

---

# CPE - TIDAL

*Techniques de l'Internet Dynamique et Architecture Logicielle*

## Travaux pratiques : sujet fil rouge

2021 - Yannick Joly

---

### Résumé des instructions

#### Approche guidée

1. **Ébauche du site (maximum 4h)** : lecture du cahier des charges, réalisation de **maquettes** en dessin puis ébauche en HTML/CSS (sera peaufiné au fur et à mesure par la suite).
2. **Passage par le templating (environ 4h)** : intégration des maquettes à une base PHP par le templating.
3. **Dynamisation du site (environs 6h)** : architecture du PHP (gestion des formulaires, PHP objet, accès à la base de données).
4. **Mise en place d'une API REST (au moins 4h)** : côté serveur et consommation côté client (Javascript).

#### Approche libre

Si vous avez des bases en web dynamique, sentez-vous libres d'aborder le sujet dans l'ordre que vous voulez. L'objectif est d'apprendre et de progresser, mais aussi de renforcer les bases que vous avez. Mais ça ne vous dispense pas de lire l'ensemble du sujet !

Dans tous les cas, discutez d'abord de votre organisation avec un enseignant, qui pourra vous aider à déterminer quels objectifs et quelle organisation vous fixer et que ce soit en rapport avec les objectifs pédagogiques (cf fiche d'auto-évaluation).

Pour les plus à l'aise, il faudra par exemple essayer de concevoir et réaliser une API RESTful, et de reposer l'intégralité des interfaces sur cette API.



## Règles pour tous

1. Aucune bibliothèque, aucun framework n'est autorisé (ni en Javascript, ni côté Serveur en PHP), en dehors du moteur de template.
2. Pour le moteur de template : Smarty ou Twig sont fortement conseillés. Pour tout autre moteur, demander la validation d'un enseignant.
3. Pas d'attente forte côté HTML/CSS : utilisation autorisée de bibliothèques externes (Bootstrap, etc.), pas besoin de faire quelque chose de joli. Il faut juste respecter les standards et fournir un code valide et accessible (utilisez des validateurs de HTML, de CSS et d'accessibilité !)

# 1. Cahier des charges

## 1.1. Motivations pédagogiques sur le choix du sujet d'étude

Le sujet d'étude que nous vous proposons pourra vous sembler étrange : il s'agira en effet de travailler pour une association d'acupuncteurs en médecine traditionnelle chinoise.

Dans votre vie professionnelle, il vous sera très souvent demandé en tant qu'informaticien de proposer des solutions pour des sujets d'étude dont vous ignorez complètement les périmètres. Aussi, il nous a semblé pertinent de vous confronter à un problème dont le sujet d'étude sera pour la très grande majorité (voire la totalité) d'entre vous complètement nouveau.

Sachez toutefois que les données que vous utiliserez sont des données réelles.

Certains aspects de la base de connaissance ont toutefois été simplifiés, afin de ne pas complexifier outre mesure la prise en main du contexte d'étude.

Un autre intérêt est que ce problème a réellement été posé : même si le cahier des charges a été (très) allégé, il correspond à une vraie demande de la part de vrais acupuncteurs.

Enfin, que l'on y croit ou non, de nombreux symptômes ainsi que leurs associations sont plutôt amusants voire parfois déroutants. Cela vous encouragera à vérifier l'exhaustivité de vos requêtes... et sera peut-être un moyen de vous faire sourire en TP et/ou de détendre l'atmosphère !

## 1.2. Présentation générale du sujet d'étude

L'association des acupuncteurs soucieux de l'accessibilité (AAA...) vous a sollicité pour la conception et la réalisation d'une plateforme en ligne dont les principales fonctionnalités seraient :

1. de disposer d'un service en ligne leur permettant de consulter la liste des symptômes des principales pathologies en acupuncture (voir **table 1.2.a**) ;
2. de pouvoir n'afficher que certaines des pathologies en fonction de différents critères (type de pathologie, choix des méridiens<sup>1</sup>, etc., voir la **table 1.2.b**) ;
3. de rechercher les pathologies comportant certains symptômes.

