

DOSSIER PROJET PROFESSIONNEL

Agence d'architecture Bolikian



Chloé Mayer

26/09/2020

RNCP 3 – La Plateforme

SOMMAIRE

Introduction

1.1	Résumé du dossier de projet	2
-----	-----------------------------	---

Présentation

2.1	Présentation personnelle	2
2.2	Présentation de La Plateforme	3

Projet entreprise

3.1	Présentation de l'entreprise	3
3.2	Présentation du projet	4
3.3	Compétences utilisées	5
3.4	Cahier des charges	7
3.5	Organisation	9
3.6	Spécifications techniques	10

Développement

4.1	Réalisations	11
4.1.1	Maquettage du site	11
4.1.2	Développement de la partie front-end	12
4.1.3	Développement de la partie back-end	19
4.2	Jeu d'essai	23
4.3	Difficultés rencontrées	24
4.4	Sites anglophones	25

Conclusion	26
------------	----

Annexes	26
---------	----

INTRODUCTION

1.1 Résumé du dossier de projet

Je vais vous présenter dans ce dossier une partie du travail réalisé lors de ma formation à La Plateforme, située à Marseille. Plus précisément le développement de mon projet professionnel réalisé pour une agence d'architecture. Ce dossier regroupe un ensemble de compétences pour mon titre RNCP de niveau 5. Vous y retrouverez donc un panel de compétences mises en avant lors de ce projet, ainsi que son déroulement, les attentes de la personne en charge du projet, mais aussi les problèmes rencontrés et sa réalisation finale.

PRÉSENTATION

2.1 Présentation personnelle

Je me présente Chloé Mayer, 22 ans, résidente de Marseille dans le 1er arrondissement. Après avoir passé mon baccalauréat Littéraire (spécialité Art Plastique) au lycée Périer, je me suis dirigée vers un service civique pour combler mon envie de voyage et découvrir des nouveaux champs d'activités. Après cela, j'ai découvert l'école de La Plateforme, par la mission locale, qui propose une formation en tant que développeur web / web mobile. Attirée par les métiers du numérique et surprise par la facilité d'accès, j'ai suivi cette formation par

curiosité et opportunité. J'ai maintenant pour but de me professionnaliser dans le front-end pour réaliser ce qui me plaît le plus, c'est à dire pouvoir faire ressortir l'identité de mes clients (tout en apaisant mon côté perfectionniste).

2.2 Présentation de La Plateforme

La Plateforme est une école du numérique et des nouvelles technologies co-fondée avec le Club Top 20 réunissant les grandes entreprises de la Métropole Aix Marseille. Elle comprend une offre de formations diversifiées destinées à former des codeurs et développeurs web, des experts en sécurité, des ingénieurs spécialisés en Intelligence Artificielle, et des cadres d'entreprises au travers de cycles de formations continues. Je suis actuellement la formation "Coding School" qui s'adresse à tous les passionnés sans diplômes et sans limite d'âge. C'est une formation gratuite qui offre donc ses chances à toutes et à tous.

PROJET DE L'ENTREPRISE

3.1 Présentation de l'entreprise

Ce projet professionnel est réalisée avec l'agence d'architecture Bolikian. Elle se situe à Marseille dans le 1er arrondissement. Architecte DPLG depuis 15 ans, Marjorie BOLIKIAN, base son travail autour de la qualité de vie.

Au-delà de ses compétences généralistes, elle s'est tournée de manière significative vers la construction responsable et durable depuis plusieurs années

en gérant des projets de la conception au suivi de chantier.

Du micro projet au bâtiment public elle est convaincue que la conception architecturale est nécessaire pour apporter une valeur ajoutée au site, et une qualité de vie aux usagers qui l'utilisent. Pour cela elle aborde ses projets au travers de différentes échelles de conception (paysage, architecture, design, signalétique) qui sont autant de dimensions à maîtriser pour créer une qualité d'espace.

