
DÉVELOPPEMENT WEB



SOMMAIRE :

Remerciements	page 3
Présentation de l'entreprise	pages 4-5
Spécification des exigences	pages 6-9
Analyse	page 10
Participation à un projet/étude	page 11
Conclusion	page 22

REMERCIEMENTS

Je tiens à remercier dans un premier temps, toute l'équipe pédagogique du lycée Turgot et les intervenants professionnels responsables de la formation de BTS SIO 2^{ème} année, pour avoir assuré la partie théorique de celle-ci.

Je remercie également monsieur Leguern pour l'aide et les conseils concernant les missions évoquées dans ce rapport, qu'il m'a apporté lors des différents entretiens dans le cadre du lycée.

Je tiens à remercier tout particulièrement et à témoigner toute ma reconnaissance aux personnes suivantes, pour l'expérience enrichissante et pleine d'intérêt qu'elles m'ont fait vivre durant ces deux mois au sein de l'agence SENZO : Franck BADUEL et Thomas POZZO DI BORGO.

PRÉSENTATION DE L'ENTREPRISE



J'ai effectué mon stage dans l'Agence SENZO qui est une SARL (Société A Responsabilité Limitée), située dans le Val de Marne. Elle a été créée par deux associés, monsieur Baduel et monsieur Pozzo Di Borgo en 2012. Ensemble, ils ont mené plus de 100 projets et travaillent en collaboration avec plus de 40 clients. La société est un leader dans le développement d'applications web avancées. Elle accompagne ses clients dans leur stratégie digitale.

C'est une TPE (Très Petite Entreprise), elle compte 4 salariés dont tous ont une spécification différentes et un cursus différent. Le directeur artistique, à l'issue de ses études en école de communication, a commencé par travailler en agence web. Il s'occupe du côté commercial et financier de l'entreprise. Le directeur technique a obtenu un diplôme d'ingénieur grande école (SUPELEC), spécialisé dans l'informatique et les réseaux, doublé d'un Master de Gestion des Entreprises à la Sorbonne. Il a commencé par être consultant expert nouvelle technologie pendant plusieurs années pour le groupe Altran ainsi que chez de prestigieux clients (SAGEM Défense Sécurité, J.C. Decaux, IBM). Quant aux deux restants, l'un a suivi un cursus dans la communication pour se réorienter ensuite dans le graphisme, il s'occupe donc de l'intégration des sites web, autrement dit, les graphismes. Quant au second, il est issu d'un DUT informatique et se charge essentiellement du développement web.

L'agence SENZO dispose de plusieurs ressources :

- Matérielles : tout le matériel informatique nécessaire pour travailler, le local où le travail s'effectue.
- Immatérielles : le savoir et le savoir-faire qu'ils maîtrisent.
- Financières : un chiffre d'affaires s'élevant à 20 000 euros.
- Humaines : la main d'œuvre, ici les 4 salariés.

Elle fait partie du domaine tertiaire et propose des sites web ou applications mobiles sur toutes sortes de thèmes. Pour le développement de ses divers sites et applications, ils utilisent le fameux Framework Zend

mais aussi différentes solutions stratégiques comme le CMS Pimcore et pour finir, Magento pour tout ce qui concerne la réalisation de boutiques e-commerce.

Elle a développé des partenariats avec divers clients tels que Orpi (immobilier), Veolia (eau) ou encore Longboard (mode, textile).

Les sites web qu'ils créent en fonction de la demande du client sont hébergés par la société OVH sur ses propres serveurs. La société OVH est le 3^{ème} hébergeur Internet dans le monde. En quelques chiffres : elle est présente dans 17 pays, héberge 18 millions d'applications web, 180 000 serveurs, 17 datacenters et 700 000 clients. Elle s'appuie également sur les normes ISO 27002 (sécurité de l'information) et ISO 27005 (gestion des risques de la sécurité des Systèmes d'Information).

I. SPÉCIFICATION DES EXIGENCES

Tout d'abord nous allons nous intéresser aux exigences fonctionnelles, qui sont les services rendus par le site web concernant l'activité d'un courtier.

Celles que j'ai rencontrées sont les suivantes :

- Faire connaître son activité auprès de nouveaux clients
- Informer ces derniers des taux de prêts

Pour cela, j'ai dû développer le site web avec une certaine structure, des templates m'ont été fourni afin de rendre le produit plus attractif pour les éventuels utilisateurs.

Nous allons désormais voir les exigences non-fonctionnelles auxquelles j'ai été confrontée. Premièrement, les langages avec lequel j'ai développé ce site : HTML5, CSS3 ainsi que le PHP.

Afin de réaliser des produits qui correspondent aux exigences des clients, les développeurs de l'agence utilisent divers logiciels, dont tous sont distribués sous une licence libre, sans pour autant que ce soit la même. La licence libre s'applique sur une œuvre de l'esprit et confère les droits d'auteur à la personne qui en est l'auteur mais qui laisse tout de même des droits aux utilisateurs : utiliser l'œuvre, l'adapter à ses besoins, la modifier, la redistribuer dans le même format dans lequel elle a été trouvée.

Git :



C'est un logiciel de version décentralisé, il permet de travailler en équipe chacun de son côté sur des choses différentes et lorsque les diverses versions seront mises en commun, chaque membre du groupe pourra voir les modifications qui ont été faites par rapport à la dernière version enregistrée.

Il est distribué sous une licence libre appelée : GNU GPL v2.

Redmine :



C'est un logiciel de gestion de projets. Il est considéré comme étant l'un des outils de gestion de projets collaboratifs Open Source parmi les plus aboutis. Il va donc permettre aux développeurs de planifier leur projet grâce à un diagramme de GANTT. Il possède également l'avantage de supporter plusieurs bases de données comme MySQL, PostgreSQL, SQL Server ou bien SQLite.

Il existe aussi une fonctionnalité qui permet de saisir le temps passé sur le projet ou une partie, afin de pouvoir avoir une idée de la facturation que l'entreprise va faire vis-à-vis du client.

Il est distribué sous une licence libre appelée : GNU GLP.

/!\ Facturation client – retard /avancement /!\

Zend (Version 2):



Zend est un Framework qui permet d'accélérer le développement de sites web, faciliter la maintenance et coder de manière « unique » en PHP, c'est-à-dire sans réutiliser plusieurs fois le même code.