---

<sup>1</sup> un méridien en acupuncture est un chemin composé de différentes branches lié à un organe, à un viscère ou à un « merveilleux vaisseau ». Sa partie superficielle contient les fameux « points » d'acupuncture que les praticiens poncturent, chauffent ou massent. Les différentes branches des chemins n'empruntent pas forcément des trajets anatomiques ou physiologiques, ce qui est l'un des mystères que la médecine occidentale n'a pas encore pu expliquer scientifiquement.

Catégorie de pathologie	Caractéristiques possibles	Exemples
Pathologies de méridien	<ul style="list-style-type: none"> <li>interne</li> <li>externe</li> </ul>	<ul style="list-style-type: none"> <li>méridien du poumon interne</li> <li>méridien du rein externe</li> </ul>
Pathologies d'organe/viscère (tsang/fu)	<ul style="list-style-type: none"> <li>plein</li> <li>vide</li> <li>chaud</li> <li>froid</li> </ul>	<ul style="list-style-type: none"> <li>poumon vide</li> <li>poumon froid</li> <li>rate vide et froid</li> <li>foie chaud</li> </ul>
Pathologies des tendino-musculaires (jing jin)	néant	jing jin du rein
Pathologie des branches (voies luo)	<ul style="list-style-type: none"> <li>vide</li> <li>plein</li> </ul>	<ul style="list-style-type: none"> <li>voie luo du poumon vide</li> <li>voie luo du rein pleine</li> </ul>
Pathologies des merveilleux vaisseaux	néant	pathologie du Ren Mai

Table 1.2.a – Types de pathologies en acupuncture. Il existe d'autres types de pathologies qui ne seront pas utilisées dans l'application, et qui ne figurent pas dans la base de connaissances fournie.

Critère	Valeurs possibles	Exemples
Méridien	Nom du méridien (20 méridiens en tout)	Poumon, Ren Mai, rein, Yanq Qiao Mai
Type de pathologie	méridien, organe/viscère, luo, merveilleux vaisseaux, jing jin	sélectionner les pathologies voie luo
Caractéristiques	plein, chaud, vide, froid, interne, externe	sélectionner les pathologies de vide

Table 1.2.b – Critères pour les filtres. Les filtres peuvent se combiner : sélectionner les pathologies de méridien, interne, pour poumon et foie.

### 1.3. Contraintes techniques

#### 1.3.1. Accessibilité et versions **HTML**

- L'ensemble du site respectera les normes d'accessibilité de niveau AAA.
- Il sera écrit en XHTML1.1 ou HTML 5.
- De plus, toute page du site devra être accessible de la page d'accueil en 3 interactions au maximum (clic, raccourci, etc.).
- La navigation devra être possible sans souris ou sans clavier.



### 1.3.2. Environnement d'exécution

Le site devra être affichable et utilisable à partir d'une version de Firefox récente.

La partie dynamique (serveur) devra fonctionner sur une architecture **Linux/Apache/PHP/PostgreSQL**.

### 1.3.3. Base de connaissances

La base de connaissances vous est fournie sous la forme d'un script SQL de génération de base de données.

Vous ne pouvez en aucun cas modifier les tables de cette base, mais vous êtes libres d'ajouter de nouvelles tables si vous en avez besoin.

La base de connaissance a été réalisée à partir du Vademecum d'acupuncture traditionnelle de Jean Motte [Motte, 2014].

### 1.3.4. Graphisme

La partie graphique doit avant tout éviter les erreurs d'accessibilité et d'ergonomie.

Il ne vous est pas demandé de faire un site esthétiquement joli, mais simple et fonctionnel.

S'il vous reste du temps en fin de module, vous pourrez améliorer cet aspect à ce moment-là.

## 1.4. Contraintes de réalisation

Toutes les pages doivent être consultables en basse résolution et disposer d'une mise en page spéciale pour l'impression.

### 1.4.1. Page d'accueil

La page d'accueil doit disposer d'un menu et d'un formulaire d'identification pour les utilisateurs.

### 1.4.2. Autres pages

Vous êtes totalement libre de l'organisation des autres pages (nombre, architecture, etc...).