Persuadée que son métier s'enrichit constamment des compétences connexes de l'ingénierie du bâtiment aux démarches artistiques, elle travaille en atelier avec des confrères architectes d'intérieurs, photographes, plasticiens qui lui permettent de composer à chaque projet une équipe adaptée.

3.2 Présentation du projet

Ce projet professionnel est réalisé pour l'Agence d'Architecture Bolikian en collaboration avec mon camarade Jonathan Prevedan. Il est sur une durée de 5 mois.

Afin de faciliter la mise en relation entre prestataires et clients, l'idée d'un site "vitrine" c'est tout de suite imposé. Ce projet professionnel consiste donc à créer un portfolio par lequel nous améliorons la visibilité des projets effectués et le parcours de l'architecte et par conséquent la mise en relation entre architecte et clientèle.

A l'arrivée sur le site l'utilisateur est directement confronté à une page d'introduction représentant l'identité de la cliente. Il pourra en cliquant sur le logo voir l'ensemble des projets postés par catégories avec toutes leurs

descriptions ainsi que les études et le parcours menée par madame Bolikian, mais aussi un page “contact” redirigeant chaque message vers la boîte mail de l’entrepreneur.

Pour terminer, ce site possède une interface administrative accessible au gérant du site permettant de gérer entièrement le site :

- modifier/supprimer/ajouter des projets
- Pouvoir gérer les catégories
- modifier/supprimer/ajouter ses études et parcours
- modifier son “avatar”

Ayant beaucoup travaillé sur la partie front-end, ce projet m’a beaucoup motivé du fait de travailler pour un réel client, qui de plus est un architecte, donc l’esthétique est d’autant plus important. L’identité de l’agence a été une source d’inspiration et m’a forcée à être méticuleuse.

3.3 Compétences utilisées

Compétences	Rôle
Maquettage	Exceptionnellement nous avons utilisé Wix pour la maquettage. la maquette propose une belle perception du travail à accomplir, esthétiquement parlant mais aussi optionnellement. Cette maquette nous a surtout aidé à mettre en commun nos idées en commun et ainsi avoir une vision d’ensemble.

Langages HTML et CSS	J'ai utilisé HTML / CSS comme structures des pages à développer et pour l'intégration. Le CSS m'a aidé pour le responsive des pages grâce aux médias queries ainsi que pour le design complet du site.
Langages PHP et SQL	Le PHP est utilisé pour les requêtes avec la base de données et l'affichage des informations récupérées. De nombreuses requêtes avec la base de données sont réalisées pour le formulaire de connexion, la partie admin mais également pour la messagerie personnelle...
JavaScript / jQuery / JSON	Le JavaScript est utilisé pour le responsive (navbar) mais également pour la vérification des formulaires, si un champ n'est pas rempli ou incorrect, une erreur sera affichée sans rafraichissement de la page. Pour cela, j'utilise Ajax, une communication est faite lorsque j'appuie sur un bouton de validation du formulaire avec un fichier php, qui lui va exécuter une requête sql et renvoyer le résultat sous forme de tableaux json. En fonction du résultat j'affiche un contenu différents dynamiquement en JavaScript.
Utilisation du framework Bootstrap	Bootstrap est une collection d'outils utiles à la création du design de sites et d'applications web.

	J'ai utilisé ce framework afin de faciliter le responsive du site et de m'aider dans le design de celui-ci.
Logiciel d'organisation	Pour s'organiser plusieurs logiciel peuvent être très utiles. Ici en particulier nous avons utilisé Trello, SuiteGoogle, Github, GenMyModel et Bitrix24.

