|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | |  |  | |
| *Nom de naissance* |  | | | **MEZIANE** | |
| *Prénom* |  | | | **Maamar** | |
| *Adresse* |  | | | **17 Avenue Charles Camoin- 13013 Marseille** | |

|  |
| --- |
| **Titre professionnel** |
|  |
| **Développeur Web et Web Mobile** |

**DOSSIER-PROJET**

Remerciement

Tout d’abord, je souhaite remercie les financiers de cette formation (le Conseil Régional et le Pole Emploi), ensuit je remercie la conseillère de AFPA (Ilham ZIRAOUI).

Merci particulièrement à mes formateurs Nicolas et Yannis pour leur disponibilité et leur pédagogie et leur soutien dans cette formation pour le titre professionnel développeur web et web mobile.

Même si je sais que l’on ne peut pas tout apprendre et tout connaître en dix mois de formation, que l’on peut toujours apprendre et s’améliorer, que ce soit avec des ressources en lignes, comme des tutoriels et vidéos en lignes

Ensuite, je remercie Slim TOHAMI de m’avoir accepté en stage (en télétravail) dans son Boite. Ce n’est pas facile de trouver un stage, par rapport aux autres années. Je le remercie aussi pour tous les conseils qu’il a pu me donner que ce soit dans la création de site et aussi pour plus tard le projet de créer mon agence web en tant qu’entrepreneur.

Sommaire

[1. Présentation 4](#_Toc89634345)

[1.1 Présentation personnelle 4](#_Toc89634346)

[1.2 Présentation du centre de formation 4](#_Toc89634347)

[1.3 Présentation de la formation DWWM 4](#_Toc89634348)

[2. Stage en entreprise 6](#_Toc89634349)

[2.1 Remerciement 6](#_Toc89634351)

[2.2 Introduction 6](#_Toc89634352)

[2.3 Projet de stage 6](#_Toc89634353)

[2.3.1. Intégration d’API Stripe sur le site de « abla.io » 6](#_Toc89634354)

[Paiement 8](#_Toc89634355)

[Approvisionnement 9](#_Toc89634356)

[2.3.2..création un blog pour l’application Abla « abla.io » 10](#_Toc89634357)

[3. Les projets personnels 13](#_Toc89634358)

[3.1 Activités type n°1 : Développer la partie front-end d’une applications web/web mobile en intégrant les recommandations de sécurité « Site vitrine de société FMHYDRO » 13](#_Toc89634359)

[3.1.1. Compétence N° 01 Maquetter une application 13](#_Toc89634360)

[3.1.2Compétence N° 02 : Réaliser une interface utilisateur web statique et adaptable 16](#_Toc89634361)

[3.1.3 Compétence N° 03 : Réaliser une interface utilisateur avec une solution de gestion de contenu -CMS 19](#_Toc89634362)

[3.2. Activités type N°2 : Développer la partie Back-End d’une applications web/web mobile en intégrant les recommandations de sécurité 21](#_Toc89634363)

[Projet blog Meziane dev 21](#_Toc89634364)

[3.2.1. Compétence N° 05 Crée une base de données (blog) 21](#_Toc89634365)

[3.2.2. Compétence N° 06Développer les composants d’accès aux données (blog) 25](#_Toc89634366)

[3.2.3. Compétence N° 07  Développer la partie Back-End d’une application web ou web mobile (blog) 27](#_Toc89634367)

[4. Jeu d’essai 33](#_Toc89634368)

[4.1 La vérification des champs vides 34](#_Toc89634369)

[4.2 Le format du champ de type email 35](#_Toc89634370)

[4.3 La concordance entre les deux champs mot de passe 36](#_Toc89634371)

[5. Veille technologique 37](#_Toc89634372)

[5.1. Contexte 37](#_Toc89634373)

[5.1.1. Extrait du site anglophone 37](#_Toc89634374)

[5.1.2. Traduction en Français 38](#_Toc89634375)

[Conclusion 39](#_Toc89634376)

# 1. Présentation

## 1.1 Présentation personnelle

Je m’appelle Maamar MEZIANE, 40 ans, j’habite a Marseille 13013. J’ai pratiqué le métier de la comptabilité et aussi l’administration générale dans l’université pendant près 15 ans . en Algérie. Ayant l’impression d’avoir fait Jérômele tour de ces professions et étant attiré depuis de longue années par les métiers de l’informatique j’ai entamé une reconversion professionnelle vers la profession de développeur web et web mobile.