Il est principalement connu pour son architecture MVC (Modèle – Vue – Contrôleur) qui permet de séparer le code en trois parties afin de repérer plus facilement, ce qui permet un gain de temps considérable. Mais grâce à ce Framework, une authentification est possible ainsi que les ACL (droits d'accès).

Il est distribué sous une licence libre appelée : BSD, elle permet de réutiliser tout ou une partie du logiciel sans restriction, qu'il soit intégré dans un logiciel libre ou propriétaire.

Pimcore :



Le logiciel Pimcore est un système de gestion de contenu, autrement appelé un CMS. Il permet la création ainsi que la gestion de sites et applications web. Il est basé sur deux frameworks notamment Zend (du côté serveur) et ExtJS (du côté du client).

Il va permettre de personnaliser le Back-Office (partie non visible par l'utilisateur) mais également de permettre une automatisation de certaines tâches telles que la conversion des documents offices ou bien la prévisualisation des médias.

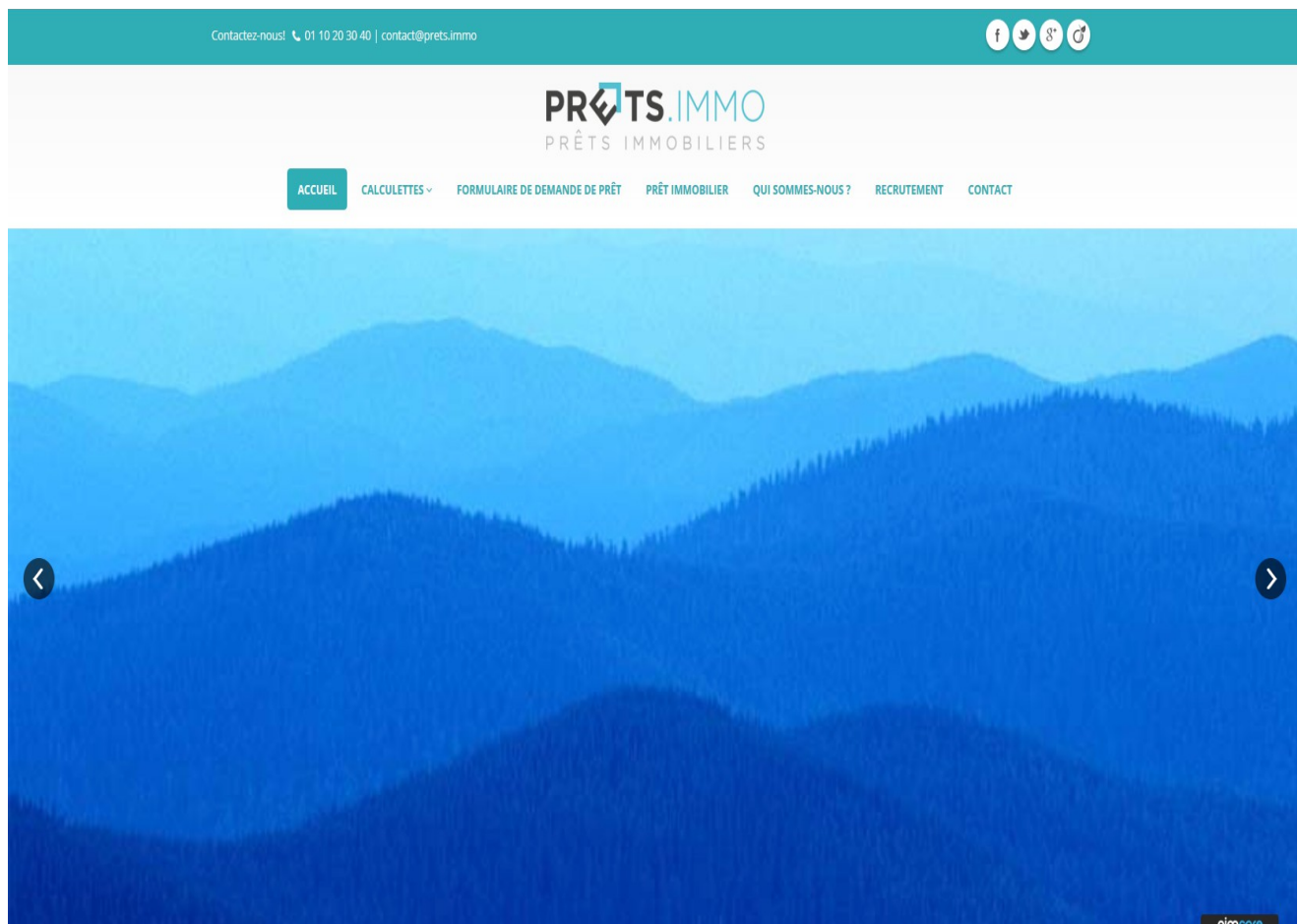
Il est distribué sous une licence libre appelée : BSD.

Concernant les codes développés par l'agence SENZO, elle les possède, seulement si le client s'y connaît en informatique, il pourra modifier ce dernier s'il désire ajouter quelque chose, en revanche, s'il ne s'y connaît pas, il devra faire appel une nouvelle fois à l'agence afin qu'elle rajoute ce qu'il désire. Le code est donc la propriété de l'agence digitale SENZO.

II. ANALYSE

Tout d'abord, j'ai dû utiliser des templates¹ afin de faire le squelette du site avant même de remplir les différentes pages. Pour cela, j'ai utilisé l'outil Pimcore qui m'a servi à introduire divers champs, comme un wysiwyg (de l'anglais « What you see is what you get », autrement dit « Ce que l'on voit et ce qu'on a ») ou bien une image ou encore un input.

Une fois le squelette entièrement créé, il a fallu remplir les pages. J'ai donc commencé par les pages « simples », c'est-à-dire celles qui ne nécessitaient pas de fonctions particulières. Ces pages sont les suivantes : Prêt immobilier, qui sommes-nous, recrutement ainsi que mentions légales. Je n'ai pas fait toutes les pages, le plan du site, l'accueil ainsi que la page de contact a été faite par mon collègue. Voici donc l'aperçu de ces dernières.



Ici, voici le footer (« bas de page ») qui est divisé en trois parties pour une question de référencement, c'est-à-dire la visibilité du site web sur les moteurs de recherches.

¹ Structure du site internet (exemple : colonne à gauche, à droite, pleine page..).

