IT340 : Génie logiciel

AGIL Clément, DA SILVA Simon, PERROT Rémi, NGUYEN Thomas - Novembre 2016

Architecture de la solution

Cf. Architecture.plantuml

**Flèche pleine** : Relation réelle

**Flèche pointillée** : Relation sémantique (via le routeur)

# Conception détaillée “générale”

### Liste des fichiers

- /

*- .htaccess*

- bdd/

*- BDD.sqlite3*

- librairies/

- server/

*- Router.php*

*- config.php*

*- GetData.php*

*- PostData.php*

*- Session.php*

*- Workshop.php*

*- Grant.php*

*- SendMail.php*

- client/

*- Menu.twig*

*- Home.twig*

*- Lab.twig*

*- WorkshopsForLab.twig*

*- SingleWorkshopForLab.twig*

*- Workshops.twig*

*- SingleWorkshop.twig*

*- Food.twig*

*- Subventions.twig*

*- ResearchJob.twig*

*- Admin.twig*

Les fichiers sont accessibles en lecture-écriture-exécution par le propriétaire (typiquement l’utilisateur représentant le serveur web / PHP). Ils sont bloqués aux autres utilisateurs (au sens Linux du terme).

Le fichier de base de données est accessible en lecture et en écriture par le serveur.

## Fonctionnement

Le fichier .htacess s'occupe de l'URL Rewriting. Toutes les URL auront pour destination réelle (côté serveur) le fichier Router.php.

Chaque URL sera donc réecrit selon le format :

/URL/Navigateur → /server/Router.php?path=/URL/Navigateur

Indiquant ainsi au fichier Router.php le lien d'accès.

Le fichier Router contient une liste des URL d'accès (citées dans la partie “client”), et les actions à effectuer correspondantes. Avant d'accéder aux actions, le fichier charge certain module (par exemple la vérification d'authentification), qui auront un impact sur ces mêmes actions.

Pour chaque action, une liste de modules à charger est donnée, ainsi que la vue (Twig) correspondante.

# Conception détaillée : Serveur

## Router.php

* Class Routage :
  + Objets de classe "Route" : structure de données contenant
    - URL d'accès publique (voir Fonctionnement général)
    - Champ dynamique de l’URL à convertir en paramètres (e.g /atelier/[id])
    - Conditions d'accès (Exemple : être authentifié). En cas de tentative d'accès non connectée à une page privée, un code d'erreur est retourné.
    - Modules serveur à exécuter (Exemple : Workshop.php)
    - Vues à charger (Exemple : Workshops.twig)
    - Optionnel : Redirection vers une URL si un certain nombre de critères sont réunis (inscription effectuée par exemple)
  + Action "main" : parcours des routes en fonction de l'URL d'accès. Gestion de routes par défaut (404, etc.).
* Appel des valeurs par défaut de config.php

## config.php

* Liste des valeurs par défaut :
  + Accès à la BDD
  + Nom de domaine public du site

## Data.php

* Class Data :
  + "connect" : Connexion à la BDD
  + "disconnect" : Déconnexion de la BDD
* Class GetData : Classe de récupération de données, fille de Data
  + Récupération données générales statiques
  + Récupération lieux restaurations
  + Récupération métiers de recherche
  + Récupération des données pour tous les ateliers
  + Récupération des créneaux horaires d’un atelier
* Class PostData : Classe de modification/ajout de données, fille de Data
  + Ajout d’un compte en BDD, avec un code
  + Modification des informations d’un compte
  + Ajout d’un atelier
  + Modification d’un atelier
  + Ajout/modification d’horaires
  + Ajout/suppression de créneaux horaires pour un atelier

## Session.php

* Class Session :
  + "isConnected" : Vérification de connectivité (authentifié)
  + "connect" : Vérification des informations de connexion, et connexion
  + “subscribe” : Vérification des informations et ajout du compte en BDD (via PostData)

## Workshop.php

* Vérification de la validité des informations, et échange avec la BDD (ajout-suppression-modification)
* Synthétisation des données des ateliers

## Grant.php

* Vérification des données de subventions
* Traitement des données des subventions

## SendMail.php

* Envoi d’un email

# Conception détaillée : Client

Cf. Client.plantuml

Une fois la conception établie, il est possible de détailler chaque point essentiel du client. La conception détaillée du client est orientée autour des vues possibles du site.

Pour des raisons de clarté, toutes les fonctionnalités (représentées par les rectangles rouges sur le schéma) seront séparées dans leur propre fichier.

Les langages utilisés sont PHP, HTML5, CSS.

Twig sera utilisé au cours du projet, Twig est un moteur de templates. Un important avantage est la clarté du code : Twig permet de séparer le code HTML du code PHP, évitant ainsi de mélanger les parties serveur et client.