Ce demain suscité mon intérêt car malgré son expansion accrue, il reste un « territoire » inconnu.

De plus, le développement peut être utilisé pour le web, mais ouvre également la porte à d’autre horizons, comme la programmation d’application, mais également de jeux vidéos, ou encore dans la cyber sécurité, et bien d’autre encore.

C’est donc pour ces raisons que je me suis orienté vers une formation en tant que développeur web et web mobil, au sien de centre de formation de l’AFPA de Saint Jérôme 13013.

## 1.2 Présentation du centre de formation

l’AFPA (agence pour la formation professionnelle des adultes) dispose d’un réseau de centres, proposant diverses formation.

Ayant choisi le développement web, le centre correspondant à mon cursus se situe à Saint Jérôme, dans le 13 ème arrondissement de Marseille.

J’ai apprécié d’avoir été affecté à ce centre car l’environnement me rappelait mess années à la faculté. Avant de débuter ma formation, j’ai bénéficié d’une remise a niveau pour une durée de huit semaines, afin de me préparer correctement à ma période d’apprentissage. Dans le centre, mes horaires étaient de 08h00 à 12h00 et 13h00 a 17h00 du lundi au Jeudi, puis de 08h00 à 12h00 le Vendredi

## 1.3 Présentation de la formation DWWM

Ma formation de développeur Web & mobile a commencé le 29 Mars 2021,nous étions une classe d’une quinziane de personnes pour un formateur. En premier temps, notre objectif était de voir les bases de HTML et CSS pour pouvoir coder des pages statiques simples, comme par exemple un site vétrine d’une société de forage d’eau. Les bases en dans ces deux langages de balisages ont été enseignés, pour se consacrer plus dans des langages tels que le JavaScript ou le PHP Egalement en a penchez sur les CMC tels que Wordpress ou Prestashop, ou nous avons pu les installer sur nos postes et en voir certaines fonctionnalités. Le fait de pouvoir réalisé une interface web rapidement avec un CMS est très intéressante et sera indispensable pour mon avenir professionnel, cependant ce qui m’intéressait davantage était le code en lui-même. Malgré les difficulté que j’ai rencontrées dans ces conditions et avec cette méthodologie, mais avec l’acharnement et de nombreuse recherches , j’au pu apprendre à apprendre et développer une certaine autonomie, que j’ai développé certaines affinités et compétences dans le codage avec PHP +SQL pour réaliser du Bach-end et pouvoir créer des bases de données en MySQL, et du JavaScripts également pour pouvoir effectuer quelques fonctionnalités frond-end . Pour renforcer mon apprentissage, j’ai pu acquérir quelques compétences et une certaine affinité, lors de ma période de stage en entreprise, dans le codage Back-end avec des API comme **STRIPE** et aussi des CMS comme **JEKYLLERB** qui est en fait des sites dynamiques.

# 2. Stage en entreprise

**Stage effectué à l’agence**

**Astra Porta**

Du 17/11/2021 AU 10/03/2021

## 2.1 Remerciement

Dans un premier, j’aimerai remercier mon tuteur et le directeur de la boite M.Slim TOUHAMI m’avoir accepté en tant que stagiaire dans son Boite qui m’a formé et accompagné tout au long de cette expérience professionnelle.

## 2.2 Introduction

Dans le cadre de ma formation de développeur web et web mobile, j’ai effectué un stage au sein de l’agence Astra Porta durant ce stage était d’apporter mon aide pour réalisé deux projets

Ce stage m’a permis d’acquérir des connaissances en matière de professionnalisation et notamment une connaissance du travail qui est pour moi très importante car ce fut en quelque sorte mon premier stage dans le domaine.

## 2.3 Projet de stage

Lors de mon stage au sein d’Astra Porta, j’ai réalisé deux projets.