The screenshot displays the Prêts.immo website. The header features the logo 'PRÊTS IMMO' and a navigation menu with links: ACCUEIL, CALCULETTES, FORMULAIRE DE DEMANDE DE PRÊT (highlighted), PRÊT IMMOBILIER, QUI SOMMES-NOUS?, RECRUTEMENT, and CONTACT. The main content area includes three paragraphs of text about the company's services and a phone number '01.10.20.30.40'. The footer is divided into three columns: 'MENU' with links to Accueil, Calculettes, Formulaire de demande de prêt, Prêt immobilier, Qui sommes-nous?, Recrutement, and Contact; 'LIENS' with links to Courtier en prêt immobilier à Sucy-en-Brie (94370) and Courtier en prêt immobilier à Ormesson (94999); and 'LIENS' with links to Le prêt immobilier, Le prêt gratuit, Le crédit immobilier, and Le crédit gratuit. On the right side of the footer, there is a 'CONTACTEZ-NOUS' section with the phone number '01 10 20 30 40', fax number '01 11 21 31 41', address '51 rue du Général de Gaulle, 94430 Chennevières-sur-Marne, France', and email 'contact@prets.immo'. Social media icons for Facebook, Twitter, and Google+ are also present. The footer also includes a copyright notice '© Copyright 2013. Tous droits réservés.' and a plan du site link.

Ensuite, j'ai donc fait les pages énoncées précédemment, nous allons commencer par expliquer la page prêt immobilier. Sur cette page utilise un template avec une colonne sur la droite, aussi appelé snippet² dans laquelle figure la durée des prêts ainsi que leur taux. Des pictogrammes ont été insérés afin de montrer au futur utilisateur si le taux est égal, en hausse ou en baisse.

² Portion de page ou de code réutilisable.

[ACCUEIL](#)
[CALCULETTES](#)
[FORMULAIRE DE DEMANDE DE PRÊT](#)
[PRÊT IMMOBILIER](#)
[QUI SOMMES-NOUS ?](#)
[RECRUTEMENT](#)
[CONTACT](#)

Accueil / Prêt immobilier

Prêt immobilier

Votre prêt immobilier au meilleur taux avec nos courtiers en crédit.

Faire un crédit immobilier permet de **financer une acquisition immobilière d'un terrain** en vue d'une future construction, ou la **réalisation de travaux** destinés à réparer ou à améliorer un bien.

On compte différents types de crédit immobilier : **le prêt amortissable classique** (avec échéances), **les prêts aidés** (PTZ+, prêt conventionné, PEL ...) et **les prêts complémentaires** (prêt Action Logement, prêt aux fonctionnaires...).

Prêt immobilier : comment bénéficier des meilleures offres ?

Vous êtes à la recherche d'un crédit immobilier pour financer un bien immobilier ? Il existe une solution pour bien choisir son prêt immo : faire appel à Prêts.immo !

Nous négocions auprès de nos partenaires bancaires les meilleures offres de financement adaptées à votre budget, nous mettrons tout en œuvre pour obtenir le meilleur taux possible pour votre prêt immobilier.

Nos meilleurs taux actuels

Durée	Prêt taux fixe	Prêt taux variable
5 ans	10 %	11 %
10 ans	2 %	12 %
15 ans	3 %	13 %
20 ans	4 %	14 %
25 ans	5 %	15 %
30 ans	6 %	16 %

Meilleurs taux immobiliers actuels négociés par Prêts.immo au 19 janvier 2015, hors assurances. Les taux négociés avec les banques dépendent aussi de votre profil emprunteur (apport, durée prêt, travail...)

Demande en ligne

Votre demande de prêt en 2 minutes chrono !

Concernant les pages restantes : qui sommes-nous ainsi que recrutements ce sont de simples pages contenant du texte.

En revanche, après j'ai dû faire le formulaire de demande de prêt, principalement en HTML5 et CSS3.

[ACCUEIL](#)
[CALCULETTES](#)
[FORMULAIRE DE DEMANDE DE PRÊT](#)
[PRÊT IMMOBILIER](#)
[QUI SOMMES-NOUS ?](#)
[RECRUTEMENT](#)
[CONTACT](#)

Demande de prêt immobilier

Vous avez un projet immobilier et pas de temps à perdre ? Déposez votre demande de prêt immobilier en ligne en **2 minutes**

Formulaire de demande de prêt

VOTRE PROJET D'ACHAT CONCERNE

Indiquez votre type de projet :

NATURE DU BIEN

Indiquez la nature du bien à acheter :

INFORMATIONS SUR LE BIEN FAISANT L'OBJET DE VOTRE DEMANDE

Lieu du projet :

Prix d'achat (en Euros) :

Montant de votre apport (en Euros) :

III. PARTICIPATION A UN PROJET

INSTALLATION :

Dans ce document nous allons voir comment coder pas à pas une liste de voitures stockées dans une base de données grâce à Pimcore et Zend.

Pour installer Pimcore, il m'a juste fallu me rendre sur leur site (accessible à cette [adresse](#)) et télécharger un dossier Zip, qu'il a simplement fallu décompresser. A partir de là, j'ai pu accéder à la page d'accueil par défaut, puis je me suis connectée dans Pimcore en tant qu'admin afin de commencer le projet.

Pour ce dernier, j'ai utilisé la version 2 de Zend et la version 2.3.0 pour Pimcore.