3.4 Cahier des charges

Les objectifs du site Internet :

→ Proposer un site vitrine qui permettrait à l'agence d'architecture Bolikian de pouvoir améliorer sa visibilité et sa gestion en :

- Facilitant la mise en relation entre clientèle et prestataires.
- Faciliter l'accès à l'information par un design et efficace.
- Créer un panel d'administration simple et facile d'utilisation
- Faire ressortir l'identité de l'agence

L'architecture du site Internet :

→ 4 pages (Menu) :

- Projets
- Etudes
- Parcours

- Contact

→ Panel Admin

→ Sous catégorie :

- Architectures
- Intérieurs
- Autres

→ Page Intro

Visuel esthétique :

- Couleur : Blanc/Blanc cassé et jaune
- Police : CALIBRI ou CALIBRI LIGHT
- Grid avec filtre (hover) + filtre
- Adaptable/responsive
- Source d'inspiration à prendre en compte :
 - <http://krads.info/>
 - <http://tongtong.co/>
 - <https://hapticarchitects.com/>
 - <http://www.chzon.com/>

A ajouter :

- Ajout du nouveau logo
- Ajout photo professionnelle
- Se servir du serveur OVH

Rendu fini pour la rentrée.

3.5 Organisation

L'organisation du projet a été un peu tumultueuse, nous étions 3 au départ puis nous avons finis à 2 (car la deuxième personne a été basculé en deuxième promotion).

Afin de se répartir au mieux les tâches, nous avons fait en fonction des compétences de chacun, moi pour la partie front-end et mon partenaire pour la partie back-end. Nous avons quand même fait en sorte de toucher un peu à l'autre partie.

Toutes les 2/3 semaines, nous avons organisé avec l'entreprise, des réunions par téléphone ou via [Teamviewer](#) afin qu'elle puisse voir notre avancé, donner son avis en voyant notre site de par le partage d'écran. Ainsi nous avons ajouté des fonctionnalités en fonctions de ce qu'elle demandait.

Pour être encore plus efficace, des logiciels d'organisation et de planification nous ont beaucoup aidés à prévoir les tâches à accomplir dans le temps imparti; mais aussi à les effectuer au mieux.

Pour cela nous avons utilisé :

- [GenMyModel](#) (qui nous a servi à faire des maquettes pour notre base de donnée)
- [Suite Google](#) (en particulier Drive pour le partage d'image)
- [Trello](#) (nous a servi à organiser notre projet par tableaux et liste)
- [Bitrix24](#) (planificateur)

Nous avons utilisé [GitHub](#) afin de pouvoir envoyer nos fichiers a chaque modifications et pouvoir les récupérer depuis n'importe quel appareil. Mais aussi pouvoir mettre une description lors d'un commit / push d'un fichier afin de savoir ce qui a été fait pour chaque push de fichier.

3.6 Spécifications techniques et contexte

Lors du confinement, l'organisation a été totalement chamboulé. Mais grâce à tous ces outils, comme aussi Discord, nous avons pu garder le contact et avancer malgré le manque de contact et les autres soucis qu'a pu engendrer ce confinement.

Versioning et stockage

Pour ce projet nous avons utilisé Git comme logiciel de versionning, pour stocker notre code versionné sur un serveur distant afin de pouvoir travailler plus facilement dessus chacun de notre côté. Pour cela nous avons créé plusieurs branches sur GitHub, avec nos noms puis la branche master qui elle est réservée pour le projet final. Pour chaque nouvelle tâche effectuée, je faisait un push sur ma branche via l'outil TortoiseGit afin de mettre à jour mon avancé.

Technologies :

- PHP : Langage back-end
- HTML / CSS : Structure du site
- JavaScript / jQuery : Création contenu dynamique

- PHPmyAdmin : Base de données
- Bootstrap : Responsive et design du site

DÉVELOPPEMENT

4.1 Réalisations

4.1.1 Maquettage du site

Au commencement du projet, j'ai réaliser une maquette du site afin de faire un aperçu du design et du jeu de couleur de celui-ci. Exceptionnellement, je n'ai pas choisis Balsamiq ici car ma cliente avait auparavant commencé création de son site par un ami sur cette plateforme qui, finalement, n'as pas fini par aboutir son travail. Bien sur plusieurs autres Wireframes existent mais je n'ai pas encore eu l'occasion de les tester.

Je voulais, avant de passer au code, donner un aperçu à la cliente concernant le design de son site afin qu'il puisse me dire ses préférences et pouvoir avancer dans sa direction.

