

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études
pour l'obtention du diplôme de Master en Informatique

Option: Système d'Information et de Connaissances (S.I.C)

Thème

Création D'une Application Web Pour La Gestion d'un laboratoire de recherche

*Cas pratique laboratoire de recherche : Structures Intelligentes "SSL"
Pour Centre Universitaires Belhadj Bouchaib – Ain Témochent*

Réalisé par :

- Louh Menouar

Présenté le 20 Juin 2019 devant le jury composé de :

- *Hadjila Fethalleh* (Président)
- *Merzoug Mohammed* (Encadreur)
- *Abderahim Alae-Eddine..* (Examineur)
- *Benaissa Mohammed* (Invité)

Année universitaire : 2018-2019

Remerciements

Nous remercions Dieu Tout Puissant de nous avoir donné la force, le courage et la patience pour l'élaboration de ce modeste travail.

Nous remercions nos parents pour leurs sacrifices

<< Merci d'être ce que vous êtes >>

Nous remercions nos enseignants.

Merci d'avance aux membres du jury, qui nous ont honorés de leurs participations et attentions portées à notre mémoire de fin d'études.

Enfin, nos remerciements vont à tous ceux et à toutes celles, qui ont contribué de près ou de loin à la réalisation de ce travail avec générosité et un égard exemplaires.

Dédicace

Je tiens à dédie ce modeste travail à :

Ma très chère mère qui m'a toujours soutenu et que j'aime

beaucoup qui ma appris le courage et la patience ;

Tous ceux qui mon aidés de près ou de loin.

Sommaire

TABLEAUX DES MATIÈRES

Dédicace	
Remerciement	
Tableaux des Matières	01
Listes des figures	03
Introduction générale	05
Chapitre I : Généralité laboratoire de recherche .	
Introduction :	07
I.2 Disposition générales	08
I.3 Règles de création	08
I.4. Organisation et fonctionnement	09
I.5 Conclusion	11
Chapitre II : Les outils de web .	
II. Introduction :	13
II.1 Laravel	14
II.1.1 Installation et organisation	15
II.1.1.1 Installation avec Laravel Installer	15
II. 1.1.2 Organisation de Laravel	16
II.1.2 MVC	20
II.1.2.1 Le routage(Routing)	21
II.1.2.2 Controller	21
II.1.2.3Migrations et modèles	23
II. 1.1.2.3.1 Les migrations	23
II.1.3.2 Les modèles	24
II.1.2.4 Les vues	25
II. 2. HTML (L'HypertextMarkupLanguage)	25
II. 2.1 Définition	26
II. 2.2 Les différentes Versions du langage HTML	26
II. 2.3 Les balises	27
II. 1.2.4 Les attributs	28

II. 2.5 Structure de base d'une page HTML	28
II. 3 CSS (Cascading Style Sheets)	29
II. 3.1 Définition	31
II. 3.2 Les versions de CSS	31
II. 3.3 Les avantages des feuilles de style	32
II. 1.4. Langages Script	32
II.4.1 Définition d'un langage script	33
II.4.2 PHP	33
II. 4.3 JSP	35
II.4.4 Le JavaScript	35
II..5 Système de gestion de base de données	36
II.5.1 MySQL	36
II. 5.2 Oracle	37
II.6. Les navigateurs	37
II.7. URL	38
II. Conclusion :	39
Chapitre III : La conception .	
III. Introduction :	40
III.1. Le langage UML	41
III.1.2 Définition	41
III.2. Modélisation	41
III.2.1 Diagramme de cas d'utilisation (Use case)	41
III. 2.2 Les acteurs et leurs rôles	42
III. 2. 3 Diagramme de classes	43
III. 2.4 Schéma relationnel	44
III. 2. 5 Diagrammes de séquences	45
III. 3. Maquettes (Design)	47
III. Conclusion :	50
Conclusion générale	52
Liste bibliographe	

LISTE DES FIGURES

Figure II.1 : MVC	20
Figure III.1 : Diagramme de cas d'utilisation	43
Figure III.2: Diagramme de classe	44
Figure III.3 : Base de données	46
Figure III.4 : Diagramme de séquence de cas d'utilisation « Consulter le site »	46
Figure III.5 : Diagramme de séquence de cas d'utilisation «traitez une demande d'inscription »	47
Figure III.6 : la page d'accueil de laboratoire de recherche	48
Figure III.7: la page d'un membre de laboratoire de recherche	48
Figure III.8 : le formulaire d'ajouter une équipe	49
Figure III.9 : la liste des équipes	49
Figure III.10 : le détail de chaque équipe	50
Figure III.11 : le contact de laboratoire de recherche	50

Introduction Générale

L'informatique est aujourd'hui présente dans la plupart des domaines de la vie professionnelle et privée.

Elle occupe bien évidemment une grande place dans les entreprises, banques, enseignement...

Une page personnelle (page home) pour un chercheur contribue à la publication de ses informations professionnelles et avec la propagation d'utilisation de l'internet ces informations deviennent plus accessibles et plus faciles depuis à consulter.

Les pages constituent le site décrivant les informations personnelles : Nom, Prénom, E-mail,...etc. et professionnelles telles que ses communications, ses publications, ses activités de recherche, ses encadrements,... etc.

Dans ce travail nous avons réalisé un site web intitulé un laboratoire de recherche, notre mémoire comporte trois chapitres sont les suivant :

Un premier chapitre consacré à une présentation générale de laboratoire de recherche informatique (disposition générales, règles de création, organisation et fonctionnement, dispositions financières.)

Le deuxième présente les principaux outils utilisés pour la réalisation d'un site Web dynamique.

Le troisième présente la conception et la modélisation de la base de données.

Chapitre I

GENERALITE LABORATOIRE DE
RECHERCHE

I. Introduction :

Le laboratoire de recherche est créé dans le cadre de la mise en œuvre du programme de recherche de l'établissement de rattachement.

Le laboratoire de recherche a pour missions notamment de :

- ★ Réaliser des objectifs de recherche scientifique et de développement technologique dans un domaine Scientifique précis.
- ★ Exécuter des études et travaux de recherche en rapport avec son objet.
- ★ Contribuer à l'élaboration des programmes de recherche dans le domaine de ses activités.

I.2 Disposition générales

Le laboratoire de recherche associé est créé dans le cadre de la mise en œuvre d'un programme commun à deux ou plusieurs établissements. Les modalités d'association sont fixées par voie de convention.

Le laboratoire de recherche, propre ou associé, est chargé de la mise en oeuvre d'un ou de plusieurs thèmes de recherche scientifique et de développement technologique.

Le laboratoire de recherche a pour missions notamment de :

- ✓ réaliser des objectifs de recherche scientifique et de développement technologique dans un domaine scientifique précis ;
- ✓ exécuter des études et travaux de recherche en rapport avec son objet ;
- ✓ contribuer à l'élaboration des programmes de recherche dans le domaine de ses activités ;
- ✓ contribuer à l'acquisition, à la maîtrise et au développement de nouvelles connaissances scientifiques et technologiques ;

- ✓ participer à l'amélioration et au développement, à son échelle, des techniques et procédés de production ainsi que des produits et des biens et services ;
- ✓ contribuer à la formation par et pour la recherche ;
- ✓ promouvoir et diffuser les résultats de sa recherche.
- ✓ collecter, traiter et capitaliser l'information scientifique et technologique en rapport avec son objet et en faciliter la consultation ;
- ✓ contribuer à la mise en place de réseaux de recherche appropriés.

1.3 Règles de création

La création du laboratoire de recherche est décidée sur la base des critères suivants :

- ✓ importance des activités de recherche par rapport aux besoins du développement socio-économique, culturel, scientifique et technologique du pays ;
- ✓ ampleur et permanence du programme scientifique et/ou technologique dans lequel sont insérées ses activités de recherche ;
- ✓ impact des résultats attendus sur le développement des connaissances scientifiques et technologiques ;
- ✓ qualité et effectif du potentiel scientifique et technique disponible et/ou mobilisable ;
- ✓ moyens matériels et financiers existants et/ou à acquérir ;
- ✓ le laboratoire de recherche doit être constitué d'au moins quatre (04) équipes de recherche.

1.4. Organisation et fonctionnement

Le laboratoire de recherche est dirigé par un directeur et est doté d'un conseil de laboratoire composé des responsables d'équipes de recherche et des chefs des projets de recherche.

a- Directeur de laboratoire de recherche

Le directeur de laboratoire de recherche est nommé pour une durée de trois (03) années renouvelable par l'autorité de tutelle, sur proposition du responsable de l'institution de rattachement, parmi deux (02) candidats ayant le grade le plus élevé, élus en son sein par les membres du conseil de laboratoire. Il est mis fin à ses fonctions dans les mêmes formes. Le directeur du laboratoire de recherche :

- ✓ présente un bilan des activités de recherche et de gestion au conseil de laboratoire dans un délai n'excédant pas un mois à compter de la date de sa fin de fonctions ;
- ✓ assure les directions scientifiques et la gestion financière du laboratoire. Il est ordonnateur des crédits alloués au laboratoire ;
- ✓ est responsable du bon fonctionnement du laboratoire de recherche et exerce l'autorité hiérarchique sur l'ensemble des personnels de recherche et de soutien affectés au laboratoire ;
- ✓ peut, par délégation du chef de l'établissement de rattachement, initier et engager des contrats et des conventions pour la réalisation des travaux de recherche, les études ou les prestations de services avec des organismes nationaux et/ou internationaux en rapport avec les missions du laboratoire et conformément à la réglementation en vigueur ;
- ✓ soumet ses programmes et ses bilans à l'examen des organes d'évaluation de l'institution de rattachement.

b- Equipe de recherche

L'équipe de recherche, dirigée par un chercheur qualifié comprend au minimum trois (03) chercheurs. Elle a pour mission principale :

- ✓ d'exécuter un ou plusieurs projets de recherche entrant dans le cadre du programme du laboratoire ;
- ✓ chaque projet de recherche est conduit par un responsable de projet ;
- ✓ le chef d'équipe peut également être chef de projet de recherche.

c- Conseil du laboratoire

Présidé par le directeur du laboratoire, le conseil de laboratoire est chargé notamment :

- ✓ de contribuer à l'élaboration des programmes ;
- ✓ d'évaluer, périodiquement, les activités de recherche ;
- ✓ d'examiner et d'approuver le bilan des activités de recherche et de gestion ;
- ✓ d'adopter les états prévisionnels des recettes et des dépenses présentés par le directeur ;
- ✓ de veiller à l'utilisation rationnelle des moyens humains, matériels et financiers ;
- ✓ d'élaborer et d'adopter son règlement intérieur.^[1]

^[1][Journal officiel de la république algérienne, article N° 77, 1999, p4, p5, p6 \(01/05/2019\)](#)

I.5 Conclusion

Dans ce chapitre, on s'est particulièrement intéressé à une présentation générale d'un laboratoire de recherche en énonçant les dispositions générales et règles de création, son organisation et son fonctionnement.