Dans la suite du document :

- L'expression "**Accessible à l'URL avec un compte laboratoire**" signifie que l'URL n'est accessible que par un utilisateur authentifié en tant que laboratoire. Ainsi, le serveur va vérifier si l'utilisateur est connecté comme laboratoire et interdire l'accès à la page aux visiteurs non authentifiés en tant que tel. En revanche, l'expression réduite "Accessible à l'URL" en omettant "avec un compte" signifie que l'URL est accessible à tous les visiteurs.

- Les URL composées des caractères entre crochets, comme par exemple **[id-lab]**, signifient que les caractères entre crochets doivent être remplacés par le numéro id figurant dans la table de la base de données et correspondant à l'élément concerné.

- Toutes les requêtes du client sont envoyées au script **Router.php**. Celui-ci redirigera chaque requête vers les modules et la vue twig correspondants, afin de produire la réponse attendue par le client.

## Liste des URL :

-/

- labs/login

- labs/register

- labs/[id-lab]/

- edit

- workshops/

-creation

- [id-workshop]/

- delete

- edit

- workshops/search

**-** workshops/show/[id-workshop]

- food

- grant

- jobs

- admin/login

- admin/jobs

- creation

- edit

- delete

## Menu.twig :

Ce fichier contient un menu pour permettre une navigation simplifiée. Ce menu sera modifié par l’ajout d’un onglet pour la gestion du compte laboratoire si le visiteur est connecté à un compte laboratoire. Ce menu sera intégré dans toutes les pages du site. Il permettra de naviguer entre l’accueil, la liste des ateliers ou la gestion du compte.

## Home.twig:

Accessible à l’URL : /

Ce fichier contient l’accueil du site. Il y figurera une description de l’évènement.

## Lab.twig:

Connexion :

Accessible à l’URL : **\labs\login**

Contient le formulaire de connexion à un compte laboratoire qui appellera la même URL (appelée via le script serveur Router.php de façon transparente) pour la vérification des données. (Si la vérification est concluante, le script nous redirigera vers **\labs\[id-lab]**).

Sur cette page, un lien vers la création d’un compte est également accessible si l’utilisateur n’en possède pas (URL : **\labs\register).**

Création d’un compte laboratoire :

Accessible à l’URL : **\labs\register**

Contient le formulaire de création de compte, qui appellera sa propre URL (mettant en place une vérification des données, en passant par les modules appelés par le routeur). Si la vérification est concluante, le script nous redirigera vers **\labs\[id-lab].**

Pour la création du compte un code est nécessaire. Ce code est fourni pas l’administrateur lors de l’envoi de mail pour inviter les laboratoires à participer à l’évènement. On ne gère pas l’envoi de mail. Ce mail contient un mot de passe nécessaire au laboratoire pour s’inscrire, pour des raisons de simplicité ce mot de passe sera fixe (ex : atelier\_33\_2017).

Espace laboratoire :

Accessible à l’URL avec un compte laboratoire : **\labs\[id-lab]**

Contient les informations personnelles du laboratoire comme nom, prénom, mail, numéro de téléphone,…

De cette page, un lien est disponible pour la modification des informations du compte (URL avec un compte laboratoire : **\labs\[id-lab]\edit).**

Un lien vers la page de gestion des ateliers est accessible à l’URL avec un compte laboratoire: **\labs\[id-lab]\workshops**.

Un lien pour effectuer une demande de subvention est disponible (URL avec un compte laboratoire : **\grant**).

Modification informations compte :

Accessible à l’URL avec un compte laboratoire : **\labs\[id-lab]\edit**

## WorkshopsForLab.twig:

Accessible à l’URL avec un compte laboratoire: **\labs\[id-lab]\workshops**

Contient la liste des ateliers du compte “id-lab”.

Lien vers la création d’un atelier -> redirection vers un formulaire de création accessible à l’URL avec un compte laboratoire : **\labs\[id-lab]\workshops\creation**

Chaque atelier de la liste possède :

* un lien vers l’affichage de l’atelier (URL avec un compte laboratoire : **\labs\[id-lab]\workshops\[id-workshop]**)
* un lien vers la suppression de l’atelier (URL avec un compte laboratoire : **\labs\[id-lab]\workshops\[id-workshop]\delete**)
* un lien vers la modification d’un atelier (URL avec un compte laboratoire : **\labs\[id-lab]\workshops\[id-workshop]\edit**)

## SingleWorkshopForLab.twig:

Affichage atelier:

Accessible à l’URL avec un compte laboratoire: **\labs\[id-lab]\workshops\[id-workshop]**

Contient toutes les informations de l’atelier (nom, contact, description, lieu, créneaux,..).

Atelier modifiable grâce à un formulaire.

Création d’atelier :

Accessible à l’URL avec un compte laboratoire : **\labs\[id-lab]\workshops\creation**

Contient le formulaire de création d’atelier qui est redirigé vers le script Router.php (si le formulaire est valide, redirection vers **\labs\[id-lab]\workshops\[id-workshop]**).