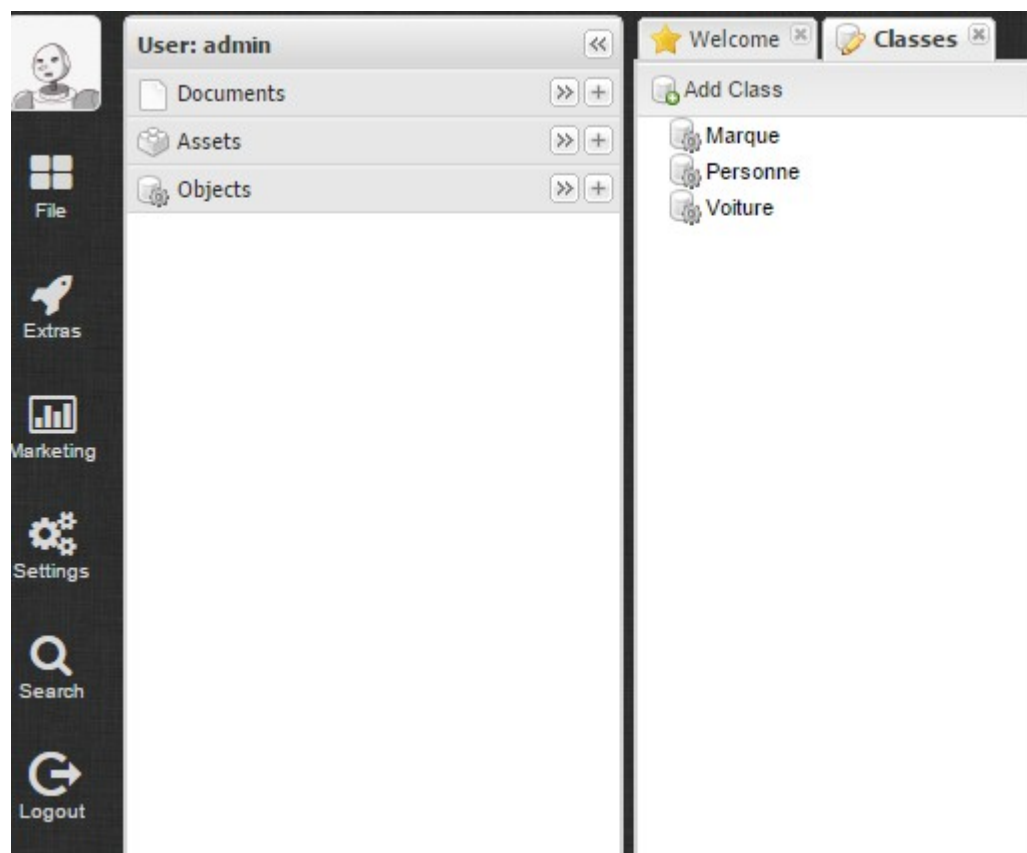
DÉVELOPPEMENT :

➤ Étape 1 :

Les rubriques Documents, Assets et Objects sont les trois grandes parties du CMS Pimcore.

Documents regroupe les fichiers créés, Assets est un regroupement des divers médias que l'on a pu ajouter et enfin, Objects définit tous les objets (poo) qui ont été créés.

A droite de l'écran se situent les classes qui ont été créées (ici Marque, Passager et Voiture). Pour en ajouter une, il suffit de cliquer sur Add Class et de lui donner un nom.



➤ Étape 2 :

Une fois la classe créée, cet écran apparaît : Pimcore enchaîne sur la création des éléments du formulaire (ici la marque). Nous avons le choix de créer un champ texte (text), une liste déroulante (select), une date (date) etc. Lorsque l'on a choisi ce que l'on désire ajouter, il suffit juste de préciser le nom du champ (name) et la classe parent en cas d'héritage.

Marque (ID: 2)

Layout

Configure Custom Layouts

Base

Layout

Add Layout Component

Add Data Component

Delete

Text

Number

Date

Select

Relation

Structured

Geographic

Other

CRM

Basic configuration

Name: Marque

Description:

Allow inheritance: ☐

Allow variants: ☐

Show variants in tree: ☐

Parent class:

Icon:

Preview URL:

Visibility of system properties

ID (Grid View): ☒

ID (Search): ☒

Path (Grid View): ☒

Path (Search): ☒

Published (Grid View): ☒

Published (Search): ☒

Modification Date (Grid View): ☒

Modification Date (Search): ☒

Creation Date (Grid View): ☒

Creation Date (Search): ☒

➤ Étape 3 :

Ici la classe Marque contient deux champs textes, le nom de la marque et son pays. La case « Mandatory field » indique qu'en la cochant le champ devient alors obligatoire. Il est également possible d'ajouter du CSS.

Configure Custom Layouts

Base
Layout
nom
pays

General Settings (Input)

Name: nom
Title (Label): Nom
Tooltip:

Mandatory field: ☒
Not editable: ☐
Invisible: ☐
Visible in Grid View: ☐
Visible in Search Result: ☐
Indexed: ☐

Layout Settings (Pimcore Admin)

CSS Style (float: left; margin: 10px; ...):

Specific Settings (Input)

Width:
Columnlength: 255

Regular Expression Validation
RegEx:
⚠ RegExp without delimiters, has to work in both JS and PHP (Delimiter: #)
Test string:

Reload Definition Import Export Save

➤ Étape 4 :

Ici nous avons la classe voiture, la classe principale qui contient toutes les informations concernant les voitures (le nom, la marque, la date de mise en circulation, la description, les options, l'image et enfin les passagers qui sont dedans).

The screenshot shows the NetBeans IDE interface for configuring the 'Voiture' class. The 'Layout' tab is active, displaying a tree view of the class structure. The 'Basic configuration' tab is also visible, showing fields for Name, Description, and various system properties.

Layout

- Base
 - Layout
 - nom
 - marque
 - miseEnCirculation
 - description
 - options
 - image
 - personnes

Basic configuration

Name: Voiture

Description:

Allow inheritance: ☐

Allow variants: ☐

Show variants in tree: ☐

Parent class:

Icon:

Preview URL:

Visibility of system properties

ID (Grid View):	<input checked="" type="checkbox"/>
ID (Search):	<input checked="" type="checkbox"/>
Path (Grid View):	<input checked="" type="checkbox"/>
Path (Search):	<input checked="" type="checkbox"/>
Published (Grid View):	<input checked="" type="checkbox"/>
Published (Search):	<input checked="" type="checkbox"/>
Modification Date (Grid View):	<input checked="" type="checkbox"/>
Modification Date (Search):	<input checked="" type="checkbox"/>
Creation Date (Grid View):	<input checked="" type="checkbox"/>
Creation Date (Search):	<input checked="" type="checkbox"/>

➤ Étape 5 :

Voici les différents objets voiture qui ont été créés grâce aux propriétés sur l'écran précédent (étape 4). Nous choisissons donc le nom de l'objet que l'on vient de créer, apparaît ensuite cette page avec les champs déjà définis précédemment.

Il suffit de remplir le nom de la voiture, sa marque (en ajoutant au fur et à mesure grâce au bouton vert situé à droite), la date de mise en circulation, la description, les options disponibles, concernant l'image, il suffit de la télécharger et de l'inclure directement dans l'objet ci-dessous. Toutes les images seront répertoriées dans la partie Assets. Et pour les passagers, il suffit de les créer aussi un par un grâce à la classe Personne.