Chapitre II

LES OUTILS DE WEB

Introduction :

Dans la tendance actuelle de la conception des sites Web, chaque un des composants joue son rôle, généralement chaque fichier Web véhicule trois éléments :

1. Une information : c'est l'essentiel du contenu que le visiteur est venu chercher sur Internet ; il est essentiel pour les Web master d'assurer qu'il le trouve dans tous les cas.
2. La mise en forme de toutes les informations.
3. Des comportements de la page Web comme des animations, la réponse à des actions du visiteur, etc.

Laravel est un Framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet, pour fonctionner Laravel a besoin d'un certain environnement :

- ✓ Heureusement il existe des solutions toutes prêtes, par exemple pour PHP + MySql : wampserver, xampp, easyphp.
- ✓ Composer
- ✓ Les éditeurs de code (Sublime Texte, Visual Studio Code).

II.1 Laravel

Laravel, créé par Taylor Otwell, initie une nouvelle façon de concevoir un framework en utilisant ce qui existe de mieux pour chaque fonctionnalité.

Par exemple toute application web a besoin d'un système qui gère les requêtes HTTP. Plutôt que de réinventer quelque chose, le concepteur de Laravel a tout simplement utilisé celui de Symfony en l'étendant pour créer un système de routage efficace.

De la même manière, l'envoi des emails se fait avec la bibliothèque SwiftMailer. En quelque sorte Otwell a fait son marché parmi toutes les bibliothèques disponibles. Nous verrons dans ce cours comment cela est réalisé. Mais Laravel ce n'est pas seulement le regroupement de bibliothèques existantes, c'est aussi de nombreux composants originaux et surtout une orchestration de tout ça.

Vous allez trouver dans Laravel :

- ✓ un système de routage (RESTful et ressources),
- ✓ un créateur de requêtes SQL et un ORM,
- ✓ un moteur de template,
- ✓ un système d'authentification pour les connexions,
- ✓ un système de validation,
- ✓ un système de pagination,
- ✓ un système de migration pour les bases de données,
- ✓ un système d'envoi d'emails,
- ✓ un système de cache,
- ✓ un système de gestion des événements,
- ✓ un système d'autorisations,
- ✓ une gestion des sessions,
- ✓ un système de localisation,
- ✓ un système de notifications...

Laravel est un framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet. ^[1]

II.1.1 Installation et organisation

Composer fonctionne en ligne de commande. Vous avez donc besoin de la console (nommée Terminal ou Konsole sur OS X et Linux). Les utilisateurs de Linux sont très certainement habitués à l'utilisation de la console mais il en est généralement pas de même pour les adeptes de Windows. Pour trouver la console sur ce système il faut chercher l'invite de commande :



Téléchargez et exécutez Composer-Setup.exe. Il installera la dernière version de Composer et configurera votre PATH afin que vous puissiez appeler composer à partir de n'importe quel répertoire de votre ligne de commande^[2].

II.1.1.1 Installation avec Laravel Installer

Une autre solution pour installer Laravel consiste à utiliser l'installateur. Il faut commencer par installer globalement l'installateur avec composer :

```
composer global require "Laravel/installer"
```

Il faut ensuite informer la variable d'environnement path de l'emplacement du dossier .../composer/vendor/bin.

Pour créer une application il suffit de taper :

```
laravel new monAppli
```

Laravel est équipé d'un serveur sommaire pour le développement qui se lance avec cette commande :

```
php artisan serve
```

^[1] <http://laravel.sillo.org/cours-laravel-5-5-les-bases-presentation-generale/>

^[2] <https://getcomposer.org/doc/00-intro.md>

On y accède à cette adresse : <http://localhost:8000>. Mais évidemment pour que ça fonctionne il faut que vous ayez PHP installé.

II. 1.1.2 Organisation de Laravel

Maintenant qu'on a un Laravel tout neuf et qui fonctionne voyons un peu ce qu'il contient...

Dossier app

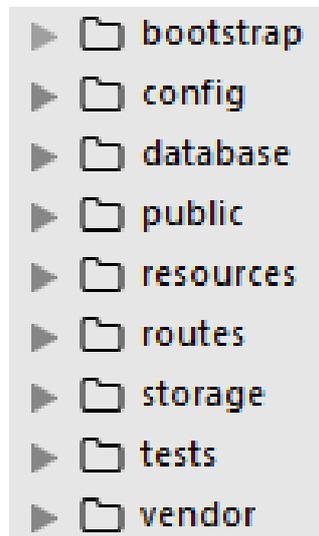
Ce dossier contient les éléments essentiels de l'application :



- ✓ **Console** : toutes les commandes en mode console,
- ✓ **Http** : tout ce qui concerne la communication : contrôleurs, middlewares (il y a 4 middlewares de base qui servent à filtrer les requêtes HTTP) et le kernel,
- ✓ **Providers** : tous les fournisseurs de services (providers), il y en a déjà 5 au départ. Les providers servent à initialiser les composants.
- ✓ **User** : un modèle qui concerne les utilisateurs pour la base de données.

Évidemment tout cela doit vous paraître assez nébuleux pour le moment mais nous verrons en détail ces éléments au fil du cours. Et on verra d'ailleurs que seront créés bien d'autres dossiers selon nos besoins.

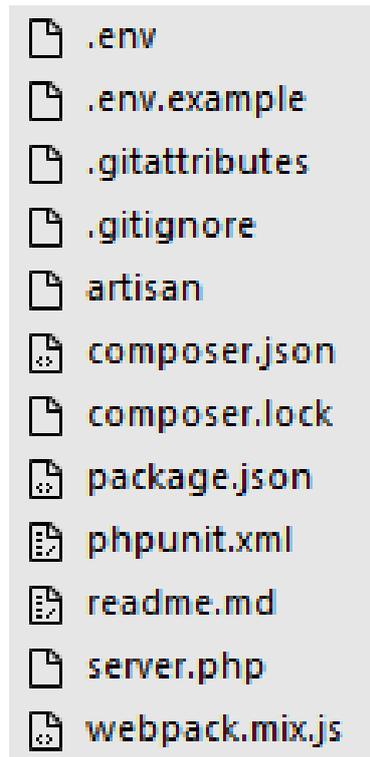
Autres dossiers



Voici une description du contenu des autres dossiers :

- ✓ **bootstrap** : scripts d'initialisation de Laravel pour le chargement automatique des classes, la fixation de l'environnement et des chemins, et pour le démarrage de l'application,
- ✓ **public** : tout ce qui doit apparaître dans le dossier public du site : images, CSS, scripts...
- ✓ **config** : toutes les configurations : application, authentification, cache, base de données, espaces de noms, emails, systèmes de fichier, session...
- ✓ **database** : migrations et populations,
- ✓ **resources** : vues, fichiers de langage et assets (par exemple les fichiers Sass),
- ✓ **routes** : la gestion des urls d'entrée de l'application,
- ✓ **storage** : données temporaires de l'application : vues compilées, caches, clés de session...
- ✓ **tests** : fichiers de tests,
- ✓ **vendor** : tous les composants de Laravel et de ses dépendances (créé par composer),

Fichiers de la racine

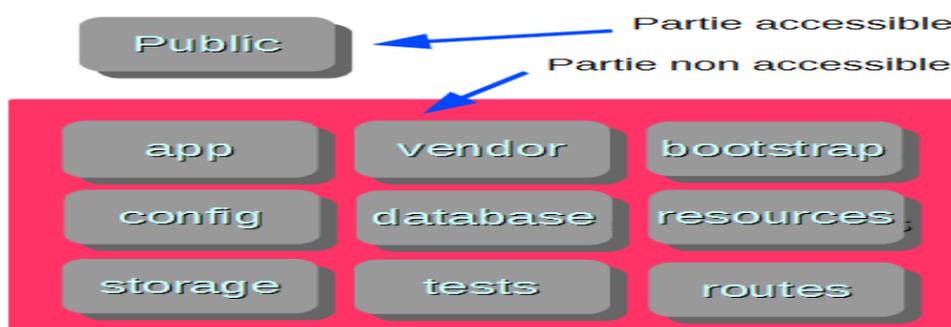


Il y a un certain nombre de fichiers dans la racine dont voici les principaux :

- ✓ **artisan** : outil en ligne de Laravel pour des tâches de gestion,
- ✓ **composer.json** : fichier de référence de composer,
- ✓ **package.json** : fichier de référence de npm pour les assets,
- ✓ **phpunit.xml** : fichier de configuration de phpunit (pour les tests unitaires),
- ✓ **.env** : fichier pour spécifier l'environnement d'exécution.

Accessibilité

Pour des raisons de sécurité sur le serveur seul le dossier **public** doit être accessible :



Cette configuration n'est pas toujours possible sur un serveur mutualisé, il faut alors modifier un peu Laravel pour que ça fonctionne; j'en parlerai dans le chapitre sur le déploiement.

Par défaut lorsque vous installez Laravel, celui-ci est en mode « debug » et vous aurez une description précise de toutes les erreurs. Par exemple ouvrez le fichier `routes/web.php` et changez ainsi le code :

Environnement et messages d'erreur

The image shows a browser window on the left displaying a stack trace for a `MethodNotAllowedHttpException`. The stack trace includes frames from `Symfony\Component\HttpFoundation` and `Illuminate\Routing`. The right side shows a code editor with the source code for `RouteCollection.php`, highlighting the `methodNotAllowed` method which throws the exception.

```
Symfony \ Component \ HttpKernel \
Exception \
MethodNotAllowedHttpException
No message
COPY

Application frames (1) All frames (27)

26 Symfony\Component\HttpFoundation\Exception
MethodNotAllowedHttpException
...
\vendor\laravel\framework\src\Illuminate\Ro
:251

25 Illuminate\Routing\RouteCollection
methodNotAllowed
...
\vendor\laravel\framework\src\Illuminate\Ro
:238

24 Illuminate\Routing\RouteCollection
getRouteForMethods
...
\vendor\laravel\framework\src\Illuminate\Ro
:176

23 Illuminate\Routing\RouteCollection match
...
\vendor\laravel\framework\src\Illuminate\Ro
:584

22 Illuminate\Routing\Router findRoute
...
```

```
E:\laragon\www\laravel5\vendor\laravel\framework\src\Illuminate\Routing\RouteCollection.php
241. /**
242.  * Throw a method not allowed HTTP exception.
243.  *
244.  * @param array $others
245.  * @return void
246.  *
247.  * @throws
248.  * \Symfony\Component\HttpFoundation\Exception\MethodNotAllowedHttpException
249.  */
249. protected function methodNotAllowed(array $others)
250. {
251.     throw new MethodNotAllowedHttpException($others);
252. }
253.
254. /**
255.  * Get routes from the collection by method.
256.  *
257.  * @param string|null $method
258.  * @return array
259.  */
260. public function get($method = null)
261. {
262.     return is_null($method) ? $this->getRoutes() : Arr::get($this->routes,
    $method);
    }
    }

Arguments
1. ""

No comments for this stack frame.
```

Environment & details:

- GET Data empty
- POST Data empty
- Files empty
- Cookies

Autrement dit on va chercher la valeur dans l'environnement, mais où peut-on le trouver ? Regardez à la racine des dossiers, vous y trouvez le fichier `.env` :