Evidemment, Wix n'est pas une plateforme pour maquetter, cela m'a donc pris un peu plus de temps, d'autant plus que la cliente est très pointilleuse.

(Voir annexe page 27)

Après que le client est validé la maquette, j'ai commencé cette fois-ci à réaliser la maquette côté base de données. Pour cela j'ai choisi GenMyModel qui est un outil permettant de créer plusieurs type de diagrammes / modélisations dont une pour les bases de données. J'ai donc pu construire plusieurs tables avec les données qu'elles devront avoir afin de faire marcher l'ensemble du site.

(Voir annexe pages 27)

4.1.2 Développement de la partie front-end

Je me suis occupé en grande partie du css. En prenant exemple sur la maquette qui a précédemment été réalisé, j'ai commencé par la page d'intro puis le header/footer, la page d'accueil qui affiche l'ensemble des projets postées par la cliente ainsi que les catégories pour pouvoir les répertorier. Suite à ça, j'ai fait les pages contact, parcours, études et la page commune aux projets (qui sera par la suite exclusive à chaque projets grâce aux insertions avec php/html).

➔ Intro :

(voir Annexe page 27)

Pour l'intro, Je suis allée au plus simple. J'ai créé une page en tant qu'index.php avec un lien sur le logo de l'agence qui nous redirige vers la page d'accueil.

```
<main id="intro">

    <div id="introclick">

        <a href="accueil.php">

        </a>

    </div>

</main>
```

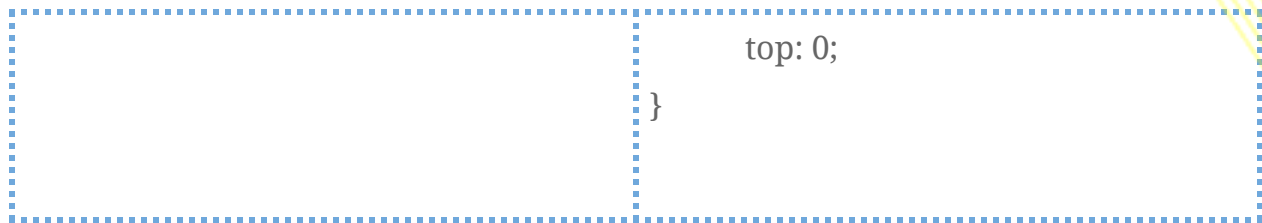
→ Header/Footer :

(voir Annexe page)

J'ai créé deux header différent; un général, en transparence, "sticky" et un autre seulement pour l'accueil, blanc et fixe. J'ai utilisé ici un z-index: 1 pour ne pas que le header passe derrière le main en "scrollant".

```
#headersticky{
    width: 100%;
    position:sticky;
    background:transparent;
    height: 23vh;
}
```

```
#headerfixed{
    width: 100%;
    position:fixed;
    background:rgb(255, 255, 255);
    height: 23vh;
    z-index: 1;
    left: 0;
```



→ Accueil :

(voir Annexe page)

La page d'accueil est séparée en deux parties. La première partie affiche par une transition CSS, en s'aidant de Keyframes et du préfixe -webkit-, les catégories de projets en entrant sur la page par un fondu.

```
.navaccueil{
    padding: 0.5% 0% 0.5% 0%;
    -webkit-animation-duration: 5s;
    animation-duration: 5s;
    -webkit-animation-fill-mode: both;
    animation-fill-mode: both;
    width: 30%;
}

@-webkit-keyframes fadeInLeft {
    0% {
        opacity: 0;
        -webkit-transform: translateX(-20px);
    }
}
```

```
100% {  
  opacity: 1;  
  -webkit-transform: translateX(0);  
}  
}  
  
.fadeInLeft {  
  -webkit-animation-name: fadeInLeft;  
  animation-name: fadeInLeft;  
}
```

La deuxième partie comprend un grid permettant d'afficher tous les projets proposés à la clientèle. Pour cela j'ai utilisé encore une fois Bootstrap, par soucis de temps. Effectivement un des atouts de ce framework, est la facilité d'utilisation du responsive. Pour ce faire plusieurs attributs sont à dispositions pour choisir la taille voulu pour telle ou telle résolution.

