les inscrits(admin)



La partie inscrit permet a l'admin de voir les inscrits au séminaire et de voir les conférences auxquelles ils sont inscrits.

Voir les inscrits a un événement choisi (admin)



La partie voir inscrit permet a l'admin de voir les inscrits a un événement choisi.

Fonction Accès Données

Fonctions Principales

`chargeJSONseminaire()`

```
function donnerIntituleSeminaire()  
{  
  $json_source = file_get_contents('data/seminaire.json');  
  $document = json_decode($json_source);  
  return $document->seminaire->intitule;  
}
```

Cette fonction permet de charger les données du séminaire à partir d'un fichier JSON.

`donnerIntituleSeminaire()`

```
function donnerIntituleSeminaire()
{
$json_source = file_get_contents('data/seminaire.json');
$document = json_decode($json_source);
return $document->seminaire->intitule;
}
```

Cette fonction retourne l'intitulé du séminaire à partir des données chargées.

donnerLesConferences(\$heure)

```
function donnerLesConferences($heure)
{
    $json_source = file_get_contents('data/seminaire.json');
    $document = json_decode($json_source);
    return $document->seminaire->creneau[$heure]->conference;
}
```

Cette fonction retourne toutes les conférences commençant à l'heure donnée sous forme d'un tableau.

sauverDonneesInscription(\$nom, \$prenom, \$mail, \$ville, \$profession)

```
function sauverDonneesInscription($nom, $prenom,$mail,$ville, $profession){
    $_SESSION['nom'] = $nom;
    $_SESSION['prenom'] = $prenom;
    $_SESSION['mail'] = $mail;
    $_SESSION['ville'] = $ville;
    $_SESSION['profession'] = $profession;
}
```

enregistre(\$lesChoix)

```
function enregistre($lesChoix) {
    $tab = [
        "nom" => $_SESSION['nom'],
        "prenom" => $_SESSION['prenom'],
        "mail" => $_SESSION['mail'],
        "ville" => $_SESSION['ville'],
        "profession" => $_SESSION['profession'],
    ];
    $jsonTab = json_encode($tab);
}
```

```

$json = file_get_contents("data/seminaire.json");
$parsed_json = json_decode($json, true);
foreach ($parsed_json['seminaire']['creneau'] as &$creneau) {
    $meschoix = $lesChoix[$creneau["heure"]];
    foreach ($creneau['conference'] as &$laconference) {
        if ($laconference['id'] == $meschoix) {
            $laconference['participants'][] = json_decode($jsonTab, true);
            print_r("je passe ");
        } else {
            print_r("je ne passe pas");
        }
    }
}
$updated_json = json_encode($parsed_json, JSON_PRETTY_PRINT);
file_put_contents("data/seminaire.json", $updated_json);
}

```

Attention j'ai utilisé : & pour pouvoir faire passer une chaîne

Cette fonction enregistre un participant et ses choix de conférences dans les données du séminaire.

donnerToutesLesConferences()

```

function donnerToutesLesConferences(){
    $json_source = file_get_contents('data/seminaire.json');
    $document = json_decode($json_source);
    return $document->seminaire->creneau;
}

```

Cette fonction retourne toutes les conférences sous forme d'un tableau.

Fonction Gestion

Fonctions principales

verifier(\$login, \$mdp)

```

function verifier($login,$mdp){
    $admin =chargeJSONadmin();
    foreach($admin as $un_admin){
        if($un_admin->login == $login && $un_admin->mdp == $mdp){

```

```

        $_SESSION['admin']=true;
        $message = true;
    }else{
        $message = false;
        unset($_SESSION['admin']);
    }
}
if($message){
    ajouterErreur("Vous êtes Admin");
}else{
    ajouterErreur("Login ou mot de passe incorrect");
}
}

```

Cette fonction vérifie le login et le mot de passe fournis en paramètres en comparant avec les informations stockées dans un tableau JSON d'administrateurs. Si les informations correspondent, elle définit une variable de session `$_SESSION['admin']` à `true`. Sinon, elle la supprime. Ensuite, elle ajoute un message d'erreur approprié à l'aide de la fonction `ajouterErreur`.

estAdmin()

```

function estAdmin(){
    if($_SESSION['admin']){
        return true;
    }
    else{
        return false;
    }
}

```

Cette fonction vérifie si l'utilisateur actuel est un administrateur en vérifiant la variable de session `$_SESSION['admin']`. Si elle est définie à `true`, la fonction renvoie `true`, sinon, elle renvoie `false`

verifierDonneesInscription(\$nom, \$prenom, \$mail, \$ville)

```

function verifierDonneesInscription($nom, $prenom, $mail, $ville)
{
    if ($nom=="" || $prenom=="" || $mail=="" || $ville=="")
    {
        ajouterErreur("Chaque champ suivi du caractère * est obligatoire");
    }
    if (!preg_match("/^[a-z0-9\._]+@[a-z0-9\._]+\.[a-z]{2,4}$/i", $mail))

```

```
        ajouterErreur("Le format de l'email n'est pas valide");
    }
}
```

Cette fonction vérifie les données saisies lors de l'inscription. Elle vérifie si les champs *nom*, *prenom*, *mail*, *etville* ne sont pas vides et si l'adresse email est au format valide. Si une vérification échoue, elle ajoute un message d'erreur à l'aide de `ajouterErreur`

ajouterErreur(\$msg)

```
function ajouterErreur($msg)
{
    if (! isset($_GET['erreurs']))
        $_GET['erreurs']=array();
    $_GET['erreurs'][]=$msg;
}
```

Cette fonction ajoute un message d'erreur.

donnerNbErreurs()

```
function donnerNbErreurs()
{
    if (!isset($_GET['erreurs']))
        return 0;
    else
        return count($_GET['erreurs']);
}
```

Cette fonction renvoie le nombre d'erreurs stockées

afficherErreurs()

```
function afficherErreurs()
{
    echo '<div id="erreur">';
    if (isset($_GET['erreurs'])) {
        $erreurs=$_GET['erreurs'];
        for ($i=0;$i<count($erreurs);$i++){
            echo $erreurs[$i]. '<br />';
        }
    }
    echo '</div>';
}
```


Cette fonction affiche toutes les erreurs stockées