User: admin

Documents

Assets

Objects

Home

- marque
- personne
- voiture
 - v3
 - v4
 - v5
 - v6

Welcome | Classes | voiture | v3 | v4 | v5 | v6

Save & Publish | Unpublish | Delete | Reload | Show in Tree | Info | ID 10 | Voiture

Edit | Properties | Versions | Schedule | Dependencies | Notes & Events | Children Grid

Nom

Nom:

Marque

ID	Path	Type			
13	/marque/renault	Marque	↑	↓	✕


miseEnCirculation:

description

Moteur essence, ESP, ABS.

Options:

Image



Personnes

ID	Path	Type			
17	/personne/jacques	Personne	↑	↓	✕
16	/personne/jean	Personne	↑	↓	✕
18	/personne/jule	Personne	↑	↓	✕

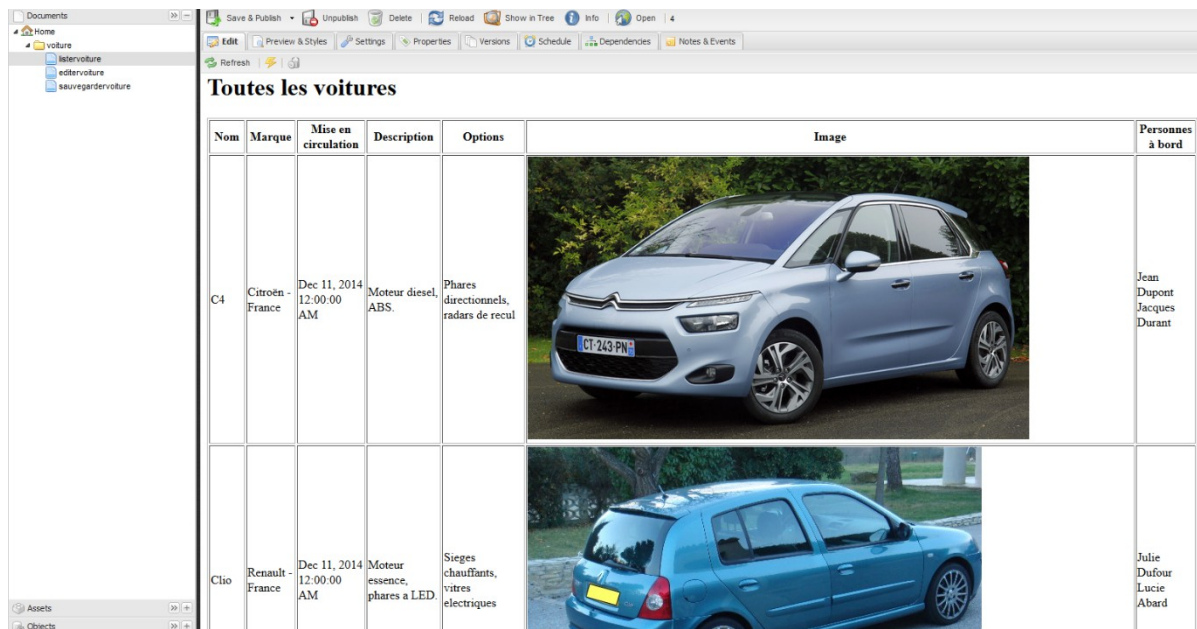
➤ Étape 6 :



Nous arrivons désormais aux vues qui va falloir créer. Dans la partie « Documents » on va créer un dossier Voiture qui représente la classe, et les divers documents « listervoiture », « editervoiture » et « sauvegardervoiture » qui représentent les vues correspondantes : Pimcore générant du code Zend, on retrouve ces fichiers dans le dossier « views » du projet Zend.

Par exemple, dans Pimcore nous rajoutons seulement un champ input, image, select ou autre, et dans NetBeans ce sera du code PHP pour rajouter une image :

```
< ?php $this->image('content') ; ?>
```

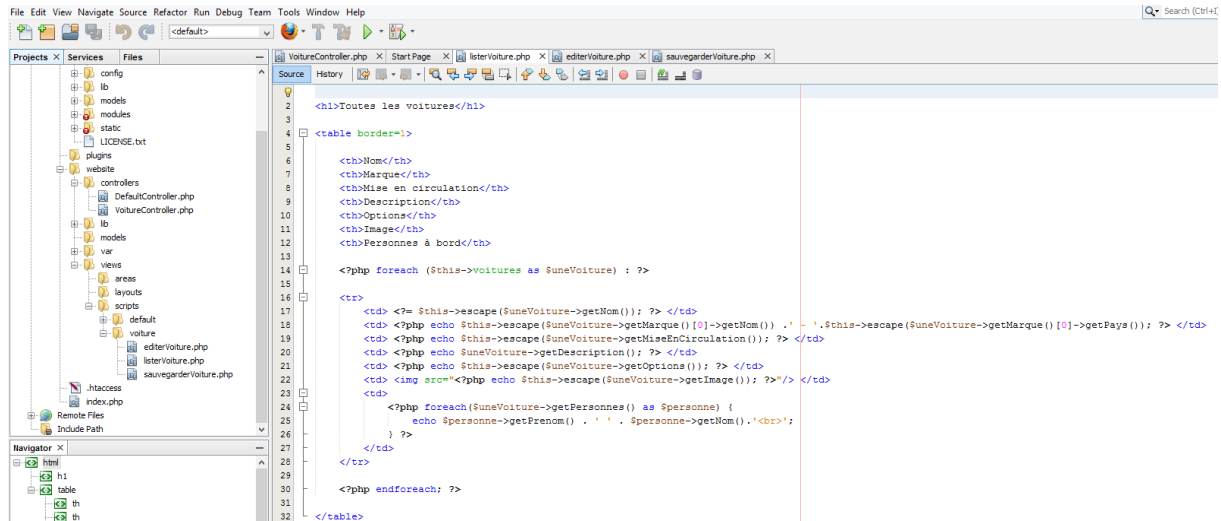
Le nom que l'on met entre parenthèse est celui qui sera utilisé dans la base de données. En revanche, sur une page, il ne peut pas y avoir deux fois le même nom, même s'il y a plusieurs input (dans notre exemple). Il faut donc le renommer explicitement.



Nom	Marque	Mise en circulation	Description	Options	Image	Personnes à bord
C4	Citroën - France	Dec 11, 2014 12:00:00 AM	Moteur diesel, ABS.	Phares directionnels, radars de recul		Jean Dupont Jacques Durant
Clio	Renault - France	Dec 11, 2014 12:00:00 AM	Moteur essence, phares à LED.	Sieges chauffants, vitres électriques		Julie Dufour Lucie Abard

➤ Étape 7 :

Voici le code dans Netbeans qui permet d'afficher toutes les voitures créées dans Pimcore. Lorsque l'on crée des classes dans Pimcore, automatiquement un fichier est créé dans Netbeans.