Sur chaque cellules, un filtre apparaît en fondu avec par dessus, en overlay, le titre du projet concerné.

IMAGE CODE

➔ Parcours/Etudes :

(voir Annexe page)

Pour ces deux pages, j'ai changer la dispositions pour chacune d'elle mais la dynamique reste la même. Pour les rendre plus attractives, j'ai opté pour utiliser un système d'ancre. Pour cela je dois rentrer dans le href="" l'id de l'endroit ou je veux que l'utilisateur arrive en cliquant sur le lien.


```
<ul>
```

```
<li class="liancre"><a href="#Formation">Formation</a></li>
```

```
<li class="liancre"><a href="#Maitrise">Maitrise d'oeuvre</a></li>
```

```
</ul>
```



```
<article class="articleetu">
```

```
<h1 id="programmation">Programmation</h1>
```

```
<h2>Sous-titre</h2>
```

```
<p>Programme : Étude de programmation + AMO
```

```
Maître d'ouvrage : Commune de Cuges les Pins / Faconeo
```

```
Lieu : Cuges les Pins</p>
```

```
</article>
```

➔ Projet :

(voir Annexe page)

Cette page se constitue de peu d'éléments (un titre, un sous-titre, une description et un carrousel). J'ai utilisé pour le carrousel le framework Bootstrap.

```
<section id="sectioncarousel">
  <!-- carroussel -->
  <div id="carouselExampleControls" class="carousel slide carousel-fade " data-ride="carousel">
    <div class="carousel-inner">
      <!-- intégrer ici vos images -->
      <div class="carousel-item active">
        
      </div>
      <div class="carousel-item">
        
      </div>
      <div class="carousel-item">
        
      </div>
    </div>
    <!-- boutons de contrôle -->
    <a class="carousel-control-prev" href="#carouselExampleControls" role="button" data-slide="prev">
      <span class="carousel-control-prev-icon" aria-hidden="true"></span>
      <span class="sr-only">Previous</span>
    </a>
    <a class="carousel-control-next" href="#carouselExampleControls" role="button" data-slide="next">
      <span class="carousel-control-next-icon" aria-hidden="true"></span>
      <span class="sr-only">Next</span>
    </a>
  </div>
  <!-- fin carroussel -->
</section>
```

➔ Contact :

La page contact contient un titre en H1, et deux articles. Dans le premier on y trouve une toute petite description et dans le deuxième un formulaire.

Pour mettre un peu plus en valeur ce formulaire et donner un peu de relief à la page, j'ai décidé de ne pas mettre de bordure mais plutôt des ombres pour souligner les champs. Et pour encore souligner cet effet de relief j'ai décidé de superposer le bouton d'envoi sur le textarea. Pour cela j'ai utilisé un "transform" et un z-index pour ne pas qu'il passe en dessous du textarea mais bien au dessus.

```

.textarea{
  z-index: 0;
  padding: 7px 10px;
  background-color: rgba(0, 0, 0, 0);
  resize: none;
  margin: 0;
  width: calc(100% + 17px);
  height: 85%;
  overflow: auto;
  border: none;
  box-shadow: -0.5px 3px 6px 1px rgba(0, 0, 0,
0.4);
  font-family: 'Questrial', sans-serif;
  color: #333333;
}

.textarea:focus{
  outline: none;
}

```

```

.envoyercon{
  background-color: #F8F281;
  cursor: pointer;
  color: white;
  font-size: 17px;
  font-weight: bold;
  padding: 14px 30px;
  text-decoration: none;
  border: 2px solid #F4EE7F;
  z-index: 1;
  transform: translateY(-30px);
  font-family: 'Questrial', sans-serif;
}

```

J'ai développé ce projet en gardant à l'esprit que l'utilisateur devrait pouvoir se servir du site dans n'importe quel navigateur, et sur n'importe quel périphérique. En conséquence, j'ai développé le CSS du site en me servant des media-queries, afin de pouvoir modifier l'affichage en fonction de la taille de l'écran et donc du périphérique utilisé.