### 2.3.1. Intégration d’API Stripe sur le site de « abla.io »

**« Abla Analytics » Abla.io** c’est une application web développée par **la start-up Marseillaise Astra Porta.** **Abla Analytics** est un outil de mesure d’audience conforme aux exigences **réglementaires** et aux recommandations de la CNIL. L’ensemble des données collectées par l’intermédiaire d’Abla Analytics demeure la propriété exclusive de nos clients et ne peut en aucun cas être utilisé à d’autres fins par l’éditeur. Notre solution de mesure d’audience est proposée en trois versions : **Abla Analytics**, **Abla Analytics +** et **Abla Analytics x.** Pour garantir la vie privée des visiteurs, par défaut et sur toutes ces versions :

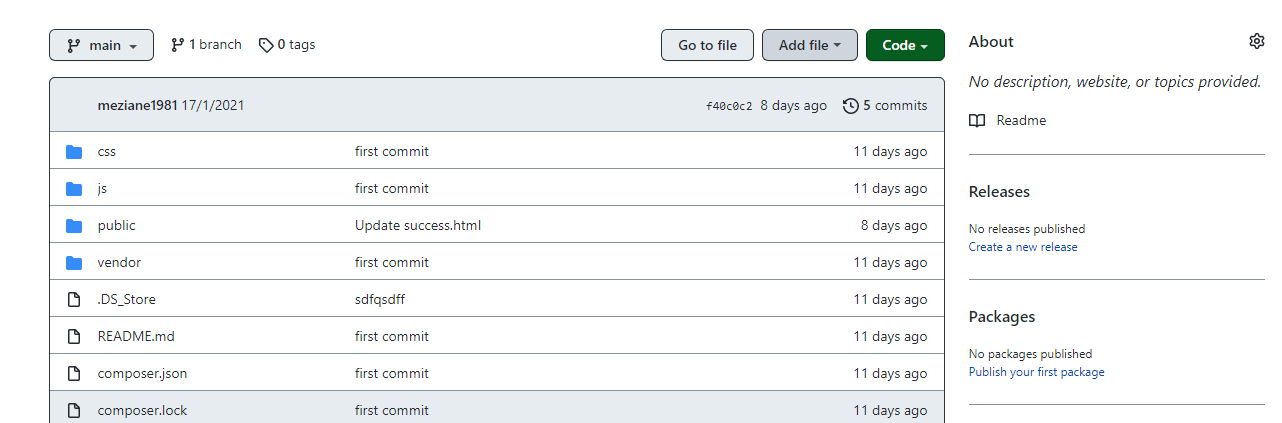
- Il est impossible de créer des cohortes de visiteurs pour l’affichage de contenus différenciés

- La collecte d’informations de localisation est limitée au niveau du pays

- Les adresses IP des visiteurs sont anonymisées instantanément (impossibilité d’identifier un visiteur)

- Il est impossible de suivre la navigation d’un visiteur sur plusieurs sites ou/et avec différents terminaux

**Stripe** c’est Une plateforme de paiement complète pour les entreprises à forte croissance. Acceptez des paiements et transférez des fonds dans le monde entier grâce à la puissance des API et des solutions logicielles de Stripe, conçues pour vous aider à accroître vos revenus.



**Image N°1 : dépôt de projet ABLA - STRIP**

#### Fonctionnent des abonnements

Avec les [abonnements](https://stripe.com/docs/billing/subscriptions/creating) , les clients effectuent des paiements récurrents pour accéder à un produit. Les abonnements vous obligent à conserver plus d'informations sur vos clients que les achats ponctuels, car vous devez les facturer automatiquement à l'avenir.

#### [Objets d'abonnement](https://stripe.com/docs/billing/subscriptions/overview" \l "subscription-objects)

#### Utilisez les objets API principaux suivants pour créer et gérer des abonnements :

[**Les clients**](https://stripe.com/docs/api/customers) sont vos acheteurs. L'objet Client stocke des informations telles que les noms, les adresses e-mail et les modes de paiement (cartes de crédit, cartes de débit, etc.).

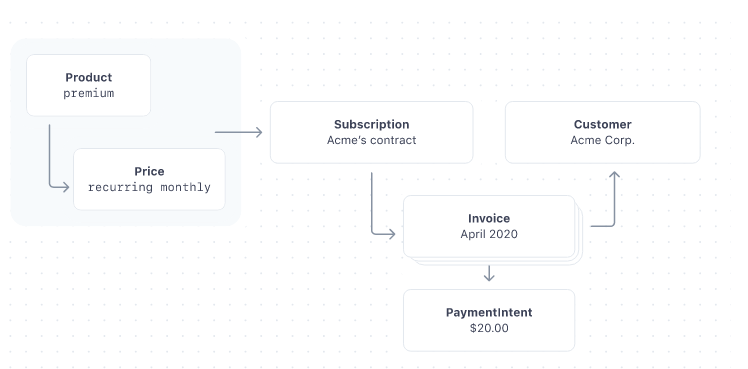
[**Les produits**](https://stripe.com/docs/api/products) sont ce que vous vendez.