Avec ce contenu :

```
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:yrTZiYq/TxheZ3tBcqHo4X47uuSXROCY5gg3H0Pljt8=
APP_DEBUG=true
APP_LOG_LEVEL=debug
APP_URL=http://localhost
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=homestead
DB_USERNAME=homestead
DB_PASSWORD=secret [1].
```

II.1.2 MVC

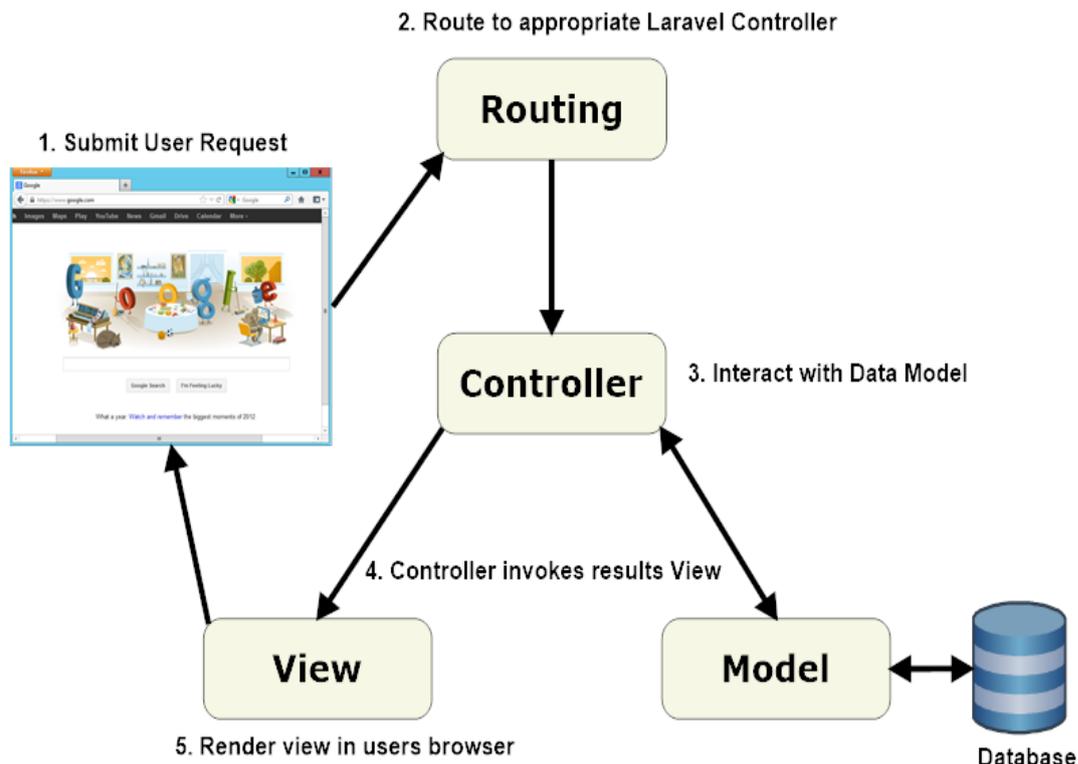


Figure II.1 : MVC

[1] <http://laravel.sillo.org/cours-laravel-5-5-les-bases-installation-et-organisation/>

Laravel est un framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet

II.1.2.1 Le routage(Routing)

Il est indispensable de connaître les principales méthodes du HTTP :

```
Route::get('A/equipes','AdminController@equipements');
Route::get('A/create','AdminController@create');
Route::post('/A','AdminController@store');
Route::get('A/{id}/detail','AdminController@show');
Route::get('A/{id}/edit','AdminController@edit');
Route::put('A/{id}','AdminController@update');
Route::delete('A/{id}','AdminController@destroy');
```

- ✓ **GET** : c'est la plus courante, on demande une ressource qui ne change jamais, on peut mémoriser la requête, on est sûr d'obtenir toujours la même ressource,
- ✓ **POST** : elle est aussi très courante, là la requête modifie ou ajoute une ressource, le cas le plus classique est la soumission d'un formulaire (souvent utilisé à tort à la place de PUT),
- ✓ **PUT** : on ajoute ou remplace complètement une ressource,
- ✓ **DELETE** : on supprime une ressource. ^[1].

Dans chaque requête HTTP en fait l'appel le contrôleur et le processus auquel il appartient.

II.1.2.2 Controller

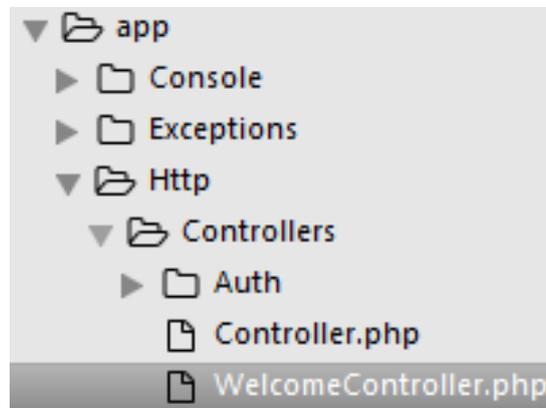
La tâche d'un contrôleur est de réceptionner une requête (qui a déjà été sélectionnée par une route) et de définir la réponse appropriée.

Pour créer un contrôleur nous allons utiliser Artisan. Dans la console entrez cette commande :

[1] <http://laravel.sillo.org/cours-laravel-5-5-les-bases-le-routage/>

PHP artisan make:controller WelcomeController --Resource

Si tout se passe bien vous allez trouver le contrôleur ici :



Il est indispensable de connaître les principales méthodes du Controller :

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class WelcomeController extends Controller
8 {
9     //Récupérer les informaion
10    public function index()
11    {
12        $users = User::all();
13        return view('equipe.index');
14    }
15    //Afficher le formulaire
16    public function create()
17    {
18        return view('equipe.create');
19    }
20    //Récupérer les information
21    public function details($id){ return view('equipe.details');}
22    //Enregistrer les information
23    public function store(equipeRequest $request){}
24    //Modéfier les information
25    public function update(equipeRequest $request,$id) {}
26    //Supprimer les information
27    public function destroy($id){}
28 }
```

Dans chaque méthode du contrôleur nous appellerons les modèles que nous utilisons^[1].

[1] <http://laravel.sillo.org/cours-laravel-5-3-les-bases-artisan-et-les-contrôleurs/>

II.1.2.3 Migrations et modèles

II. 1.1.2.3.1 Les migrations

Une migration permet de créer et de mettre à jour un schéma de base de données. Autrement dit, vous pouvez créer des tables, des colonnes dans ces tables, en supprimer, créer des index... Tout ce qui concerne la maintenance de vos tables peut être pris en charge par cet outil. Vous avez ainsi un suivi de vos modifications.

Pour créer une migration nous allons utiliser Artisan. Dans la console entrez cette commande :

```
PHP artisan make:migration create_table_users --create=users
```

Si tout se passe bien vous allez trouver le contrôleur ici :

```
└─ migrations
  └─ 2019_03_18_191103_create_themes_table....
  └─ 2019_03_18_191519_create_labos_table.php
  └─ 2019_03_18_200719_create_axes_table.php
  └─ 2019_03_18_203619_create_projet_table.p...
  └─ 2019_03_19_073157_create_table_users.php
```

Avec ce code de base :

```
1  <?php
2
3  use Illuminate\Support\Facades\Schema;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Database\Migrations\Migration;
6
7  class CreateTableUsers extends Migration
8  {
9      /**
10     * Run the migrations.
11     *
12     * @return void
13     */
14     public function up()
15     {
16         Schema::create('users', function (Blueprint $table) {
17             $table->increments('id');
18             $table->string('name')->nullable();
19             $table->string('prenom')->nullable();
20             $table->string('sex')->nullable();
21             $table->string('grad')->nullable();
22             $table->string('deplome')->nullable();
23             $table->string('speciatlite')->nullable();
24             $table->string('email')->nullable();
25             $table->string('photo')->nullable();
26             $table->string('date_naissance')->nullable();
27             $table->string('grade')->nullable();
28             $table->string('password')->nullable();
29             $table->string('num_tel');
30             $table->timestamps();
31         });
32     }
```

```

33
34
35     /**
36     * Reverse the migrations.
37     *
38     * @return void
39     */
40     public function down()
41     {
42         Schema::dropIfExists('users');
43     }
44

```

Pour enregistrer la table users dans la base de donner nous allons utiliser Artisan.

Dans la console entrez cette commande :

PHP artisan make:migrate

Lorsque vous lancez les migrations la table contacts est créée :

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id	int(10)		UNSIGNED	Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	axe_id	int(10)		UNSIGNED	Oui	NULL			Modifier Supprimer Plus
3	name	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
4	prenom	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
5	sexe	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
6	grad	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
7	deplome	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
8	specialite	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
9	email	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
10	photo	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
11	date_naissance	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
12	password	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
13	num_tel	varchar(255)	utf8mb4_unicode_ci		Oui	NULL			Modifier Supprimer Plus
14	created_at	timestamp			Oui	NULL			Modifier Supprimer Plus
15	updated_at	timestamp			Oui	NULL			Modifier Supprimer Plus
16	is_admin	varchar(255)	utf8mb4_unicode_ci		Non	0			Modifier Supprimer Plus

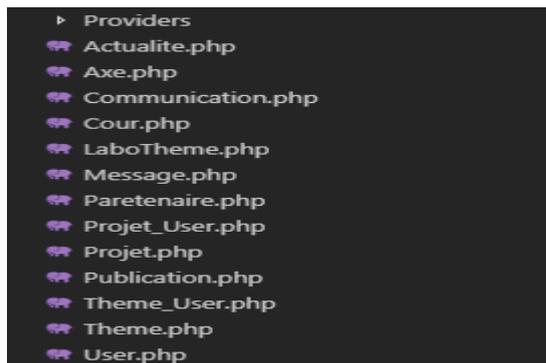
II.1.3.2 Les modèles

Une migration permet de créer une image sur migration.

Pour créer une migration nous allons utiliser Artisan. Dans la console entrez cette commande :

PHP artisan make:modele User

Si tout se passe bien vous allez trouver le contrôleur ici



Avec ce code de base :