J'ai également fait en sorte que le comportement du site soit adapté à un maximum de périphériques différents. De plus j'utilise également le display flex afin de disposer les éléments de façon à ce qu'ils soient adaptable selon la taille de l'écran avec un "flex-wrap".

Par exemple ici, je modifie la disposition de ma div “” pour une taille d’écran inférieur ou égale à 991px

L’ensemble du site possède le panel de couleur jaune, blanc et gris les codes hexadécimal sont les suivant :

#ffffff

■ #FFFF00

■ #7F7E7E

■ #333333

Comme expliqué précédemment, mon JavaScript permettra de vérifier si les formulaires sont bien remplis lors de leur validation et si c’est le cas, d’effectuer une requête avec la base de données et d’afficher du contenu dynamiquement en fonction du résultat. (ex: login ou mot de passe incorrect, encadrement des input en rouge...).

4.1.3 Développement de la partie back-end

Je n’ai pas énormément touché à la partie back-end. Bien sur j’ai quand même dû y toucher pour certaines chose comme l’envoi de formulaire, ou l’intégration de donnée.

J’ai d’abord commencé à créer en collaboration avec Jonathan mon binôme la base de données avec phpmyadmin en prenant exemple sur notre maquette faite au préalable sur Genmymodel. Pour exemple, voici la table utilisateurs qui contiendra donc toutes les informations correspondantes aux inscrits du site.



	id	login	password	email	droit
<input type="checkbox"/> Éditer Copier Supprimer	1	Marjorie	\$2y\$12\$h16.bpWoWCkgn1UU.Sd4eAO9xNN5so2ihRDyYYp7ZG...	agence@bolikian.fr	1

Une fois la base de données créée, j'ai aidé Jonathan à développer la partie connexion/inscription afin que la cliente puisse accéder à la gestion de son site. Lorsqu'un utilisateur s'inscrit, il doit renseigner l'ensemble des informations suivantes : pseudonyme/login, email, mot de passe et confirmation de mot de passe. Une fois les données renseignées, nous avons récupéré le contenu des champs pour pouvoir les insérer dans la base avec des requêtes SQL.

Une fois les vérifications faites, je passe en données le contenu des champs et les envoie à un fichier php qui va effectuer les requêtes d'insertion en base de données.

En revanche si les champs sont vides, nous avons, en s'aidant du CSS, montrer qu'il faut remplir le champ.

Une fois inscrit, nous envoyons l'utilisateur sur la page de connexion (même fonctionnement que pour l'inscription).

La grande majorité des sites web dynamiques ou des applications ont besoin de réutiliser des parties de code identique à plusieurs endroits d'une même page, ou bien dans plusieurs pages différentes.

C'est le cas par exemple des bibliothèques de fonctions utilisateurs ou bien le cas présent ici pour utiliser mon header. Plutôt que de réécrire à chaque fois le code, il existe des fonctions (structures de langage en réalité) capables d'importer et exécuter le code à réutiliser dans la page. Il s'agit des fonctions `include()` et `require()`. J'ai pour ma part utilisé un `require()`.

Je peux donc ainsi intégrer un header ou un autre en changeant juste le fichier demandé à l'envoi :

```
<?php require_once 'header.php'; ?>
```

OU

```
<?php require_once 'header1.php'; ?>
```

Veille et sécurité :

Concernant la sécurité, il est tout d'abord important de vérifier les données rentrées par l'utilisateur dans les formulaires, que ce soit pour se connecter ou pour envoyer un message. Car ses données sont ensuite récupérées et insérées dans la base de données, il faut donc faire attention à ce qui est récupéré. C'est pourquoi j'utilise PHP afin de vérifier que tous les champs soient bien remplis dans un premier temps.