Modification atelier :

Accessible à l’URL avec un compte laboratoire: **\labs\[id-lab]\workshops\[id-workshop]\edit**

On réutilise le formulaire de création d’atelier avec les valeurs pré-remplies.

Suppression d’un atelier :

Accessible à l’URL avec un compte laboratoire: **\labs\[id-lab]\workshops\[id-workshop]\delete**

Une requête de suppression de l’atelier est envoyée à Router.php.

## Workshops.twig:

Accessible par l’URL: **\workshops\search**

Ce fichier permet de lister tous les ateliers disponibles. Une recherche est aussi disponible avec différents critères de recherche (titre, thèmes, …).

Une fois les ateliers listés, l’utilisateur doit pouvoir sélectionner un atelier afin d’avoir plus d’informations sur celui-ci avec un lien (URL : \workshops\show\[id-workshop])

Un lien vers les lieux de restaurations est disponible (URL : \food)

## SingleWorkshop.twig:

Accessible par l’URL: **\workshops\show\[id-workshop]**

Cette page permet l’affichage de toutes les informations sur un atelier comme les contacts, la localisation, une description de l’atelier avec le thème. Il y a aussi les créneaux proposés pour cet atelier.

## Food.twig:

Accessible par l’URL: **\food**

Cette page regroupe des lieux de restaurations.

## Subventions.twig:

Accessible par l’URL avec un compte Laboratoire : **\grant**

Cette page permet de faire une demande de subvention par les laboratoires grâce à un formulaire.

## ResearchJob.twig :

Pour un visiteur :

Accessible à l’URL : **\jobs**

Permet de visionner des articles sur les différents métiers de la recherche.

Pour l’administrateur :

Accessible à l’URL avec un compte administrateur : **\admin\jobs**

Gestions des articles (ajout/modification/suppression).

Création d’article :

Accessible à l’URL avec un compte administrateur : **\admin\jobs\creation**

Contient le formulaire de création d’article qui est vérifier par les modules serveurs

Modification d’article :

Accessible à l’URL avec un compte administrateur : **\admin\jobs\[id-job]\edit**

On réutilise le formulaire de création d’article avec les valeurs pré-remplies.

Suppression d’un article :

Accessible à l’URL avec un compte administrateur : **\admin\jobs\[id-job]\delete**

Une requête de suppression de l’article est traitée par les modules du serveur (en passant de façon transparente pour le client par Router.php).

## Admin.twig :

1. Connexion :
2. Accessible à l’URL : **\admin\login**

Contient le formulaire de connexion à un compte administrateur. L’administrateur aura tous les privilèges : accès à toutes les pages, possibilité de modifier les différents contenus (ateliers, compte laboratoire…).

# Conception détaillée : BDD

Cf. BDD.mcd

La conception détaillée de la base de données donne les principales liaisons entre chaque table et définit l'utilité de chacune d'elle.

## Admins :

Table contenant les comptes d'authentification des administrateurs du site. Cette table sera remplie manuellement lors du développement du site. La mise à jour de la table se fera uniquement par modification de la table directement. Aucun formulaire de création de compte administrateur ne sera présent sur le site.

## Laboratoires :

Table contenant les comptes d'authentification des laboratoires. Cette table se remplit au moment de l'inscription d'un laboratoire via le formulaire de création du fichier Lab.twig. Le champ "code\_acces" est fixe et contient la valeur "atelier2017". La valeur par défaut est renseignée au moment de la création de la table SQL. Seul l'administrateur pourra modifier ce champ manuellement. Un laboratoire peut être lié à plusieurs ateliers. La liaison sera faite par une contrainte de type FOREIGN Key.

## Ateliers :

Contient les différents ateliers. Cette table se remplit au moment de la création d'un atelier via le formulaire de création du fichier SingleWorkshopForLab.twig.

## Zones :

Contient le nom des zones géographiques localisant les ateliers. Cette table sera remplie au moment de la création ou de la modification d'atelier avec un champ dans le formulaire réservé. Une zone peut être liée à plusieurs ateliers. Cette liaison sera faite par une contrainte FOREIGN Key.

## Themes :

Contient le nom des thèmes des ateliers. Cette table sera remplie au moment de la création ou de la modification d'atelier avec un champ dans le formulaire réservé. Un thème peut être lié à plusieurs ateliers et réciproquement. Une table "est\_defini\_comme" permettra de faire la liaison entre les deux tables.

### Creneaux :

Contient les créneaux horaires des ateliers. Cette table sera remplie au moment de la création ou de la modification d'atelier avec un champ dans le formulaire réservé. Un créneau peut être lié à plusieurs ateliers et réciproquement. Une table "est\_ouvert" permettra de faire la liaison entre les deux tables.

## Metiers :

Contient le nom des métiers de la recherche. Cette table sera remplie par le formulaire de création ou de modification d'articles du fichier ResearchJob.twig.