Vous réaliserez cependant une page d'information présentant les développements que vous avez réalisés, vos sources, les auteurs, ainsi que les ressources bibliographiques et la webographie que vous avez utilisé.



## 1.5. Fonctionnalités à implémenter

### 1.5.1. Consultation des pathologies, critères de filtrage

Vous réaliserez une page permettant d'afficher la liste des pathologies.

Ces pathologies pourront faire l'objet de filtrage comme indiqué en introduction.

Vous vous efforcerez d'optimiser au maximum votre solution en limitant autant que possible les requêtes sur le serveur MySQL.

Vous êtes totalement libres dans le choix de l'interface graphique.

### 1.5.2. Recherche de pathologies par mot-clé

Vous implémenterez une fonctionnalité de recherche de pathologie par mot-clé.

Les mot-clefs sont associés aux symptômes dans la base de connaissance.

Cette fonctionnalité ne sera accessible qu'aux **utilisateurs authentifiés**.

### 1.5.3. Compte utilisateur

Vous proposerez un système de gestion des utilisateurs (inscription, login, session, etc.).

Un utilisateur connecté aura la possibilité d'accéder à la fonctionnalité de recherche de pathologies par mot-clé (cf. **1.5.2**).

## 2. Première ébauche statique du site

**Durée conseillée : 4h**

### 2.1. Méthodologie

La méthodologie a été présentée en cours. Vous pouvez si vous le souhaitez l'adapter à vos besoins, en imaginer une qui vous parle plus, etc.

Quoi qu'il en soit, les exigences resteront les mêmes !

Les informations techniques dont vous avez besoin sont sur internet...

N'hésitez cependant pas à faire valider ce que vous avez trouvé et **maintenez votre webographie**.

### 2.2. Travail à réaliser

1. **(2.1) Constituez-vous en binôme** Il est interdit de travailler seul sur le projet fil rouge, sauf contre indication du responsable du module et raison valable (absences aux premiers TP principalement).  
En cas de nombre impairs d'étudiants dans le groupe, seul un groupe sera autorisé à travailler à trois.
2. **(2.2) Lisez le cahier des charges !** Vous pouvez vous arrêter à la liste des premières fonctionnalités incluse.
3. **(2.3) En appliquant la méthode vue en cours, ou une méthode équivalente, réalisez les maquettes « papier-crayon » des pages du site.** Vous penserez notamment à prévoir les réactions aux événements, et prendrez soin de découvrir les fonctionnalités nécessaires mais non exprimées directement par le client 1
4. **(2.4) Réalisez les maquettes statiques (en HTML et CSS) de vos pages.** Vous réaliserez notamment les formulaires, qui seront pour le moment passifs (il n'y aura pas d'action côté serveur lorsqu'ils seront validés).  
Vous pouvez cependant déjà prévoir et réaliser des contrôles javascript.

5. (2.5) Associez les feuilles css demandées

**! Attention aux pertes de temps !**

Nous ne vous demandons pas de maîtriser en quelques heures HTML et CSS !

Le temps passe souvent vite quand on cherche à régler de petits détails. C'est formateur, mais il ne faut pas oublier que vous êtes en temps limité.

Profitez donc au maximum de la présence des enseignants qui pourront vous dépanner ! Cherchez d'abord une solution, et si vous ne trouvez rien au bout de 15 minutes, sollicitez-les !

### 2.3. Ne pas oublier...

- de citer vos sources
- de vérifier que vous avez le droit d'utiliser telle ou telle ressources
- de bien respecter TOUS les termes des licences (citation explicite, etc.)
- de maintenir votre webographie...



### 3. Projet : Dynamisation de site avec PHP

L'objectif de ce TP est de rendre votre site dynamique en utilisant **PHP**.

**Durée conseillée** : 4h

Les indications seront données dans le cas d'une architecture **LAMP** uniquement.

#### **Faire du vrai Modèle Vue Contrôleur (MVC) avec les templates (ou du meilleur en tous cas !)**

Vous utiliserez un système de templating pour votre site. Nous vous conseillons d'utiliser Smarty (sera présenté dans un point de cours) ou Twig, qui sont des références dans le domaine.