De plus, lorsque notre seul et unique utilisateur souhaite se connecter, il est pour nous primordial que celle-ci soit la bonne personne. Pour ce-là nous avons rajouté des droits pour être sûre qu'il s'agisse bien d'elle.

En parallèle, je regarde si les mots de passes correspondent lors de l'inscription (mot de passe et confirmation mot de passe), si la taille du mot de passe est supérieure ou égale à 8 (pour plus de sécurité) et si le login lui est supérieur ou égale à 5 caractères.

Si toutes les conditions sont remplies, nous cryptons le mot de passe de l'utilisateur afin de stocker dans la base de données un mot de passe que personnes ne peut voir, ni même nous. Pour cela nous utilisons la fonction "password_hash" qui va donc créer une clé de hachage pour le mot de passe d'une longueur de 60 caractères et qui ressemblera à ceci :

“\$2y\$12\$h16.bpWoWCKkgn1UU.Sd4eAO9xNN5so2ihRDyYYp7ZG8slJwtKIMK”

Après afin de vérifier la correspondance entre un mot de passe crypté et un mot de passe renseigné par l'utilisateur lors d'une connexion, j'utilise la fonction “password_verify”.

Concernant la page contact, je regarde grâce à une fonction si les champs de mander sont bien remplis (pour éviter des spams ou autre) mais aussi si par exemple l'email possède le bon format “...@.. .fr/.com”. Sinon j'indique à l'utilisateur que le format n'est pas le bon.

Côté utilisateur :

The screenshot shows a web page with a contact form. On the left, there is a yellow diagonal striped decorative element. The form is titled 'Contact' and includes the following fields: 'Nom *', 'Email *', 'Sujet', and 'Message'. A yellow 'Envoyer' button is at the bottom right of the form. Below the form, a warning message reads: '⚠️ Votre email doit comporter un arobase.' On the left side of the form, there is a greeting 'Bonjour,' followed by the text: 'N'hésiter pas à me contacter via ce formulaire pour prendre un rendez-vous ou si avez des questions.'

Cette erreur correspond au mauvais format renseigné par l'utilisateur.

Côté code :


```

<?php
if (isset($_POST['envoyer'])) {
    $position_arobase = strpos($_POST['email'], '@');
    if ($position_arobase === false)
        echo '<p id="erreurcon"> ⚠ Votre email doit comporter un arobase.</p>';
    else {
        $retour = mail('ch.mayer@hotmail.fr', 'Envoi depuis la page Contact', $_POST['message'], 'From: ' . $_POST['email']);
        if($retour)
            echo '<p>Votre message a été envoyé.</p>';
        else
            echo '<p id="erreurcon"> ⚠ Erreur.</p>';
    }
}
?>

```

Une fois tous les champs requis complétés je les envoie à mon fichier php qui lui va exécuter les requêtes SQL.

4.2 Jeu d'essai

Pour le jeu d'essai nous n'avons pas eu à tester beaucoup de fonctionnalités sachant que la seule que nous "avons" est la connexion et l'utilisation du panel d'administrateur. Pour se faire nous avons directement fait tester son maniement par la responsable du site.

Pour le reste nous avons demandé à plusieurs personnes non rattaché à notre formation de circuler sur le site et de nous donner des avis. Sans aucune consignes ou attentes particulières pour savoir si l'effet clair, dynamique et fluide du site était efficace.

C'est après avoir recueilli tous les retours et avoir fait quelques retouches que nous avons approuvé le projet et que nous avons trouvé que notre jeu d'essai était concluant.

4.4 Difficultés rencontrées

La première difficulté que nous avons rencontré a sûrement été la contestation de la cliente. Ca n'a pas été non plus une grande difficulté mais nous avons un peu été déçu de ne pas avoir de suite ou de réponse négative à nos propositions d'ajouts de fonctionnalité et options.

le deuxième problème a été la perte de notre 3ème camarade qui a dû redoubler son année et donc nous abandonner sur le projet.

