

Documentation Technique

Introduction :

Sommaire :

Le but du projet est de faire une application web pour l'inscription et la gestion d'un séminaire. Le projet est élaboré suivant l'architecture MVC (Modèle Vue Contrôleur) pour garantir une séparation claire des responsabilités entre la logique, la gestion des données et l'affichages

Architecture du projet :

- contrôleur : gestion des fonctions
- datas : il y a tous les fichiers json
- fonctions : toutes les fonctions du projet
- vue : affichage sur le web

Partie 1 : Gestion des Inscriptions

Etape 1 : Programme du Séminaire

10h	Intervenant	Salle
Enjeux émergents, enjeux à construire pour la recherche urbaine	Alain Lecastel	salle Mozart
L'activité urbaine et l'environnement: de nouveaux modèles à construire	Yves Renard	salle Berlioz
Des infrastructures et des services urbains pour une ville économe?	Anne-Marie Brelinsky	salle Beethoven
11h30	Intervenant	Salle
Métropoles et aménagements métropolitains	Joseph Demongeot	salle Berlioz
La ville pour tous	Marc Touati	salle Mozart
Quelles dynamiques d'innovations pour les villes ?	Hervé Bachelard	salle Beethoven
14h	Intervenant	Salle
Urbanisme et environnement: des exemples en Europe	Yann Berlinne	salle Berlioz
Nouvelles vulnérabilités, nouvelles gouvernances	Hamed Djalaoui	salle Beethoven
16h	Intervenant	Salle
La ville et ses périphéries, quelles relations ?	Omar Azdine	salle Berlioz
La ville, quel avenir, quelles contraintes ?	Sophie Radon	salle Beethoven

function donnerLesHeuresCreneaux() retourne un tableau contenant tous les créneaux horaires disponibles (heures de début).

function donnerLesConferences(\$heure) retourne toutes les conférences programmées pour l'heure donnée. Les informations incluent l'identifiant, la description, l'intervenant, la salle et le nombre de places disponibles.

```
$heures = donnerLesHeuresCreneaux();
$conferencesParHeure = array();
foreach ($heures as $heure) {
    $conferencesParHeure[$heure] = donnerLesConferences($heure);
}
```

Dans le contrôleur programme.php on appelle la fonction **donnerLesHeuresCrenaux()** et **donnerLesConferences()** en fonction de l'heure.

Dans le contrôleur programme.php, les fonctions **donnerLesHeuresCrenaux()** et **donnerLesConferences()** sont appelées en fonction de l'heure sélectionnée. Le résultat est ensuite affiché dans la vue, où un **foreach** permet de lister toutes les conférences avec leurs intervenants et la salle correspondante.

Puis dans la vue on fait un **foreach** qui permet que pour chaque heure, on obtient toutes les conférences avec le nom, l'intervenant et la salle.

Base pour JSON

```
function chargeJSONseminaire()
```

```
{  
    $json_source = file_get_contents('data/seminaire.json');  
    $document = json_decode($json_source);  
    return $document;  
}
```

```
function sauveJSONseminaire($doc){
```

```
    $json_data = json_encode($doc, JSON_PRETTY_PRINT);  
    file_put_contents('data/seminaire.json', $json_data);  
}
```

Etape 2 : Inscription des visiteurs

Nouvelle inscription	
Nom*:	<input type="text"/>
Prenom*:	<input type="text"/>
mail*:	<input type="text"/>
Ville*:	<input type="text"/>
Profession*:	<div>Directeur de l'urbanisme ▼</div>

Valider

Annuler

function sauverDonneesInscription(\$nom, \$prenom,\$mail,\$ville, \$profession) enregistre les informations d'inscription du visiteur dans une variable de session. Chaque paramètre représente respectivement : le nom, prénom, email, ville et profession du visiteur.

function donnerLesProfessions() on utilise la fonction **chargeJSONprofessions()** qui contient toutes les professions puis dans la vue on fait une boucle avec un select pour afficher chaque professions.