```
1 <?php
2
3 namespace App;
4 use Illuminate\Notifications\Notifiable;
5 use Illuminate\Contracts\Auth\CanResetPassword;
6 use Illuminate\Foundation\Auth\User as Authenticatable;
7 //use Illuminate\Contracts\Auth\Authenticatable;
8 use Illuminate\Database\Eloquent\Model;
9
10
11 class User extends Authenticatable
12 { /* use Notifiable;
13
14     /**
15      * The attributes that are mass assignable.
16      *
17      * @var array
18      */
19     protected $fillable = [
20         'name', 'email', 'password', 'is_admin',
21     ];
22
23     /**
24      * The attributes that should be hidden for arrays.
25      *
26      * @var array
27      */
28     protected $hidden = [
29         'password', 'remember_token',
30     ];
31 }
```

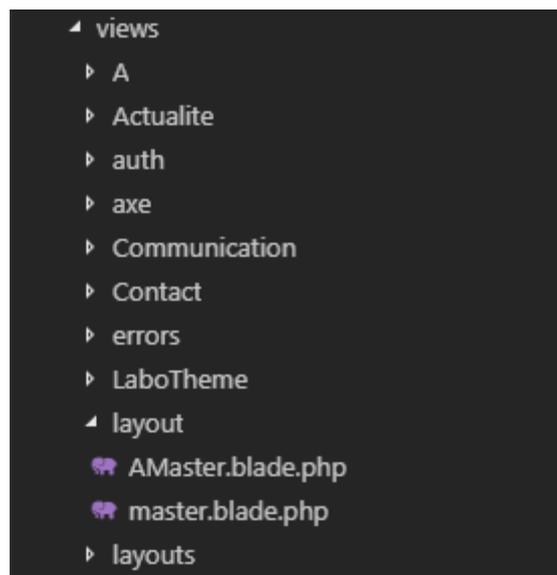
Lés méthodes des contrôler retourne des vues^[1].

II.1.2.4 Les vues

Une migration permet d'afficher le résultat de la requête basée sur HTML, CSS, JS.

Pour créer une vues, il faut créer le fichier index.blade.php dans le dossier views

[1] <http://laravel.sillo.org/cours-laravel-5-3-les-donnees-migrations-et-modeles/>



Créer un fichier master.blade.php dans le répertoire ressources/views/layouts

Mettre dedans le contenu HTML qui ne change pas dans les pages du site

Rajouter dessus la partie dynamique suivante:@yield ('content')

Dans un fichier vue index.blade.php dans ressources/viens

Mettre dedans le contenu suivant :

```
@extends ('layouts.master')
```

```
@Section ('content')
```

```
...contenu HTML
```

```
@endsection
```

II. 2. HTML (L'HypertextMarkupLanguage)

II. 2.1 Définition

L'HyperText Markup Language, généralement abrégé HTML, est le langage de balisage conçu pour représenter les pages web. C'est un langage permettant d'écrire de l'hypertexte, d'où son nom. HTML permet également de structurer sémantiquement et logiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie et des programmes informatiques.

Il permet de créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web. Il est souvent utilisé conjointement avec le langage de programmation JavaScript et des feuilles de

style en cascade (CSS). HTML est inspiré du Standard Generalized Markup Language (SGML). Il s'agit d'un format ouvert^[1].

II. 2.2 Les différentes Versions du langage HTML

Au fil du temps, les langages HTML ont beaucoup évolué. Dans la toute première version de HTML (HTML 1.0) il n'était même pas possible d'afficher des images !Voici un très bref historique de ce langage:

HTML 1 : C'est la toute première version créée par Tim Berners-Lee en 1991. Le navigateur

Internet utilisé à l'époque était nommé NCSA Mosaic.

HTML 2 La deuxième version du HTML apparaît en 1994 et prend fin en 1996 avec l'apparition du HTML3.0

HTML 3.0 : C'est cette version qui posera en fait les bases des versions suivantes du HTML.

Les règles et le fonctionnement de cette version sont donnés par le W3C (tandis que la première version a été créée par un seul homme). Cette nouvelle version du HTML rajoute de nombreuses possibilités au langage comme les tableaux, les applets, les scripts, le positionnement du texte autour des images, etc.

HTML 4 : Le 18 décembre 1997, le HTML 4.0 a été publié. La version 4.0 du langage HTML standardise notamment les frames (qui découpent une page Web en plusieurs parties), des tableaux plus complexes, des améliorations sur les formulaires, etc. Mais surtout, cette version permet pour la première fois d'exploiter des feuilles de style, notre fameux CSS. La version HTML 4.01, apparue le 24 décembre 1999 apporte quelques modifications mineures au HTML 4.0.

HTML 5 : C'est la dernière version. Le HTML 5.0 est en cours de spécification en 2012. La version 5.0 du langage HTML définit deux syntaxes : HTML5 et XHTML5. Cette version apporte de nouvelles possibilités et nombreuses améliorations en termes de création d'applications Web riches comme la possibilité d'inclure facilement des vidéos, un meilleur agencement du contenu, de nouvelles fonctionnalités pour les

^[1] https://fr.wikipedia.org/wiki/Hypertext_Markup_Language

formulaire, etc^[1].

II. 2.3 Les balises

Les pages HTML sont remplies de ce qu'on appelle des balises. Celles-ci sont invisibles à l'écran pour les visiteurs, mais elles permettent à l'ordinateur de comprendre ce qu'il doit afficher. Les balises se repèrent facilement. Elles sont entourées de « chevrons », c'est-à-dire des symboles < et >, comme ceci: <balise> mais à quoi est-ce qu'elles servent ?

Elles indiquent la nature du texte qu'elles encadrent. par exemple elles veulent dire : «Ceci est le titre de la page » « Ceci est une image» « Ceci est un paragraphe de texte», etc.

On distingue deux types de balises : les balises en paires et les balises orphelines.

a. Les balises en paires :

Elles s'ouvrent, contiennent du texte, et se ferment plus loin. Voici à quoi elles ressemblent :

Code : HTML<Titre>Ceci est un titre</titre>.On distingue une balise ouvrante (<titre>)et

une balise fermante (</titre>) qui indique que le titre se termine.

b. Les balises orphelines :

Ce sont des balises qui servent le plus souvent à insérer un élément à un endroit précis (par exemple une image). Il n'est pas nécessaire de délimiter le début et la fin de l'image, on veut juste dire à l'ordinateur « Insère une image ici ».

Une balise orpheline s'écrit comme ceci : Code : HTML < image />

II. 1.2.4 Les attributs

Les attributs sont un peu les options des balises. Ils viennent les compléter pour donner des informations supplémentaires. L'attribut se place après le nom de la balise ouvrante et a le plus souvent une valeur, comme ceci :

Code : HTML<balise attribut="valeur">

À quoi cela sert-il ? Prenons la balise <image />, on pourrait rajouter un attribut qui indique le nom de l'image à afficher :

[1] https://fr.wikipedia.org/wiki/Hypertext_Markup_Language#%C3%89volution_du_langage

Code : HTML : `<image nom="photo.jpg"/>`

II. 2.5 Structure de base d'une page HTML

On prend un code correspond à la base d'une page Web en HTML (Structure générale d'un document HTML) :

Code: HTML