[**Les prix**](https://stripe.com/docs/api/prices)représentent combien et à quelle fréquence vous facturez un produit. Vous pouvez définir plusieurs prix pour un produit afin de pouvoir facturer différents montants en fonction de la devise ou de l'intervalle (mensuel, annuel, etc.).

[**Abonnements**](https://stripe.com/docs/api/subscriptions) représentent l'accès de vos clients à un produit et vous obligent à créer un client et un mode de paiement. Le statut d'un abonnement indique quand fournir l'accès à votre service à un client.

Stripe génère des [**factures**](https://stripe.com/docs/api/invoices) lorsqu'il est temps de facturer un abonnement à un client. Les factures comportent des postes, des taux de taxe et le montant total dû par un client.

[**Intentions de paiement**](https://stripe.com/docs/api/payment_intents) représentent l'état de toutes les tentatives de paiement d'une facture.



**Image N°2 : procédure d’abonnement**

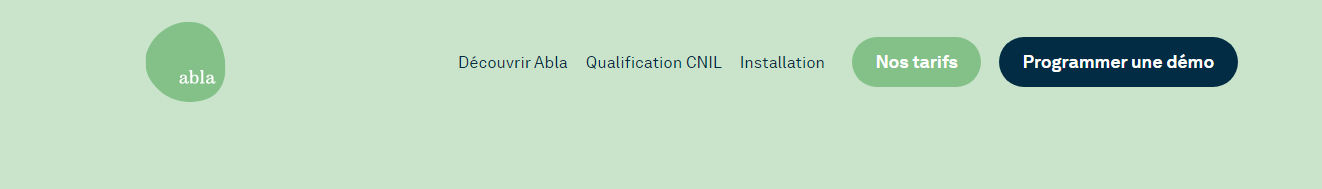
### Paiement

Le formulaire de paiement recueille un nom et des informations sur la carte. Stripe héberge ce formulaire, et c'est l'une des fonctionnalités clés qui vous permet de collecter des paiements et de rester conforme à la norme PCI. En cliquant sur le bouton S'abonner :

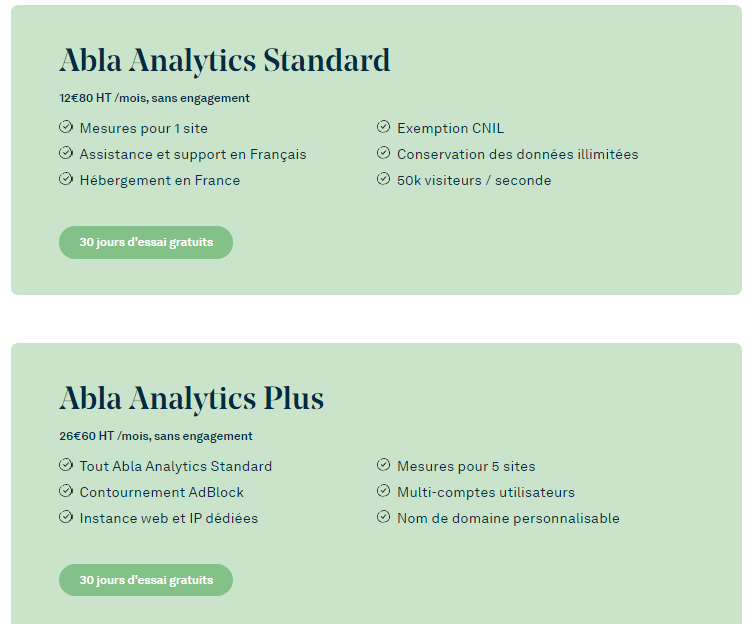
* Crée un nouvel abonnement, avec les identifiants client et prix.
* Génère une facture pour le cycle d'abonnement initial.
* Collecte les détails de paiement et paie la facture.
* Définit le mode de paiement comme mode de paiement par défaut pour l'abonnement, ce qui est obligatoire pour les paiements ultérieurs.