#### 3.1. Travail à réaliser

1. **(1.1) Mettez en place une architecture dynamique pour votre site.** Vous veillerez à anticiper l'organisation physique des scripts et des ressources sur votre serveur (arborescence, droits, etc.).  
Consultez l'exemple donné dans l'archive du TP (**projet/**).
2. (1.2) Réalisez les différents scripts php dont vous avez besoin. Vous veillerez à vous rapprocher le plus possible d'une architecture MVC.
3. (1.3) Maintenez l'accessibilité de votre site...

#### 3.2. Formulaires HTML et PHP Objet

**Durée indicative** : 2h

##### 3.2.1. Objectifs

L'objectif est d'utiliser des formulaires pour :

- envoyer des paramètres pour construire des pages dynamiques sur les pathologies (filtrage, recherche...);
- gérer la connexion de l'utilisateur.

Ces formulaires seront accessibles (HTML, CSS et ARIA).

Ils pourront utiliser AJAX pour rendre l'utilisation du site plus conviviale et ergonomique.

### 3.3. Bases de données et PHP

**Durée indicative** : 2h

L'objectif du TP est d'utiliser la base de données mysql fournie par le client (voir modèle **figure 3.3.2.a**).

#### 3.3.1. Création de la base

- ~~• créez une base de donnée avec phpmyadmin (interclassement : utf8mb4\_unicode\_ci)~~
- ~~• créez un utilisateur avec les droits d'écriture et un utilisateur avec les droits de lecture seulement~~
- importez la base de donnée avec le script **projet/bd/acuBD.sql**

#### 3.3.2. La librairie de référence pour les BDD en PHP : PDO

Pensez à réaliser :

- une classe pour représenter une pathologie
- des requêtes préparées
- des connexions sécurisées
- vous protéger contre les principales attaques « faciles », notamment l'injection SQL...



#### **Ressources**

sur les bases de données et PDO :

<http://studio.jacksay.com/tutoriaux/php/connection-mysql-avec-pdo>

protection contre injection : <http://php.net/manual/fr/pdo.quote.php>

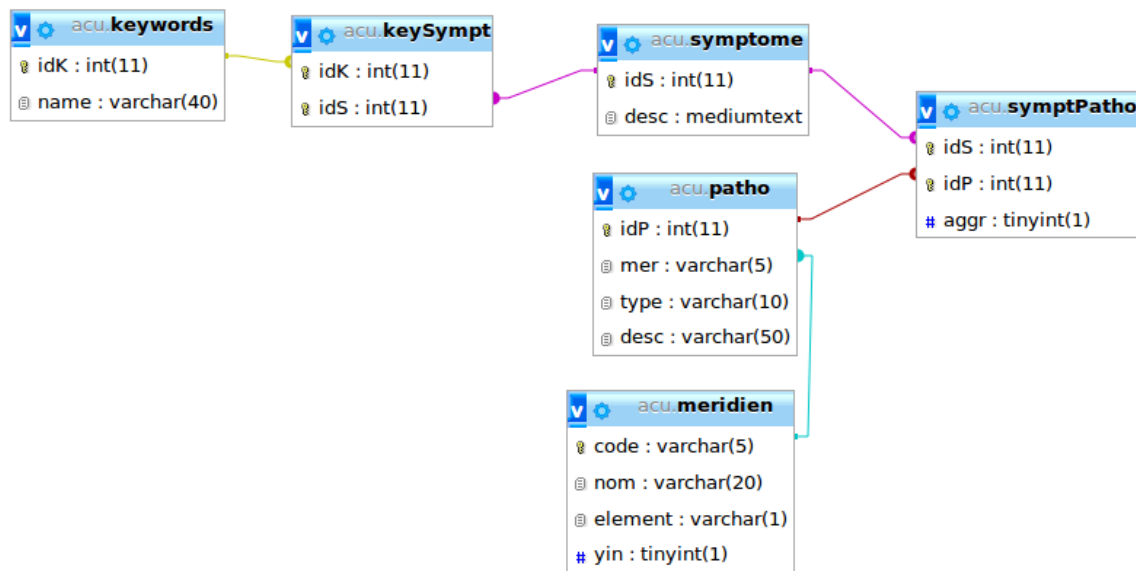


Figure 3.3.2.a – Schéma de la base de donnée des acupuncteurs

## 4. Web Services et ressources REST

### 4.1. Objectifs

Réalisez un web service respectant l'architecture REST (cf point de cours associé) et fournissant des données de pathologies avec filtrage sur le même modèle que la page web réalisée (cf 1.1.5).