Dans le contrôleur inscription on appelle la fonction **donnerLesProfessions()** puis on utilise un switch pour le bouton si on clique sur annuler ça ne fait rien et sur valider ça vérifie les données de l'inscription puis ça les sauvegarde

Etape 3 : Choix des conférences

10h		
Enjeux émergents, enjeux à construire pour la recherche urbaine	salle Mozart	<input type="radio"/>
L'activité urbaine et l'environnement: de nouveaux modèles à construire	salle Berlioz	<input type="radio"/>
Des infrastructures et des services urbains pour une ville économe?	salle Beethoven	<input type="radio"/>
11h30		
Métropoles et aménagements métropolitains	salle Berlioz	<input type="radio"/>
La ville pour tous	salle Mozart	<input type="radio"/>
Quelles dynamiques d'innovations pour les villes ?	salle Beethoven	<input type="radio"/>
14h		
Urbanisme et environnement: des exemples en Europe	salle Berlioz	<input type="radio"/>
Nouvelles vulnérabilités, nouvelles gouvernances	salle Beethoven	<input type="radio"/>
16h		
La ville et ses périphéries, quelles relations ?	salle Berlioz	<input type="radio"/>
La ville, quel avenir, quelles contraintes ?	salle Beethoven	<input type="radio"/>

function estInscrit() vérifie l'existence d'une variable de session pour déterminer si le visiteur a déjà rempli le formulaire d'inscription. Retourne un booléen (`true` ou `false`).

function enregistre(\$lesChoix) qui enregistre un participant et ses choix de conférences dans le fichier `seminaire.json`.

Résumé Partie 1 :

1. Affichage des créneaux horaires disponibles.
2. Inscription des visiteurs via la sauvegarde des données dans une session.
3. Enregistrement des choix de conférences dans le fichier `seminaire.json`.

Partie 2 : Exploitation des inscriptions

Etape 1 : Connexion Admin

Connexion	
Login*:	<input type="text"/>
Mot de passe*:	<input type="password"/>

<input type="button" value="Valider"/>	<input type="button" value="Annuler"/>
--	--

function chargeJSONadmin() :

Cette fonction charge le contenu du fichier admin.json et le transforme en un objet JSON. Ce fichier contient les informations de connexion des administrateurs, telles que leur identifiant et mot de passe.

function verifier(\$login,\$mdp) :

Cette fonction vérifie la validité des informations de connexion. Elle compare le login et le mot de passe fournis par l'utilisateur avec ceux présents dans le fichier admin.json. Si les informations correspondent, l'accès est autorisé.

function estAdmin() Vérifie si l'utilisateur est un administrateur connecté. Elle retourne true si l'administrateur a validé ses informations de connexion et est autorisé à visualiser les inscriptions.

Le contrôleur vérifie si le bouton **Valider** a été cliqué.

Si l'administrateur a saisi les bonnes informations et cliqué sur **Valider**, il est redirigé vers la page de la liste des inscrits.

Si les informations ne sont pas valides ou si le bouton n'a pas été cliqué, l'administrateur reste sur la page de connexion pour essayer à nouveau.

Etape 2 : Visualisation des Inscriptions

1 - 10h - Enjeux émergents, enjeux à construire pour la recherche urbaine				
Nom	Prénom	Ville	Profession	Email
Roberto	Jean	Rennes	Directeur de l'urbanisme	jroberto@free.fr
Jenny	Pierre	Aix	Technicien	pierre@mail.com
Segalat	Adrien	Verneuilfrère	Chef	Adrien@mail.com
Ballereau	Mael	Chateauroux	Chargé	mael@mail.com
2 - 10h - L'activité urbaine et l'environnement: de nouveaux modèles à construire				
Aucun participant inscrit pour cette conférence.				
3 - 10h - Des infrastructures et des services urbains pour une ville économe?				
Aucun participant inscrit pour cette conférence.				
4 - 11h30 - Métropoles et aménagements métropolitains				
Nom	Prénom	Ville	Profession	Email
Segalat	Adrien	Verneuilfrère	Chef	Adrien@mail.com
5 - 11h30 - La ville pour tous				

function donnerToutesLesConferences() Cette fonction retourne un tableau contenant toutes les conférences disponibles. Chaque conférence inclut des informations telles que l'identifiant, le nom, le lieu et le nombre de participants inscrits.

function donnerParticipants(\$numConference) Cette fonction retourne la liste de tous les participants inscrits à une conférence spécifique, identifiée par son numéro (numConference). Les informations des participants incluent leur nom, prénom, ville, profession, et éventuellement d'autres détails pertinents.

Le contrôleur vérifie d'abord si l'utilisateur connecté est un administrateur à l'aide de la fonction **estAdmin()**.

Si l'utilisateur est bien un administrateur, le contrôleur appelle la fonction **donnerToutesLesConferences()** pour récupérer la liste complète des conférences.

Pour chaque conférence, l'administrateur peut également accéder à la liste des participants en utilisant la fonction `donnerParticipants($numConference)`.

Résumé Partie 2 :

1. **Connexion Administrateur** : Validation des informations de connexion et vérification des droits d'accès.
2. **Visualisation des conférences** : Affichage de toutes les conférences disponibles.
3. **Consultation des participants** : Accès à la liste des participants inscrits pour chaque conférence.