### Approvisionnement

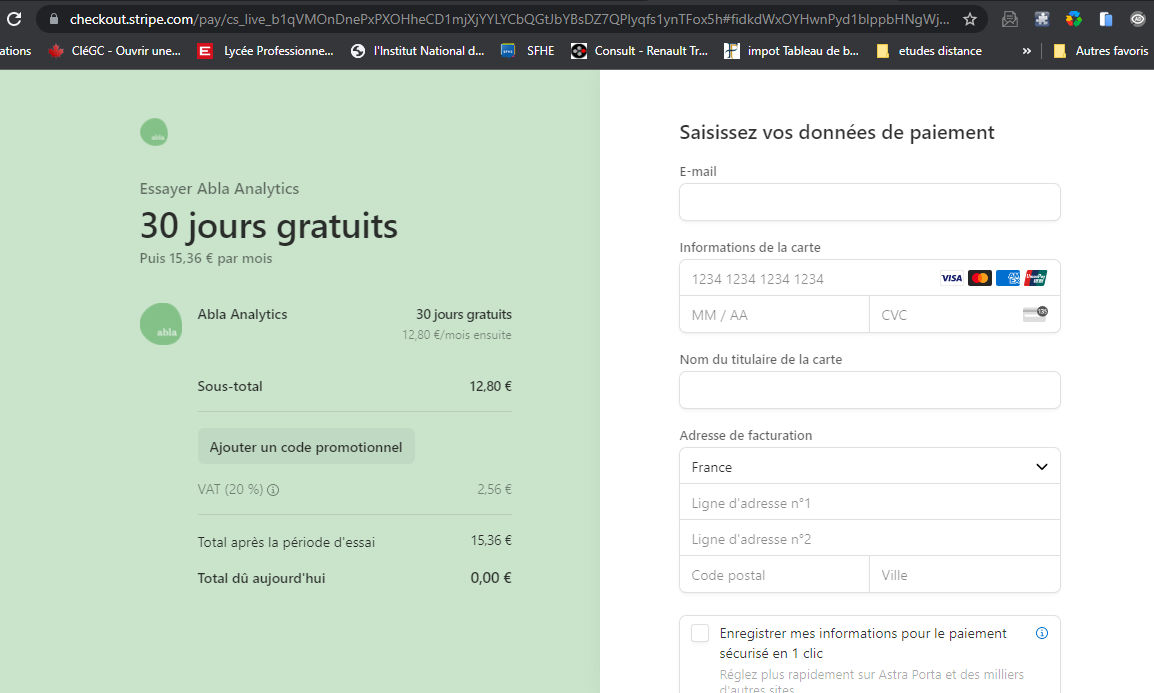
* Une page de confirmation s'affiche après un paiement réussi. À ce stade, il est sûr de provisionner le produit pour le client. Cela signifie généralement :
* La vérification du statut de l'abonnement est active.
* Accorder au client l'accès aux produits et fonctionnalités auxquels il est abonné.