```
<!DOCTYPE html>
<html>
<head>
<meta charset= "utf-8"/>
<title>Titre</title>
</head>
<body>
</html>
```

Le doctype:

La toute première ligne s'appelle le doctype. Elle est indispensable car c'est elle qui indique qu'il s'agit bien d'une page Web HTML.

Ce n'est pas vraiment une balise comme les autres (elle commence par un point d'exclamation).

La balise `<html>`:

Code : HTML `<html></html>`

C'est la balise principale du code. Elle englobe tout le contenu de la page. Comme vous pouvez le voir, la balise fermante `</html>` se trouve tout à la fin du code.

L'en-tête `<head>` :

Une page Web est constituée de deux parties :

Au plus haut niveau, L'en-tête `<head>` : cette section donne quelques informations générales sur la page comme son titre, l'encodage (pour la gestion des caractères spéciaux), etc. Cette section est généralement assez courte. Les informations que contient l'en-tête ne sont pas affichées sur la page, ce sont simplement des informations générales à destination de l'ordinateur. Elles sont cependant très importantes !

L'encodage (charset) : `<metacharset= "utf-8"/>`

Cette balise indique l'encodage utilisé dans le fichier .html. L'encodage indique la

façon dont le fichier est enregistré. C'est lui qui détermine comment les caractères spéciaux vont s'afficher (accents, idéogrammes chinois et japonais, caractères arabes, etc.).

Cette méthode d'encodage permet d'afficher sans aucun problème pratiquement tous les symboles de toutes les langues de notre planète !

Le titre principal de la page

<Title> C'est le titre de la page, probablement l'élément le plus important ! Toute page doit avoir un titre qui décrit ce qu'elle contient.

Le titre ne s'affiche pas dans la page mais en haut de celle-ci (souvent dans l'onglet du navigateur).

Le corps

Le corps <body> : c'est là que se trouve la partie principale de la page. Tout ce que nous écrivons ici sera affiché à l'écran. C'est à l'intérieur du corps que nous écrivons la majeure partie de notre code.

Les éléments de body :

Il est possible d'indiquer la langue de n'importe quelle partie du document et de gérer le mélange de texte s'écrivant de gauche à droite avec du texte de droite à gauche.

i. Listes : HTML différencie des listes non ordonnées et des listes ordonnées, selon que l'ordre formel du contenu dans le code est en soi ou non une information. Des listes de définition existent également, mais sans que leur champ d'application ne soit exactement déterminé.

ii. Tables : Cette fonctionnalité a été développée pour permettre la présentation de données tabulaires mais a été immédiatement exploitée pour ses puissantes capacités de mise en page

iii. Cadres : Aussi connu sous le nom de frames, une fonctionnalité souvent décriée qui permet d'afficher plusieurs documents HTML dans une même fenêtre.

iv. Formulaire pour l'insertion interactive de données :

Les éléments de formulaire permettent aux visiteurs d'entrer du texte et des fichiers dans les pages Web

Les éléments de formulaire permettent aux visiteurs d'entrer du texte et des fichiers v. Scripts : Permet d'associer des morceaux de programmes aux actions des utilisateurs

sur le document. Les langages utilisés sont généralement JavaScript et VB Script^[1].

II. 3 CSS (Cascading Style Sheets)

II. 3.1 Définition

CSS (Cascading Style Sheets) : feuilles de style en cascade) est un langage informatique qui sert à décrire la présentation des documents HTML et XML. C'est le langage de mise en forme des sites Web. Alors que le HTML permet d'écrire le contenu des pages Web et de les structurer, C'est en CSS que l'on choisit notamment la couleur, la taille des menus et bien d'autres choses encore.

Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C).

Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception de sites Web et bien pris en charge par les navigateurs Web dans les années 2000.

L'un des objectifs majeurs des CSS est de permettre la mise en forme hors des documents. Il est par exemple possible de ne décrire que la structure d'un document en HTML, et de décrire toute la présentation dans une feuille de style CSS séparée^[2].

II. 3.2 Les versions de CSS

Depuis sa création, le témoin de nombreux développements, on cite les versions suivantes :

CSS 1: dès 1996, on dispose de la première version du CSS. Elle pose les bases de ce langage qui permet de présenter sa page Web, comme les couleurs, les marges, les polices de caractères, etc.

CSS 2 : apparue en 1999 puis complétée par CSS 2.1, cette nouvelle version de CSS rajoute de nombreuses options. On peut désormais utiliser des techniques de

[1] <https://openclassrooms.com/fr/courses/1603881-apprenez-a-creer-votre-site-web-avec-html5-et-css3/1608357-memento-des-balises-html>

[2] https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade

positionnement très précises, qui nous permettent d'afficher des éléments où on le souhaite sur la page.

CSS 3 : c'est la dernière version, qui apporte des fonctionnalités particulièrement attendues comme les bordures arrondies, les dégradés, les ombres, etc.

CSS4 : L'écriture quatrième niveau des feuilles de styles en cascade débute dès 2010, parallèlement aux modules CSS 3^[1].

II. 3.3 Les avantages des feuilles de style

- ✓ La structure du document et la présentation peuvent être gérées dans des fichiers séparés.
- ✓ La conception d'un document se fait dans un premier temps sans se soucier de la présentation, ce qui permet d'être plus efficace.
- ✓ Dans le cas d'un site Web, la présentation est uniformisée : les documents (pages HTML) font référence aux mêmes feuilles de styles. Cette caractéristique permet de plus une remise en forme rapide de l'aspect visuel.
- ✓ Un même document peut donner le choix entre plusieurs feuilles de style, par exemple une pour l'impression et une pour la lecture à l'écran. Certains navigateurs Web permettent au visiteur de choisir un style parmi plusieurs.
- ✓ Le code HTML est considérablement réduit en taille et en complexité, puisqu'il ne contient plus de balises ni d'attributs de présentation.

HTML et CSS n'ont pas de concurrents car ce sont des standards. Tout le monde est censé les connaître et les utiliser sur tous les sites Web.

II. 1.4. Langages Script

Qu'est-ce qu'un script ?

Un script est un programme dont le code source est inclus dans un document HTML. Ce programme est interprété et s'exécute sur la machine du client lorsque le document est chargé ou lors d'une action de l'utilisateur (clic ou déplacement du curseur par exemple).

[1] https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade#Vers_une_maturation_difficile

II.4.1 Définition d'un langage script

Un langage de script est un langage de programmation qui permet de dynamiser un site Web.

Le langage de script est généralement exécuté à partir de fichiers contenant le code source du programme qui sera interprété. Les premiers langages étaient souvent appelés « langage de commande » ou « langage d'enchaînement des travaux » car ils permettaient simplement d'automatiser une succession de commandes simples. Les langages scripts sont utilisés côté serveur Web comme une méthode pour générer des pages dynamiques, côté navigateur sous forme d'Applet d'automatisation, et côté OS comme une méthode d'automatisation et pour les interfaces graphique, permettant de simuler des déplacements de la souris et des pressions de touches sur le clavier, parmi ces langages on cite :

II.4.2 PHP

Définition

Le PHP signifie : HypertextPreprocessor a été créé en 1994 par RasmusLerdorf. Il est à l'origine issu de librairies Pearl [9]. PHP un langage de scripts orienté objet permettant de rendre les pages Web dynamiques, il est principalement utilisé pour être exécuté par un serveur http, mais il peut également fonctionner comme n'importe quel langage interprété de façon locale, il ne prend vraiment de l'intérêt que s'il est combiné à un outil tel que MySQL.