#### Ressources

- rewriting avec apache : <http://www.urlrewriting.fr/tutoriel-reecriture.htm>
- site sur les expressions régulières : <http://www.regular-expressions.info/>
- tutoriel sur les expressions régulières : <http://regexone.com/rewriting>
- en PHP : <http://www.urlrewriting.fr/tutoriel-urlrewriting-sans-moteur-rewrite.htm>
- utilisation de Simple Object Access Protocol (SOAP) avec PHP : la librairie NuSoap <http://sourceforge.net/projects/nusoap/> et un tutoriel : <http://vivien-brissat.developpez.com/tutoriels/php/soap/>

#### 4.1.1. Première technique : URL rewriting côté serveur web

##### 4.1.1.1. Configuration du serveur Apache (mod\_rewrite)

1. activer le module rewrite : `a2enmod rewrite`
2. vérifier sa présence dans la liste des modules activés (répertoire `mod_enabled` de apache2)
3. autoriser la redirection dans la configuration du site actif :

```
<Directory /your/path>
    AllowOverride All
</Directory>
```

4. redémarrer apache.

Méthode détaillée à <https://help.ubuntu.com/community/EnablingUseOfApacheHtaccessFiles>

#### 4.1.2. Deuxième technique : URL rewriting via PHP

Exploitation d'un « hack » du fichier de gestion des erreurs 404.

Voir <http://www.urlrewriting.fr/tutoriel-urlrewriting-sans-moteur-rewrite.htm>

## 5. Annexes

### 5.1. Bibliographie

[Alsacreation, 2014] Alsacreation (2014). Alsacreation, communauté d'apprentissage des standards du web. <http://www.alsacreation.com/>. consulté en septembre 2014.

[documentor, 2014] documentor, P. (2014). Php documentor, auto-documentation tool for php. <http://www.phpdoc.org/>. consulté en septembre 2014.


[Motte, 2014] Motte, J. (2014). Vade-mecum d'acupuncture traditionnelle. Guy Trédaniel éditeur, 2ème édition.

[New Digital Group, 1999] New Digital Group, I. (1999). Smarty, template engine. <http://www.smarty.net/>. consulté en septembre 2014.

[Openclassrooms, 1999] Openclassrooms (1999). OpenClassrooms, cours et tutoriels en ligne. <http://fr.openclassrooms.com/>. consulté en septembre 2014.

### 5.2. Acronymes

- API Application Programming Interface. 43
- CSS Cascading Style Sheets. i, 4, 6, 7, 33, 34
- DTD Document Type Definition. 25
- HTML HyperText Markup Language. i, iv, v, 3, 4, 6, 7, 30, 33, 34
- LAMP Linux Apache MySQL PHP. 14, 30, 35
- MVC Modèle Vue Contrôleur. 35
- PHP PHP: Hypertext Preprocessor, anciennement Personal Home Page. i, 14–16, 23, 24, 35
- QCM Questionnaire à Choix Multiples. iii
- REST REpresentational State Transfer. i, 39
- RSS Really Simple Syndication. 31
- SOAP Simple Object Access Protocol. 39
- SQL Structured Query Language. i, 14, 16, 18, 23, 25, 26
- TLI Technologies et Langues de l'Internet. i, iii
- URL Uniform Resource Locator. 11, 12
- W3C World Wide Web Consortium. 3, 6
- WAI Web Accessibility Initiative. 3, 6
- WCAG Web Content Accessibility Guidelines. 3, 4, 6
- XML eXtensible Markup Language. i, 24–26, 38, 43

- 
- XSD XML Shema Definition




## 6. modèle

### 6.1. Modèle

Modèle :

<b>Modèle de code</b>
-----------------------

 Modèle de question