The screenshot shows the NetBeans IDE interface. On the left, the 'Projects' pane displays a file tree for a project named 'voiture'. The tree includes folders like 'config', 'lib', 'models', 'static', 'plugins', 'website', and 'var', along with files like 'DefaultController.php', 'VoitureController.php', 'LICENSE.txt', 'index.php', and 'htaccess'. The 'Navigator' pane at the bottom left shows the 'html' folder containing 'h1', 'table', 'th', and 'th' files. The main editor window displays the code for 'VoitureController.php'. The code is a PHP script that uses the Pimcore API to fetch a list of cars and displays them in an HTML table. The table has columns for 'Nom', 'Marque', 'Mise en circulation', 'Description', 'Options', and 'Image'. For each car, it also lists the associated 'Personnes à bord' (people on board).

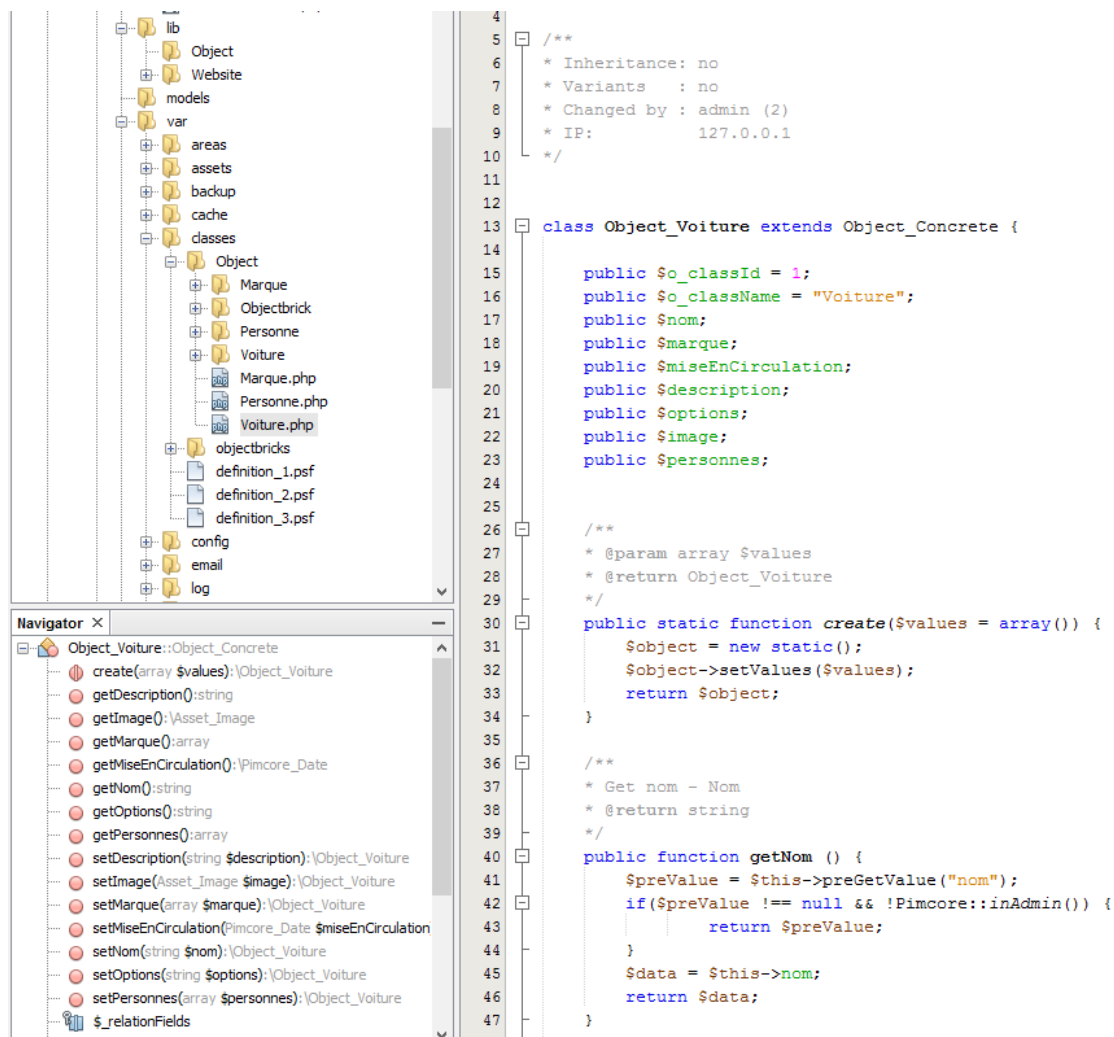
```
1 <h1>Toutes les voitures</h1>
2
3
4 <table border="1">
5
6 <tr>
7 <th>Nom</th>
8 <th>Marque</th>
9 <th>Mise en circulation</th>
10 <th>Description</th>
11 <th>Options</th>
12 <th>Image</th>
13 <th>Personnes à bord</th>
14
15 </tr>
16
17 <tr>
18 <td> <?php echo $this->escape($SuneVoiture->getNom()); > </td>
19 <td> <?php echo $this->escape($SuneVoiture->getMarque()[0]->getNom()); . ' - ' . $this->escape($SuneVoiture->getMarque()[0]->getPays()); > </td>
20 <td> <?php echo $this->escape($SuneVoiture->getMiseEnCirculation()); > </td>
21 <td> <?php echo $SuneVoiture->getDescription(); > </td>
22 <td> <?php echo $this->escape($SuneVoiture->getOptions()); > </td>
23 <td>  </td>
24 <td>
25 <tr>
26 <td> <?php foreach($SuneVoiture->getPersonnes() as $personne) {
27 <td> echo $personne->getPrenom() . ' ' . $personne->getNom(). "<br>";
28 <td> } >
29 </td>
30 </tr>
31
32 </table>
```

➤ Étape 8 :

Dans le premier encadré à gauche, on peut donc remarquer que lorsque l'on a créé nos objets dans Pimcore, les fichiers des trois classes ont été automatiquement générées, il a donc créé tous les setters et les getters ce qui nous permet de les réutiliser sans avoir à les coder, c'est donc un gain de temps. Ce sont donc ces propriétés que l'on réutilise dans la vue précédente afin de récupérer les divers champs.