Le PHP est un langage de programmation compatible avec plusieurs serveurs HTTP comme Apache et IIS. Pour PhP, les choses sont compliquées par le fait que les programmes ne sont pas censés s'exécuter sur la machine de l'utilisateur, là où ils affichent leurs résultats, mais sur un serveur lointain chez l'hébergeur du site. De fait, un programme PhP produit deux sortes de résultats :

1. Il effectue des actions comme créer ou modifier des fichiers ou des bases de données chez l'hébergeur.
2. Il produit des sorties qui seront affichées en tant que page Web chez l'internaute.

On peut insérer du code PHP n'importe où dans un code HTML. Ces bouts de code PHP

seront les parties dynamiques de la page, c'est-à-dire les parties qui peuvent changer toutes seules. Le PHP est justement de générer du code HTML, générer du CSS, mais c'est plus rare.

Les clients sont incapables de comprendre le code PHP : ils ne connaissent que le HTML et le CSS. Seul le serveur est capable de lire du PHP. Donc pour qu'un ordinateur puisse lire du PHP, il faut qu'il se comporte comme un serveur et installer des programmes comme ceux que l'on trouve sur les serveurs qui délivrent les sites Web aux internautes. Ces programmes dont nous allons avoir besoin sont :

I. Apache : C'est ce qu'on appelle un serveur Web. Il s'agit du plus important de tous les programmes, car c'est lui qui est chargé de délivrer les pages Web aux visiteurs. Cependant, Apache ne gère que les sites Web statiques (il ne peut traiter que des pages HTML). Il faut donc le compléter avec d'autres programmes.

II. PHP : Hyper text Preprocessor : c'est un plug-in pour Apache qui le rend capable de traiter des pages Web dynamiques en PHP. En clair, en combinant Apache et PHP, notre ordinateur sera capable de lire des pages Web en PHP.

III. MySQL: c'est ce qu'on appelle un SGBD (Système de Gestion de Base de Données). Le MySQL est logiciel de gestion de bases de données. Il permet d'enregistrer des données de manière organisée afin de nous aider à les retrouver facilement plus tard.

Le langage qui permet de communiquer avec la base de données s'appelle le SQL. Et d'autre comme :

- ✓ Server SideIncludes (SSI)
- ✓ JavaServer Pages (JSP)
- ✓ JavaServer Faces (JSF)
- ✓ Common Gateway Interface (CGI)^[1].

[1] <https://www.php.net/manual/fr/history.php>

II. 4.3 JSP

Le JavaServer Pages ou JSP est une technique basée sur Java qui permet aux développeurs de créer dynamiquement du code HTML, XML ou tout autre type de page web. Cette technique permet au code Java et à certaines actions prédéfinies d'être ajoutés dans un contenu statique. Depuis la version 2.0 des spécifications, la syntaxe JSP est complètement conforme au standard XML.

La syntaxe du JSP ajoute des balises XML, appelées actions JSP, qui peuvent être utilisées pour appeler des fonctions. De plus, cette technique permet la création de bibliothèques de balises JSP (taglib) qui agissent comme des extensions au HTML ou au XML. Les bibliothèques de balises offrent une méthode indépendante de la plateforme pour étendre les fonctionnalités d'un serveur HTTP. Il existe aussi un langage de script particulier, appelé Expression Language (EL) destiné à réduire l'injection de code java au sein des pages JSP ainsi qu'à étendre les possibilités des taglibs, tel que la JSTL.

Les JSP sont compilées par un compilateur JSP pour devenir des servlets Java. Un compilateur JSP peut créer une servlet Java en code source Java qui peut à son tour être compilé par le compilateur Java, ou peut créer le pseudo-code Java interprétable directement. Dans les deux cas, il est bon de comprendre comment le compilateur JSP transforme la page en servlet Java. Voir l'exemple de page JSP fourni en fin d'article, avec la servlet créée et la page HTML résultante^[1].

II.4.4 Le JavaScript

JavaScript (quelques fois abrégé JS) est un langage de programmation de scripts, principalement utilisé dans les pages Web interactives. Il incorporé dans un document HTML et permet d'apporter des améliorations à ce langage est résoudre des problèmes que HTML ne sait pas résoudre. Le JavaScript permet d'exécuter des commandes du côté client, c'est-à-dire au niveau du navigateur et non du serveur Web, Il ne nécessite donc pas de compilateur.

Historiquement il s'agit même du premier langage de script pour le Web, il a été mis

^[1]https://fr.wikipedia.org/wiki/JavaServer_Pages

au point par Netscape en 1995, à l'origine, il se nommait Live Script et était destiné à fournir un langage de script simple au navigateur Netscape Navigator 2. Le langage JavaScript est fortement dépendant du navigateur appelant la page Web dans laquelle le script est incorporé, mais en contrepartie il ne nécessite pas de compilateur^[1].

5 Système de gestion de base de données

Le système de gestion de base de données ou SGBD: est un logiciel (ou un ensemble de logiciels) de haut niveau permettant de manipuler les données d'une base de données. C'est-à-dire sélectionner et afficher des informations tirées de cette base, modifier des données, ajouter ou supprimer des données (ce groupe de quatre opérations étant souvent appelé le CRUD, pour Create, Read, Update, Delete).

La plupart de SGBD sont basés sur un modèle Client - Serveur. C'est à dire que la base de données se trouve sur un serveur qui ne sert qu'à ça, et pour interagir avec cette base de données, il faut utiliser un logiciel "client" qui va interroger le serveur, et transmettre la réponse que le serveur lui aura donnée. Le serveur peut être installé sur une machine différente du client, et c'est souvent le cas lorsque les bases de données sont importantes. Les serveurs de base de données sont très souvent utilisés avec le langage de création de pages Web dynamiques : PHP.

Il existe de nombreux systèmes de gestion de base de données dont nous citons

MySQL et

Oracle.

II.5.1 MySQL

MySQL dérive directement de SQL (StructuredQueryLanguage) qui est un langage de requête utilisé pour interroger des bases de données exploitant le modèle relationnel. Il permet d'interagir avec une base de données relationnelle, pour lui ajouter des informations, en modifier, ou en rechercher. L'une des spécificités de MySQL est de pouvoir gérer plusieurs moteurs au sein d'une seule base. Chaque table peut utiliser un moteur différent au sein d'une base. Ceci afin d'optimiser l'utilisation de chaque table. Parmi les points forts de MySQL, on cite :

[1] <https://www.commentcamarche.net/contents/577-javascript-introduction-au-langage-javascript>

- ✓ Implémentation libre et populaire.
- ✓ Facile à mettre en œuvre.
- ✓ Rapide à apprendre.
- ✓ Support multi-plateforme.
- ✓ Fiable et rapide.

Et comme tout langage, il a aussi des points faibles et on cite :

- ✓ Ne possède pas de mécanisme transactionnel dans sa version 3.
- ✓ N'implémente pas les références d'intégrité relationnelles.
- ✓ Absence de procédures stockées et triggers.

II. 5.2 Oracle

Oracle : est un Système de gestion de base de données édité par la société du même nom, il est plus célèbre, plus complet et plus puissant. Mais malheureusement payant (et cher), ce qui le réserve plutôt aux entreprises qui l'utilisent déjà massivement. Il existe cependant des versions gratuites d'Oracle, notamment pour ceux qui veulent apprendre à s'en servir. Il est édité par la société du même nom (Oracle Corporation). Oracle assure des multiples fonctionnalités comme la définition, la manipulation des données et la cohérence des données, la confidentialité des données, ou bien la sauvegarde et la restauration des données.

II.6. Les navigateurs