Mais le plus gros du soucis a sûrement était les délais. Ce projet a dû être réaliser en partie pendant le confinement (qui n'arrange pas les choses pour tout ce qui relève de la pédagogie de projet en groupe) sans compter, mais surtout pendant le déconfinement. Je pense peut être parler pour beaucoup de personnes mais entre le mental et les pertes autour de nous, tout ça n'a pas été simple.

Sans oublier que ce projet s'est déroulé durant notre formation. En plus du projet professionnel, je devais également réalisé des projets à côté pour l'école car nous apprenions encore de nouveaux langages comme JavaScript / jQuery / json, préparer l'examen et surtout chercher une alternance. Malgré notre logiciel de planification, tout gérer et jongler entre toutes nos tâches n'était pas très évident.

C'était aussi pour moi une toute nouvelle expérience. Je n'avais pas l'habitude de travailler sur un projet concret avec cette sensation de ne pas pouvoir vraiment me tromper ou faire de faute.

Au niveau des tâches à effectuer, j'ai rencontré bien sur des bugs mais j'ai aussi voulu utiliser des compétences dont je ne m'étais pas énormément servis encore, comme Bootstrap ou Javascript. J'ai donc persévérer pour arriver à mes fins, mais j'ai aussi pu compter sur mes camarades de promo pour m'expliquer certaines choses.

4.5 Sites anglophones

Le monde du développement web contient énormément de langages différents mais tous ces langages sont en partie guidé par l'anglais. Une connaissance en anglais doit alors être un minimum assimilé, même en ce qui concerne les futures clients. Très pratique dans la vie de tous les jours du développeur web, j'en ai moi même fais les frais, cette communauté ne partage pratiquement qu'en anglais. Que ce soit sur les forums ou sur certains site de référence comme <https://www.w3schools.com/>

PHP Variables

A variable can have a short name (like x and y) or a more descriptive name (age, carname, total_volume).

Rules for PHP variables:

- A variable starts with the **\$** sign, followed by the name of the variable
- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are case-sensitive (**\$age** and **\$AGE** are two different variables)

Remember that PHP variable names are case-sensitive!

Variables PHP

Une variable peut avoir un nom court (comme x et y) ou un nom plus descriptif (age, nom de voiture, volume total).

Règles pour les variables PHP:

- Une variable commence par le signe \$, suivi du nom de la variable
- Un nom de variable doit commencer par une lettre ou le tiret du 8
- Un nom de variable ne peut pas commencer par un nombre
- Un nom de variable ne peut contenir que des caractères alphanumériques et des tirets du 8 (Az, 0-9 et _)

- Les noms de variables sont sensibles à la casse (\$age et \$AGE sont deux variables différentes)

N'oubliez pas que les noms de variables PHP sont sensibles à la casse!

CONCLUSION

J'étais vraiment motivé à l'idée de réaliser un projet que je trouvais en plus de ça vraiment très intéressant. Cependant, je ne me rendais pas compte ce que cela impliquerait en terme de temps et j'ai passé beaucoup de temps à coder sur ce projet, à l'école comme à la maison, la semaine comme le week-end.

Finalement, ce projet m'a dans un premier temps permis de mettre en pratique mes acquis que j'ai pu avoir durant mon année de formation et de les consolider. De plus, j'ai pu apprendre à travailler avec un certain rythme et une certaine assiduité.

La réalisation de ce projet permet de savoir à quoi s'attendre en ce qui concerne le monde du travail car il a fallu créer un site en fonction d'une demande d'un client, d'un temps donné (4 à 6 mois) et donc d'un cahier des charges, c'était donc très formateur.

Je suis très contente de la finalité du site. La cliente en est elle aussi très contente, je pense que c'est le plus important. Un travail qui n'a pas été de tout repos mais finis et pour moi réussi.

J'ai maintenant pour but de me professionnaliser dans le front-end pour réaliser ce qui me plaît le plus, c'est à dire pouvoir faire ressortir l'identité de mes clients tout en apaisant mon côté pointilleuse.

ANNEXES



M — Marjorie Bollikian
Architecte dplg
B —

Introduction au site