Concernant la documentation des fonctions codées préalablement par Pimcore, elle est fournie en même temps ce qui nous permet de savoir quels paramètres elle reçoit ainsi que ce qu'elle doit retourner. Lorsque l'on crée un champ dans Pimcore, par exemple le nom de la voiture, automatiquement Pimcore crée la fonction `getNom()` et ainsi de suite pour tous les champs qui seront créés par la suite.

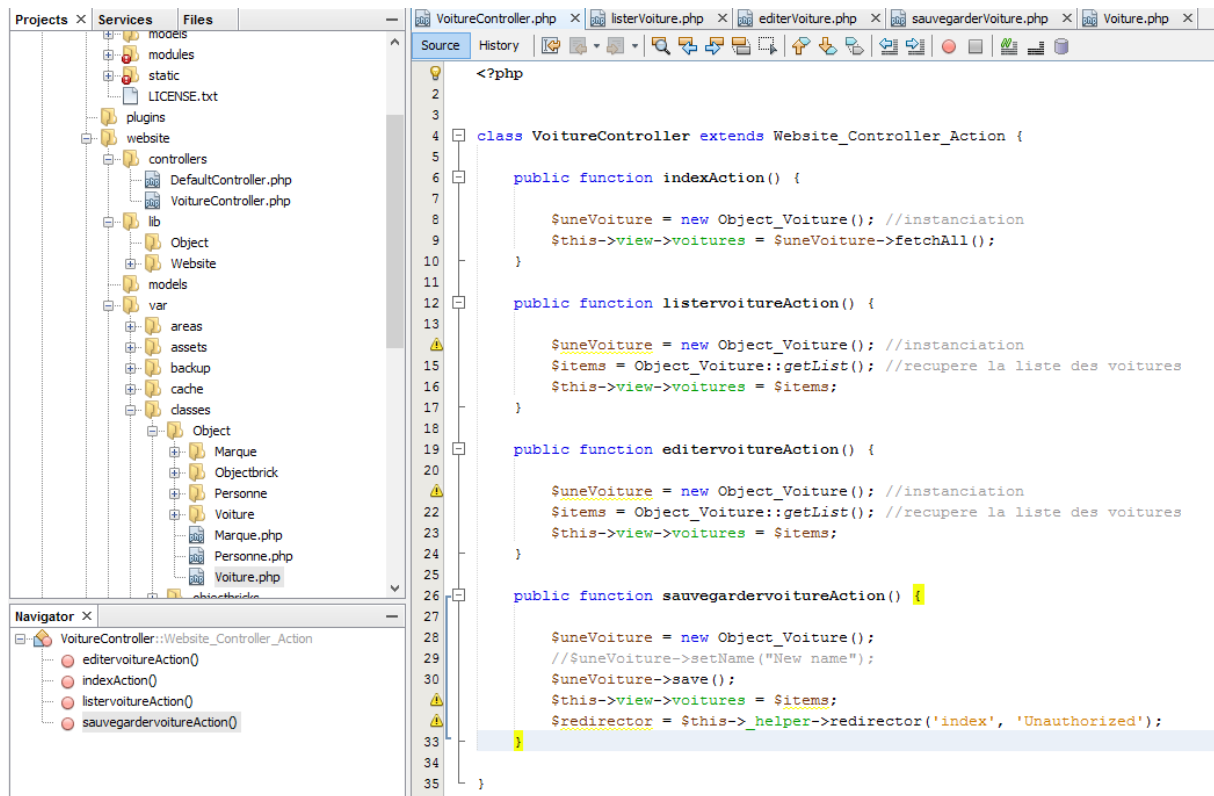
Sur l'image, on peut observer la partie en bas à gauche qui comprend toutes les méthodes qui ont été créées, et donc qui correspond aux champs créés dans Pimcore.



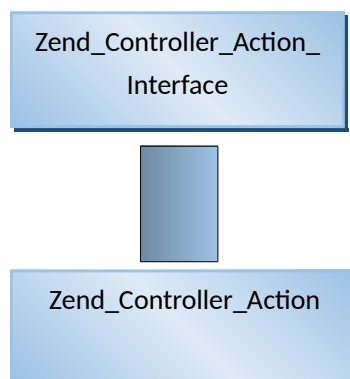
➤ Étape 9 :

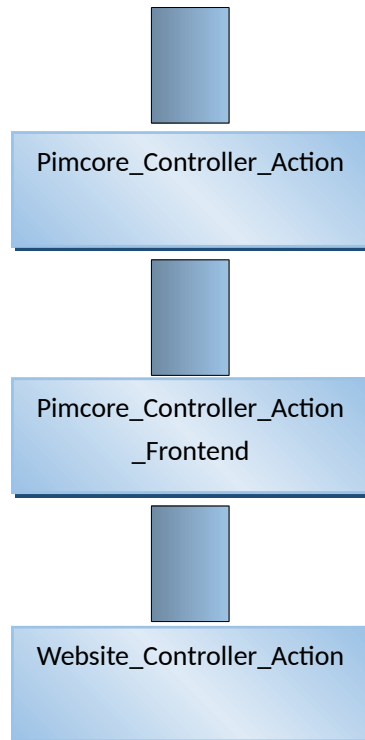
Voici le contrôleur qui gère la classe Voiture, il contient les différentes actions possibles, ici lister les voitures, les éditer et les sauvegarder. On peut aussi remarquer que la classe VoitureController hérite de la classe Website_Controller_Action.

Les deux premières sont identiques, seul les vues changent. La première affichera seulement la liste de voitures alors que la seconde, affichera non seulement la liste des voitures, mais on aura la possibilité de modifier les champs, comme son nom, sa marque etc.



Pour mieux comprendre l'héritage entre les différentes classes, voici un diagramme UML.