Le navigateur Internet est un logiciel vous permettant d'afficher des sites Internet, télécharger des fichiers et faire des recherches. Microsoft Edge est le navigateur par défaut de Windows, mais il en existe d'autres comme Firefox, Google Chrome, Opera et Safari, chacun ayant ses propres fonctionnalités. Ils sont tous gratuits et téléchargeables sur Internet. Essayez-les tous .

Le navigateur ne connaît que les langages Html et JavaScript, si les pages demandées au serveur distant sont rédigées dans un autre langage (tel que PHP), c'est le serveur en question qui se chargera de les traduire et de les envoyer en Html ou en JavaScript à l'ordinateur client.

Il existe de nombreux navigateurs Web, pour toutes sortes de matériels (ordinateur personnel, tablette tactile, téléphones mobiles, etc.) et pour différents

systèmes d'exploitation (GNU/Linux, Windows).

Et parmi ces navigateurs on cite :

Firefox

Firefox est le navigateur de la fondation Mozilla, qui a été le premier concurrent de taille d'Internet Explorer. Le petit renard roux a su séduire les internautes par les innovations qu'il a apportées .

Google Chrome

Google Chrome le petit dernier arrivé. Le navigateur du géant de la recherche : Google. Ce navigateur est léger, très simple, idéal pour les débutants et les utilisateurs en recherche de simplicité et rapidité.

Internet Explorer

Internet Explorer est le navigateur Internet de Microsoft, intégré de base sur Windows. C'est le navigateur le plus utilisé, car bien souvent les gens ignorent qu'il y a d'autres alternatives.

Opera

Opera est un navigateur d'origine norvégienne, se démarque par ses innovations, notamment la navigation par onglets (reprise par les autres navigateurs). Opera se retrouve également beaucoup sur les appareils mobiles.

Safari

Safari est le navigateur Internet d'Apple, initialement présent sur les Mac. Une version Windows est aussi disponible. Il se veut simple et léger, facile d'utilisation. Un très bon navigateur sur les systèmes Mac.

Les navigateurs permettent d'accéder via un réseau à des documents repérés par une adresse unique, appelée URL.

Que ce qu'un URL ?

II.7. URL

Pour caractériser un document dans le monde de l'Internet, on parle d'une adresse Internet ou d'URL (Uniform Resource Locator). L'URL est l'identificateur utilisé par le navigateur pour localiser un document sur le réseau. Il est composée d'un nom de domaine précédé par le nom du protocole de communication utilisé pour dialoguer avec le serveur (tel que http:// pour le web, ftp:// pour le transfert de fichier, news : //

pour le système de news, etc..).

II. Conclusion :

Dans ce chapitre nous nous sommes particulièrement intéressé à prendre connaissance des différents outils utilisés dans la mise en place d'un site Web relatif à HTML, CSS, PHP..., ils nous ont été bien utile dans notre travail, et nous permettra de réaliser une application Web (Page Home) pour un chercheur.

Chapitre III

LA CONCEPTION

III. Introduction :

La conception de logiciel met en œuvre un ensemble d'activités qui à partir d'une demande d'informatisation d'un processus (demande qui peut aller de la simple question orale jusqu'au cahier des charges complet) permettent la conception, l'écriture et la mise au point d'un logiciel (et donc de programmes informatiques) jusqu'à sa livraison au demandeur.

On effectue simultanément l'étude des données et l'étude des traitements à effectuer. C'est en général dans cette phase que s'appliquent les techniques de modélisation. Il en découle la description des bases de données éventuelles à créer et les programmes à écrire et la manière dont tout cela va être intégré.

Nous allons commencer ce chapitre par la présentation de la méthodologie d'analyse, en justifiant la raison du choix de cette méthodologie pour notre application, nous allons montrer ensuite les acteurs de l'application et leurs rôles, ainsi que les différents diagrammes UML^[1].

^[1]https://fr.wikipedia.org/wiki/Conception_de_logiciel

III.1. Le langage UML

III.1.2 Définition

Le Langage de Modélisation Unifié, de l'anglais Unified Modeling Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet^[2].

III.2. Modélisation

III.2.1 Diagramme de cas d'utilisation (Use case)

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel. Ils sont utiles pour des présentations auprès de la direction ou des acteurs d'un projet, mais pour le développement, les cas d'utilisation sont plus appropriés. Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur (humain ou machine) et un système. Il est une unité significative de travail. Dans un diagramme de cas d'utilisation, les utilisateurs sont appelés acteurs (actors), ils interagissent avec les cas d'utilisation (use cases)

L'objectif de ce diagramme est d'identifier les besoins du client pour rédiger le cahier de charge.

Principe : Définir les limites du système. Définir l'environnement du système : les utilisateurs qui interagissent avec le système^[3].

[2] [https://fr.wikipedia.org/wiki/UML_\(informatique\)](https://fr.wikipedia.org/wiki/UML_(informatique))

[3] https://fr.wikipedia.org/wiki/Diagramme_de_classes

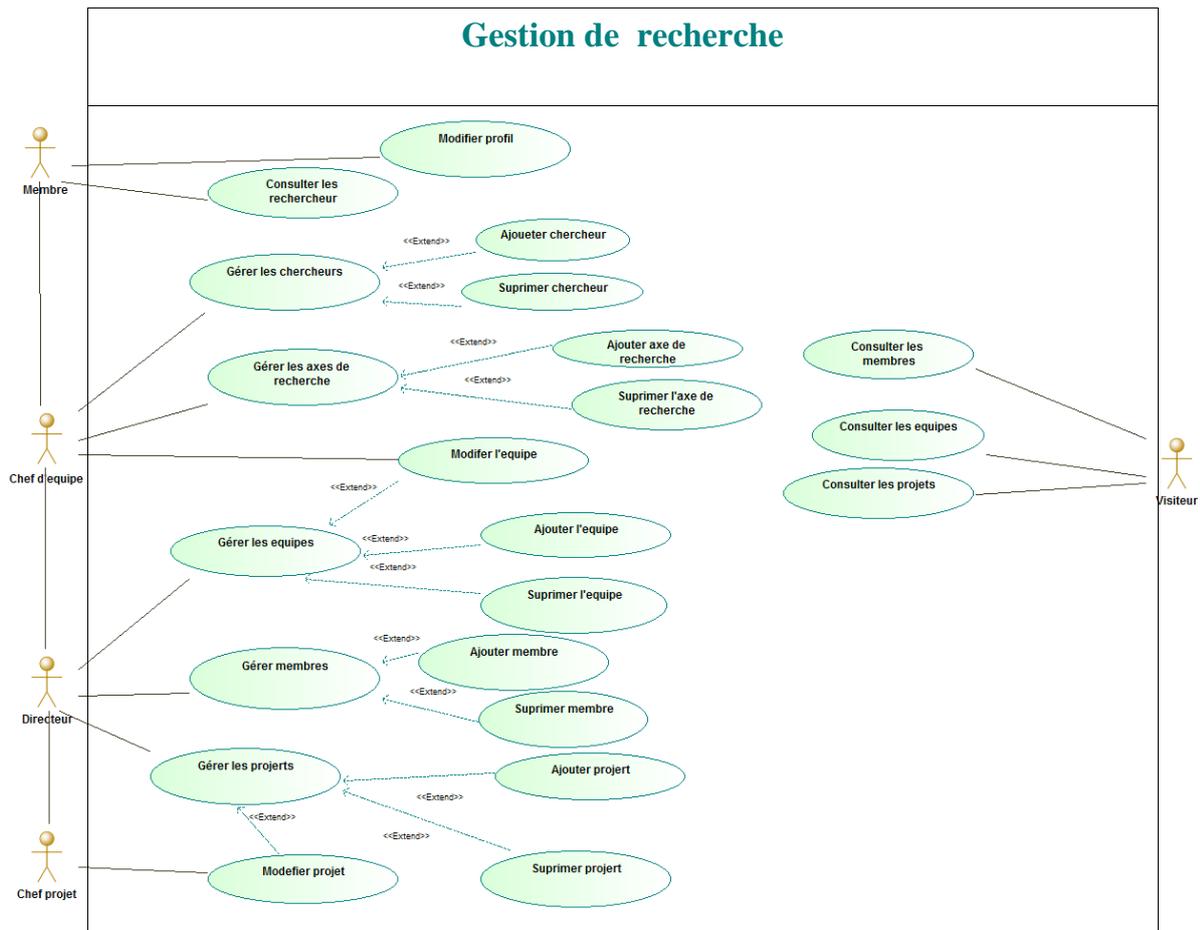


Figure III.1 : Diagramme de cas d'utilisation

III. 2.2 Les acteurs et leurs rôles

Ils sont des entités externes qui interagissent avec le système, comme une personne humaine ou un robot. Une même personne (ou robot) peut être plusieurs acteurs pour un système, c'est pourquoi les acteurs doivent surtout être décrits par leur rôle, ce rôle décrit les besoins et les capacités de l'acteur. Un acteur agit sur le système. L'activité du système a pour objectif de satisfaire les besoins de l'acteur.

Dans notre système les acteurs et ses rôles sont suivant :

- ★ **Directeur** : s'authentifier, créer équipe, créer projet, créer les thèmes de laboratoire ajouter membre, modifier profil.
- ★ **Chef d'équipe** : s'authentifier, modifier équipe, ajouter chercheur dans équipe, gérer les axes de recherche.
- ★ **Chef de projet** : s'authentifier, modifier projet, ajouter chercheur dans projet.

- ★ **Membre** : s'authentifier, modifier profil, ajouter un axe de recherche, consulter les chercheurs.

III. 2. 3 Diagramme de classes

Une classe est un ensemble de fonctions et de données (attributs) qui sont liées ensemble par un champ sémantique. Les classes sont utilisées dans la programmation orientée objet. Elles permettent de modéliser un programme et ainsi de découper une tâche complexe en plusieurs petits travaux simples.

Les classes peuvent être liées entre elles grâce au mécanisme d'héritage qui permet de mettre en évidence des relations de parenté. D'autres relations sont possibles entre des classes (association, agrégation, composition, dépendance) chacune de ces relations est représentée par un arc spécifique dans le diagramme de classes^[4].

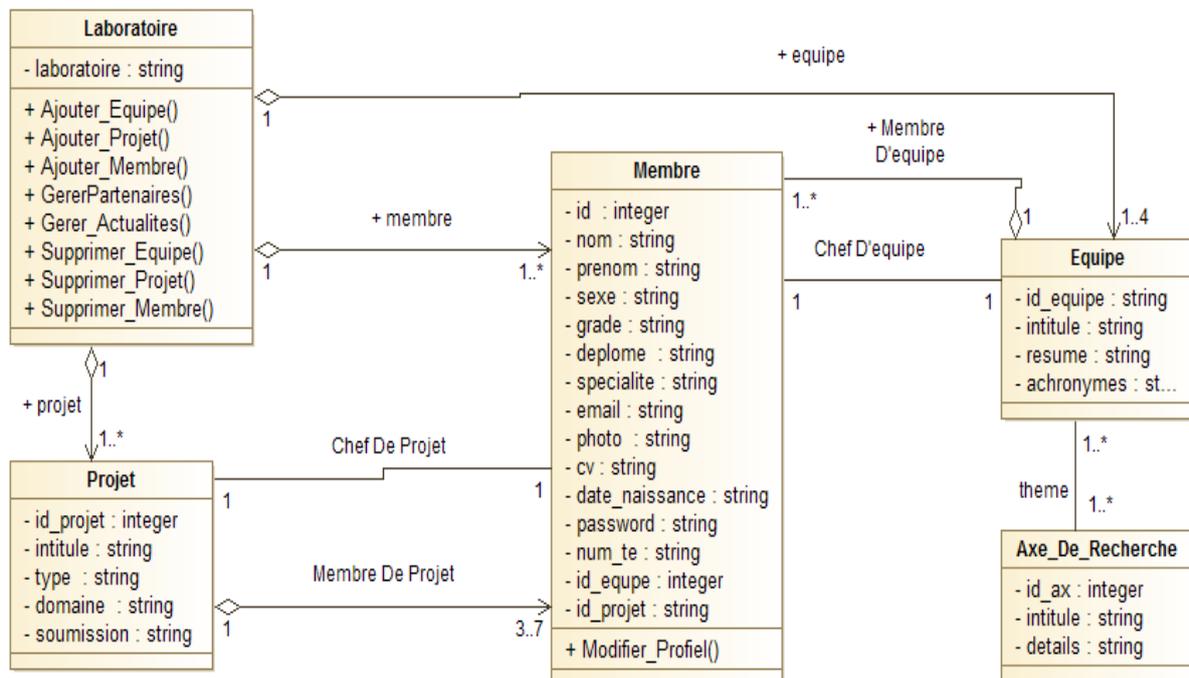


Figure III.2: Diagramme de classe

[4] https://fr.wikipedia.org/wiki/Diagramme_de_classes

III. 2.4 Schéma relationnel

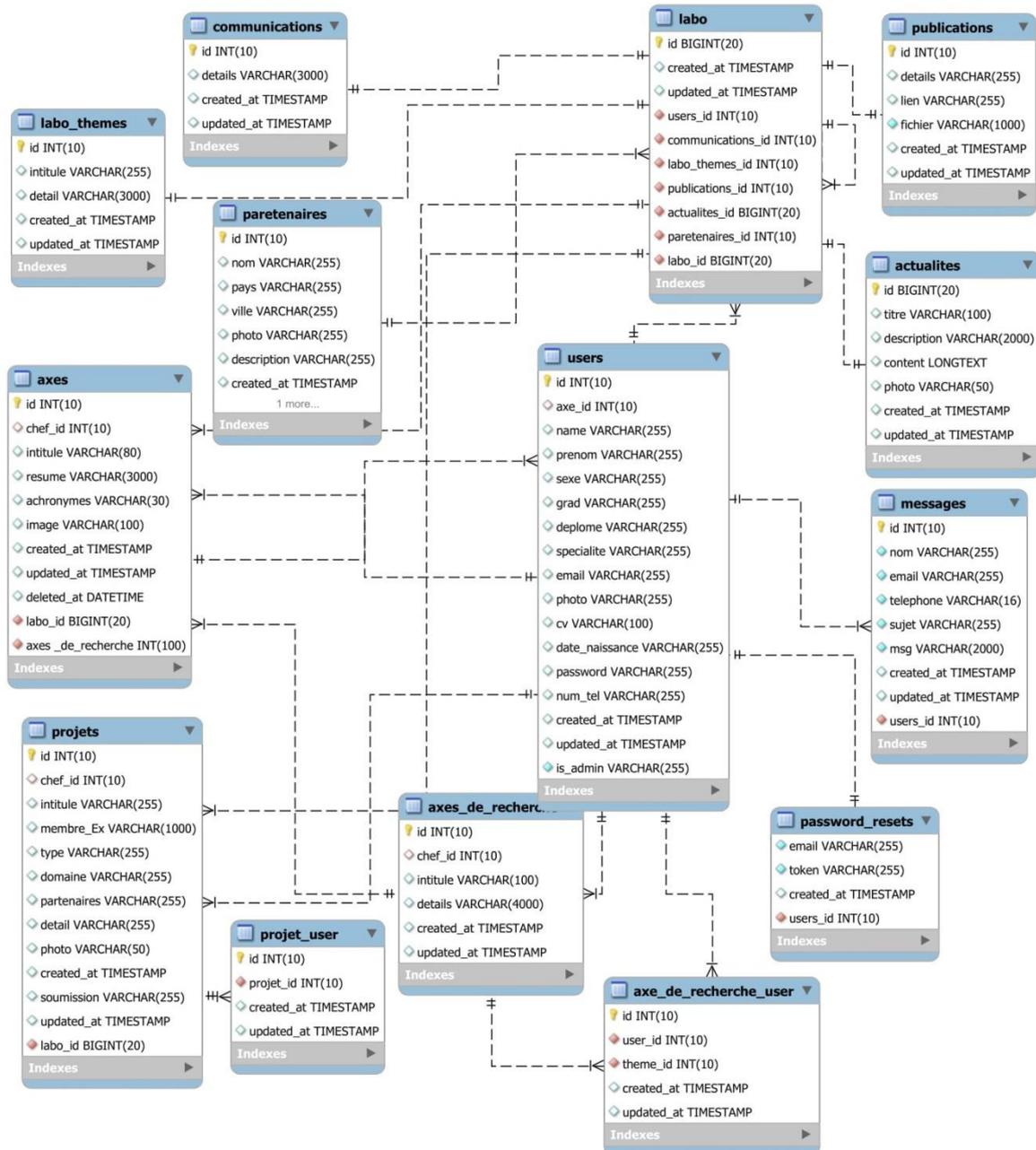


Figure III.3 : Base de données

Cette figure représente toutes les tables de la base de données indiquant les relations entre les données stockées.

III. 2. 5 Diagrammes de séquences

C'est la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique.

Le diagramme de séquence permet de montrer les interactions d'objets dans le cadre d'un scénario d'un Diagramme des cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite du système. Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets.

La dimension verticale du diagramme représente le temps, permettant de visualiser l'enchaînement des actions dans le temps, et de spécifier la naissance et la mort d'objets. Les périodes d'activité des objets sont symbolisées par des rectangles, et ces objets dialoguent à l'aide de messages.

Voilà quelques diagrammes importants^[5] .:

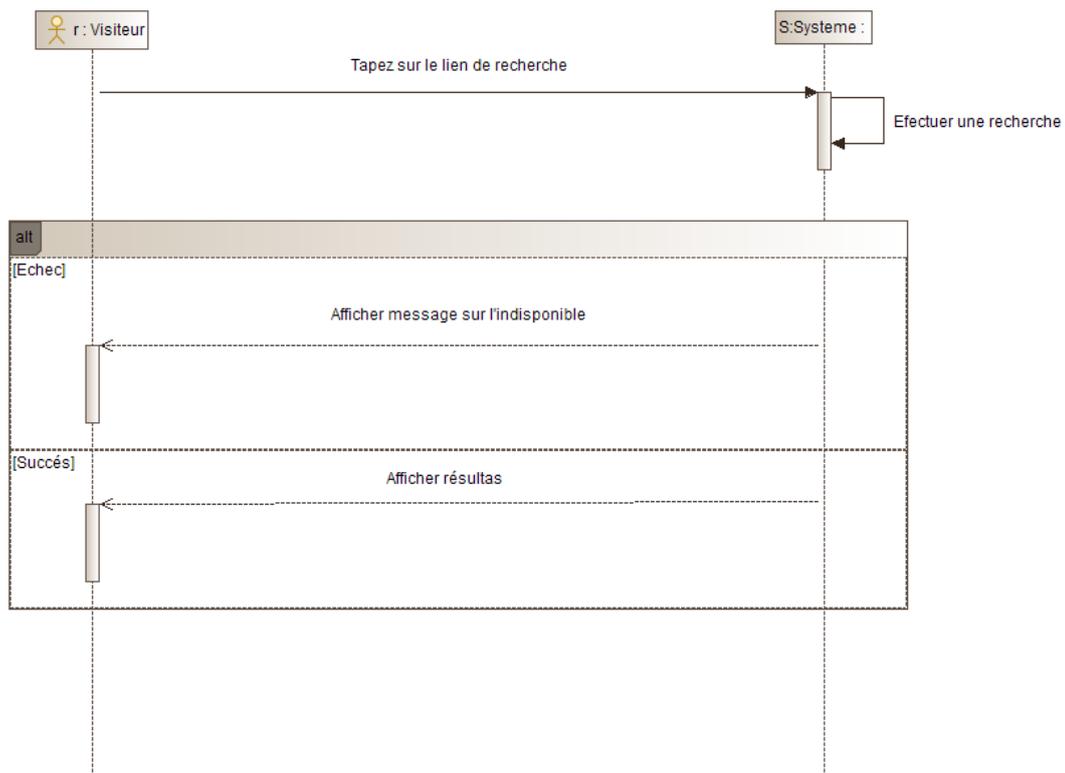


Figure III.4 : Diagramme de séquence de cas d'utilisation « Consulter le site »

[5] https://fr.wikipedia.org/wiki/Diagramme_de_s%C3%A9quence

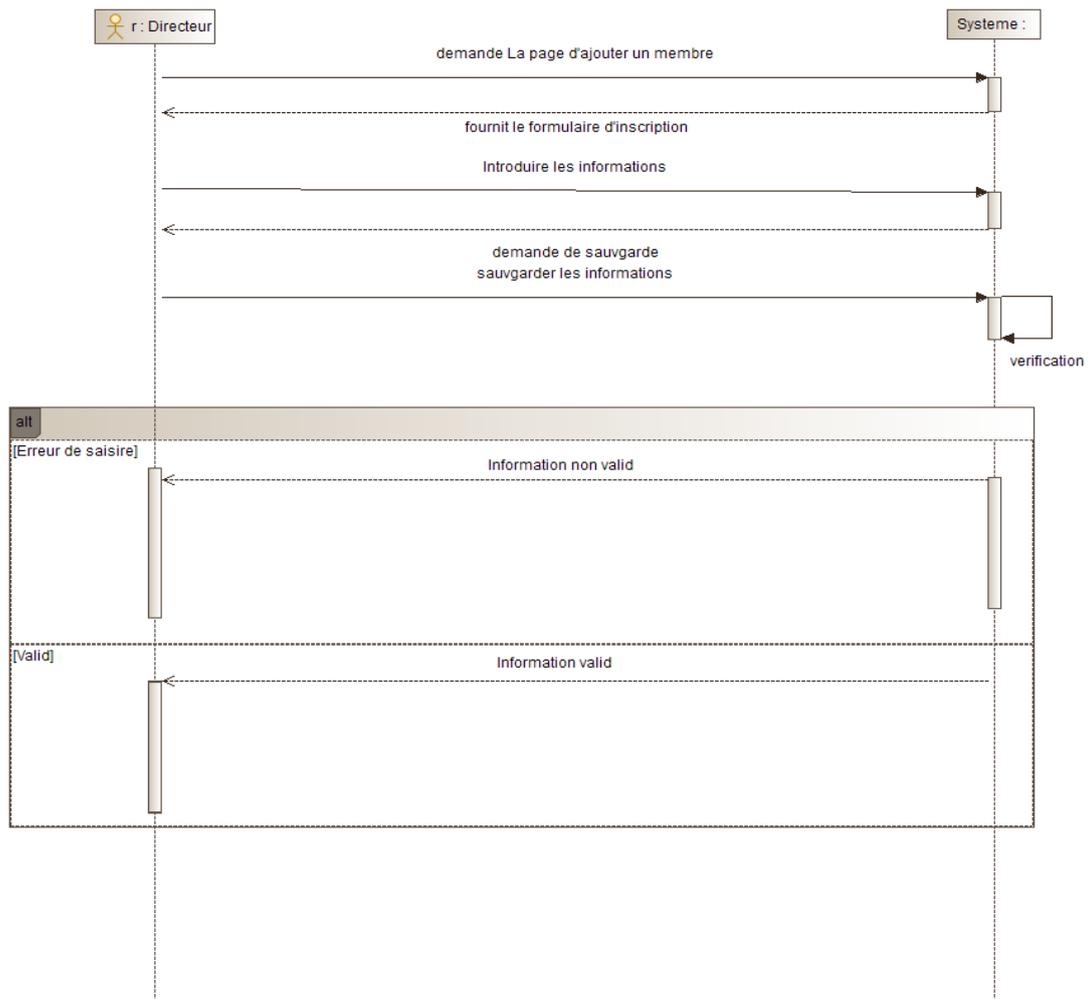


Figure III.5 : Diagramme de séquence de cas d'utilisation «traitez une demande d'inscription »

III. 3. Maquettes (Design)



Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Centre Universitaire BELHADJ Bouchaib - Ain Témouchent

Laboratoire de Structures Intelligentes "SSL"



[Accueil](#) [Equipes](#) [Actualités](#) [Publications](#) [Projets](#) [Thèmes](#) [Contacts](#)

[Membres](#) [Se Connecter](#)



Figure III.6: la page d'accueil de laboratoire de recherche

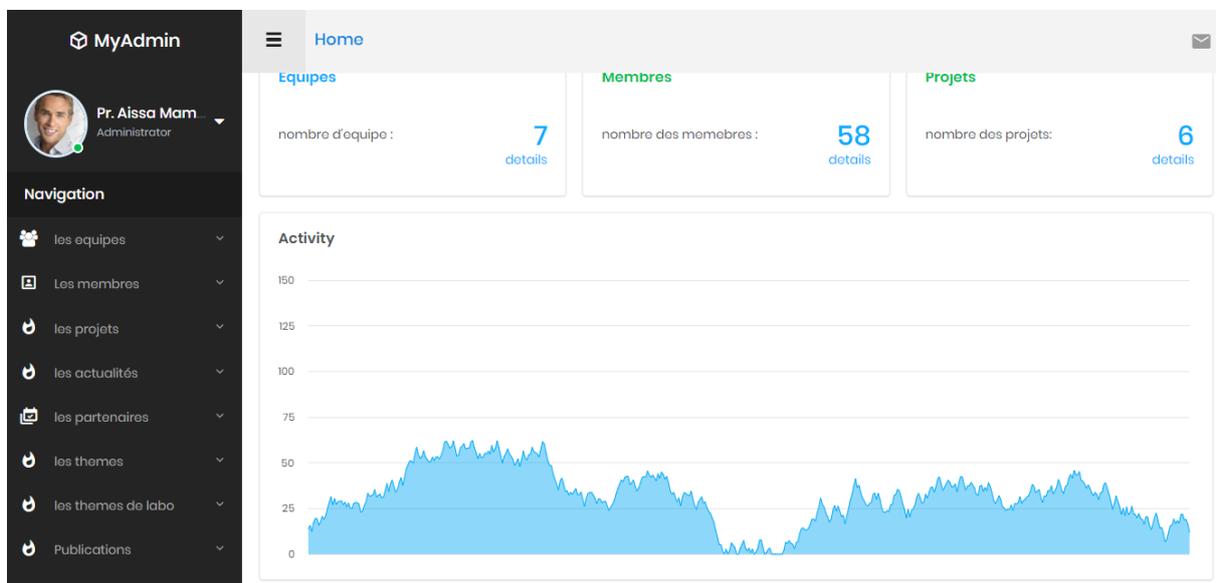


Figure III.7: la page d'un membre de laboratoire de recherche

The screenshot shows a web interface for creating a new team. On the left is a dark sidebar with the user profile 'Pr. Aïssa Mam' and a navigation menu. The main content area is titled 'Nouvelle équipe' and contains the following form fields:

- Chef de l'équipe:** A dropdown menu.
- intitule:** A text input field containing 'intitule'.
- resumo:** A large text area containing 'Write your resume'.
- Achronyme:** A text input field containing 'Achronyme'.
- Photo:** A file selection button labeled 'Choisir un fichier' and the text 'Aucun fichier choisi'.

A blue 'save' button is located at the bottom left of the form.

Figure III.8 : le formulaire d'ajouter une équipe

The screenshot shows the 'EQUIPES' page. At the top is a navigation bar with links: Accueil, Equipes, Actualités, Publications, Projets, Thèmes, Contacts, Membres, and Se Connecter. Below the title 'EQUIPES :', there is a grid of team cards:

- TH:** Thermofluide. Image: AR logo.
- MAI:** Matériaux avancés et intelligents. Image: Diagram of physical layer communication.
- CRS:** Conception et réalisation des structures. Image: Communication scheme diagram.
- SEA:** 1.18000/Publication de Systèmes d'Electronique Appliqués. Image: Satellite in space.
- DO:** Durabilité des ouvrages. Image: Circuit board.
- EMT:** Endommagement mécanique et tribologie. Image: Brain with circuitry.

Figure III.9: la liste des équipes

CHEF D'EQUIPE:

Pr NEHARI Driss

TH : THERMOUFLUIDE

L'équipe de thermofluide est organisée autour de trois thèmes de recherche : dynamique des Fluides

AXES DE RECHERCHE :

- > Développement des nouvelles méthodes pour le traitement et la transmission des images.
- > Développement des stratégies
- > d'optimisation des communications sans fil
- > Développement des méthodes de transmission d'images robuste et qui résiste aux erreurs de canal.
- > Développement des méthodes de restauration d'images.
- > Développement des méthodes de compression et d
- > Conception de réseaux d'antennes imprimées

MEMBRE D'EQUIPE

Dr. NEHARI Taieb



M. BENLEKKAM Med



MCA-HASNI Abdelhafid



M. EL MERIAH A.



M. BOUROUBA Yacine



M. RAMLAOUI Ahmed



M. BENAMAR Abdelaziz

Figure III.10 : le détail de chaque équipe

CONTACTEZ-NOUS

Etablissement	Centre Universitaire Belhadj Bouchaib – Ain Temouchent
Telephone	+213 (0) 43 79 84 31
Fax	FAX :049.81.55.81/91
E-Mail	labsmart46@gmail.com



Figure III.11 : le contact de laboratoire de recherche

III. Conclusion :

Dans ce chapitre nous avons présenté la Conception pour réaliser notre application comme nous avons présenté quelques interfaces.

Conclusion Générale

Conclusion Générale :

Le travail effectué dans ce mémoire s'inscrit dans le cadre de la conception et la réalisation d'un site web dynamique.

Notre application offre aux visiteurs la possibilité de plusieurs utilisations, tels que la Consultation, la recherche d'informations, ...etc. Avec une simplicité de manipulation des fonctionnalités pour la satisfaction des internautes.

Notre projet a été une occasion qui nous a permis d'approfondir nos connaissances informatiques en maîtrisant de nouveaux outils

Et enfin Comme perspectives, nous espérons :

D'étendre ce projet pour tout le laboratoire de recherche.

D'ajouter d'autres fonctionnalités avancées dans la recherche d'information de notre site.

Bibliographie

Les Articles

- 1- [Journal officiel de la république algérienne, article N° 77, 1999, p4, p5, p6](#)
(01/05/2019)

Les sites web

- 2- <http://laravel.sillo.org/cours-laravel-5-5-les-bases-presentation-generale/>
(12/05/2019)
- 3- <https://getcomposer.org/doc/00-intro.md>(12/05/2019)
- 4- <http://laravel.sillo.org/cours-laravel-5-5-les-bases-installation-et-organisation/>
(12/05/2019)
- 5- <http://laravel.sillo.org/cours-laravel-5-5-les-bases-le-routage/> (12/05/2019)
- 6- <http://laravel.sillo.org/cours-laravel-5-3-les-bases-artisan-et-les-controleurs/>
(12/05/2019)
- 7- <http://laravel.sillo.org/cours-laravel-5-3-les-donnees-migrations-et-modeles/>
(12/05/2019)
- 8- https://fr.wikipedia.org/wiki/Hypertext_Markup_Language(17/05/2019)
- 9- https://fr.wikipedia.org/wiki/Hypertext_Markup_Language#%C3%89volution_du_langage (17/05/2019)
- 10- <https://openclassrooms.com/fr/courses/1603881-apprenez-a-creer-votre-site-web-avec-html5-et-css3/1608357-memento-des-balises-html> (17/05/2019)
- 11- https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade (17/05/2019)
- 12- https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade#Vers_une_maturati_on_difficile (17/05/2019)
- 13- <https://www.php.net/manual/fr/history.php> (17/05/2019)
- 14- https://fr.wikipedia.org/wiki/JavaServer_Pages (17/05/2019)
- 15- <https://www.commentcamarche.net/contents/577-javascript-introduction-au-langage-javascript> (17/05/2019)
- 16- https://fr.wikipedia.org/wiki/Conception_de_logiciel(20/05/2019)
- 17- [https://fr.wikipedia.org/wiki/UML_\(informatique\)](https://fr.wikipedia.org/wiki/UML_(informatique)) (20/05/2019)
- 18- https://fr.wikipedia.org/wiki/Diagramme_de_classes (20/05/2019)
- 19- https://fr.wikipedia.org/wiki/Diagramme_de_classes (20/05/2019)
- 20- https://fr.wikipedia.org/wiki/Diagramme_de_s%C3%A9quence (20/05/2019)

الملخص:

مختبر الأبحاث مسؤول عن تنفيذ موضوع أو أكثر من مواضيع البحث العلمي والتطوير التكنولوجي. مهماتها هي:

- تحقيق أهداف البحث العلمي والتطوير التكنولوجي في مجال علمي محدد .
- المساهمة في تطوير برامج البحوث في مجال أنشطتها .
- المساهمة في اكتساب المعرفة العلمية والتكنولوجية الجديدة وإتقانها وتطويرها.
- الهدف من مشروعنا هو تسهيل العمل وتحقيق موقع إلكتروني لإدارة المختبرات البحثية وتناولنا حالة.

الكلمات المفتاحية: موقع إلكتروني، مخبر التحاليل، لارافل، المخططات

Résumé :

Un laboratoire de recherche est chargé de la mise en œuvre d'un ou de plusieurs thème de recherche scientifique et de développement technologique.

Il a pour missions :

- Réaliser des objectifs de recherche scientifique et de développement technologique dans un domaine scientifique précis ;
- Contribué à l'élaboration des programmes de recherche dans le domaine de ses activités ;
- Contribuer à l'acquisition, à la maîtrise et au développement de nouvelles connaissances scientifiques et technologiques ;

L'objectif de notre projet facilité le travail et de réaliser un site web pour facilité la gestion d'un laboratoire de recherche et nous avons pris un cas réel c'est le laboratoire de recherche Structures Intelligentes "SSL"

Mots-clés: site web, laboratoire d'analyse, larvaire, graphiques

Abstract

The research laboratory is responsible for the implementation of one or more themes of scientific research and technological development.

His tasks are:

- Research to achieve the objectives of scientific research and technological development in a specific scientific field;*
- Contribute to the development of research programs in the field of their activities;*
- Contribute to the acquisition, mastery and development of new scientific and technological knowledge;*

The purpose of our project is to facilitate the work and to create a website for the management of research laboratories. The real case is the research laboratory "Intelligent Structures".

Keywords: website, analysis lab, larval, charts