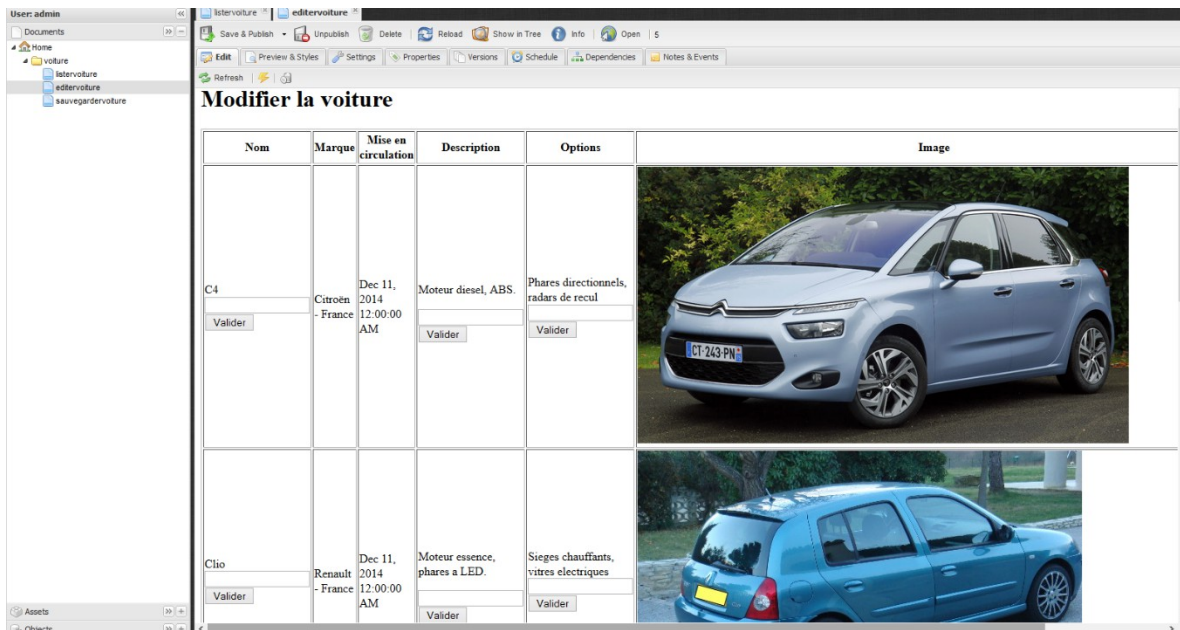




Toutes les classes qui héritent de Website_Controller_Action, héritent donc de toutes celles qui se situent au-dessus d'elles, cela évite donc le code inutile et permet donc un code PHP « unique ».

➤ Étape 10 :

Au lieu de tester dans un navigateur, Pimcore nous permet de visualiser directement ce que rend notre vue que l'on vient de coder dans l'EDI.

Et
si

jamais il y a une erreur dans le code que nous venons de réaliser, Pimcore nous indique lesquelles et l'endroit où elle se situe.

Assets

Objects

Intervolture

editervolture

savegardervolture

Exception

monpimcore.local/www/pimcore/models/Object/Abstract.php:491

Object_Abstract save

monpimcore.local/www/pimcore/models/Object/Abstract.php:491

VoitureController sauvegardervoltureAction

monpimcore.local/www/pimcore/models/Object/Abstract.php:491

Zend_Controller_Action dispatch

monpimcore.local/www/pimcore/models/Object/Abstract.php:491

Zend_Controller_Dispatcher_Standard dispatch

monpimcore.local/www/pimcore/models/Object/Abstract.php:491

Zend_Controller_Front dispatch

monpimcore.local/www/pimcore/models/Object/Abstract.php:491

Pimcore run

monpimcore.local/www/pimcore/models/Object/Abstract.php:491

Exception

invalid key for object with id [0] key is: []

C:\ServeurLocal\domains\dev-monpimcore.local/www/pimcore/models/Object/Abstract.php

484.

try {

485.

// be sure that unpublished objects in relations are saved also in frontend mode, eg. in

importers, ...

486.

\$hideUnpublishedBackup = self::getHideUnpublished();

487.

self::setHideUnpublished(\$hideUnpublishedBackup);

488.

if (!Pimcore_Tool::isValidKey(\$this->getKey()) || \$this->getId() != 1){

489.

throw new Exception("invalid key for object with id [".\$this->getId()."] key is: [" .

490.

\$this->getKey(). "]");

491.

}

492.

if (in_array(\$this->getType(), self::\$types)) {

Server/Request Data

ALLUSERSPROFILE

C:\ProgramData

APPDATA

C:\WINDOWS\system32\config\systemprofile\AppData\Roaming

CommonProgramFiles

C:\Program Files (x86)\Common Files

CommonProgramFiles(x86)

C:\Program Files (x86)\Common Files

CommonProgramW6432

C:\Program Files\Common Files

COMPUTERNAME

PC-LOTUS

ComSpec

C:\WINDOWS\system32\cmd.exe

FP_NO_HOST_CHECK

NO

HOME

C:\Users\hibaoui

LOCALAPPDATA

C:\WINDOWS\system32\config\systemprofile\AppData\Local

NUMBER_OF_PROCESSORS

4

OS

Windows_NT

Path

C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\WINDOWS\System32\WindowsPowerShell\v1.0\;C:\Program Files (x86)\Zend\ZendServer\share\ZendFramework\bin;c:\Program Files (x86)\Zend\ZendServer\bin;c:\Program Files (x86)\Zend\ZendServer\bin;c:\Program Files (x86)\Zend\ZendServer\bin;c:\Program Files (x86)\Git\cmd;c:\Program Files\Nodejs\

PATHEXT

.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.NSF;.ASHX;.JSC

CONCLUSION

Ce stage m'a apporté de nombreux points positifs. Tout d'abord, la façon dont j'ai appréhendé le métier de développeur, bien différent de la conception que je m'en faisais, il y a énormément à apprendre.

Ensuite, le fonctionnement de l'entreprise ainsi que la relation avec les clients, le côté communication de cette dernière est bien différente d'autres entreprises plus grandes. Sachant que c'est une TPE, elle fonctionne tout à fait différemment que les grandes.

Durant cette période, j'ai appris diverses choses, tout d'abord quelques commandes Linux grâce à l'outil Git le gestionnaire de versions, mais aussi au niveau du développement, d'une part avec Zend en utilisant l'IDE Netbeans, puis le CMS Pimcore. J'ai d'ailleurs trouvé que le lien entre ce dernier et Zend est très intéressant et surtout, très efficace au niveau du temps, il y a un gain de temps énorme.

De mon point de vue, j'ai pris conscience du véritable métier de développeur ainsi que les outils divers qui peuvent être utilisés dans une entreprise. La communication effectuée avec les clients m'a beaucoup intéressée car c'est tout autre chose, et il y a toujours des besoins nouveaux de la part de ce dernier, et donc toujours de nouvelles mises à jour à effectuer pour qu'ils soient comblés.