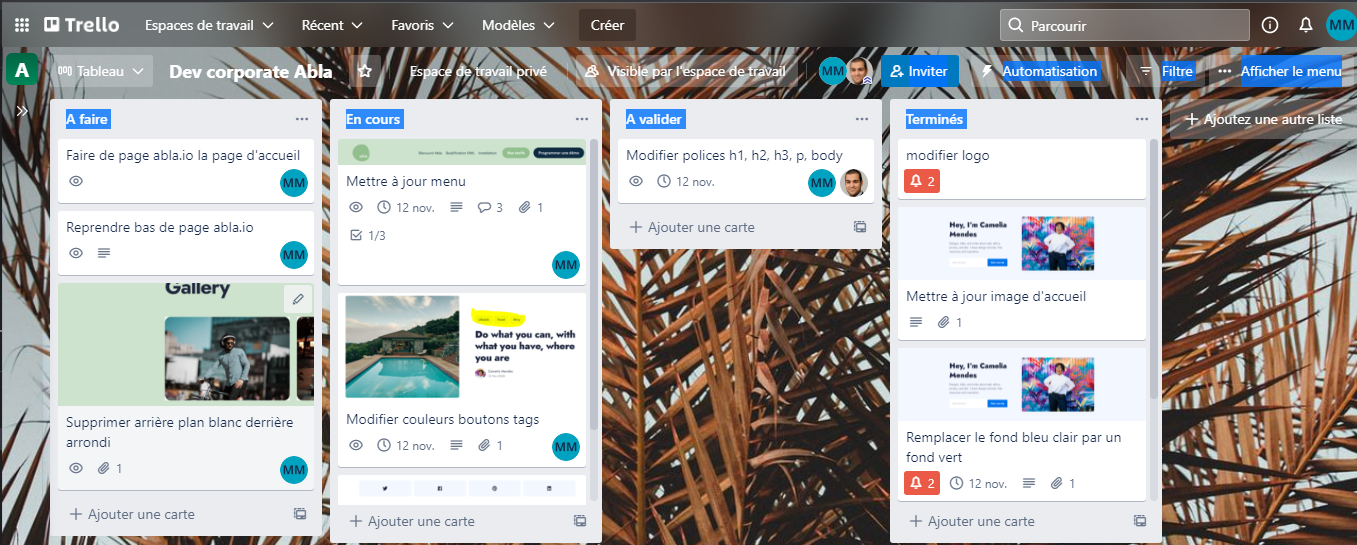
**Image N°3 : site abla.io**



**Image N°4 : les tarifs d’abla analytique**



**Image N°5 : l’interface de paiement d’abonnement au niveau de strip**

2.3.2..création un blog pour l’application Abla « abla.io »

**Image N°1 :Trello mes taches**

Pour créé le blog de site on a utilisé le CMS jekyll (Jekyll est un générateur de site statique développé par Tom Preston-Werner, le fondateur de Github. C'est un logiciel libre écrit en langage Ruby. En 2017 il est considéré comme le générateur de site statique le plus populaire)

On m’a confié la tâche de modifier le thème Jekyll pour ce la j’ai installer Ruby Gem, après j’ai installé tous les pré requis :

Installez les [gemmes](https://jekyllrb.com/docs/ruby-101/#gems) jekyll et bundler .

gem install jekyll bundler

Créez un nouveau site Jekyll sur ./myblog.

jekyll new myblog

Accédez à votre nouveau répertoire.

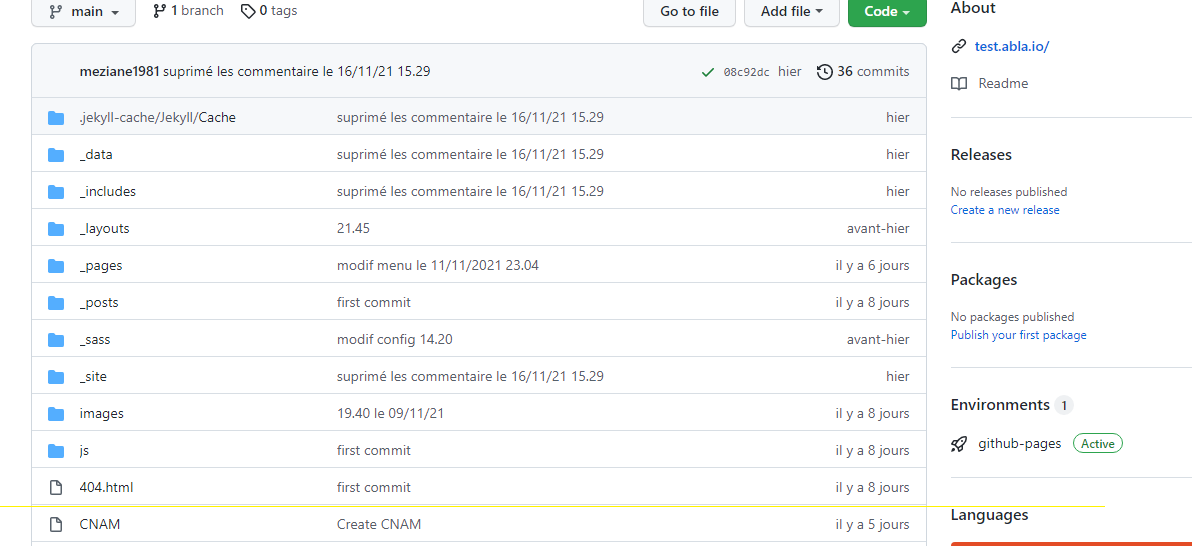
cd myblog

Construisez le site et rendez-le disponible sur un serveur local.

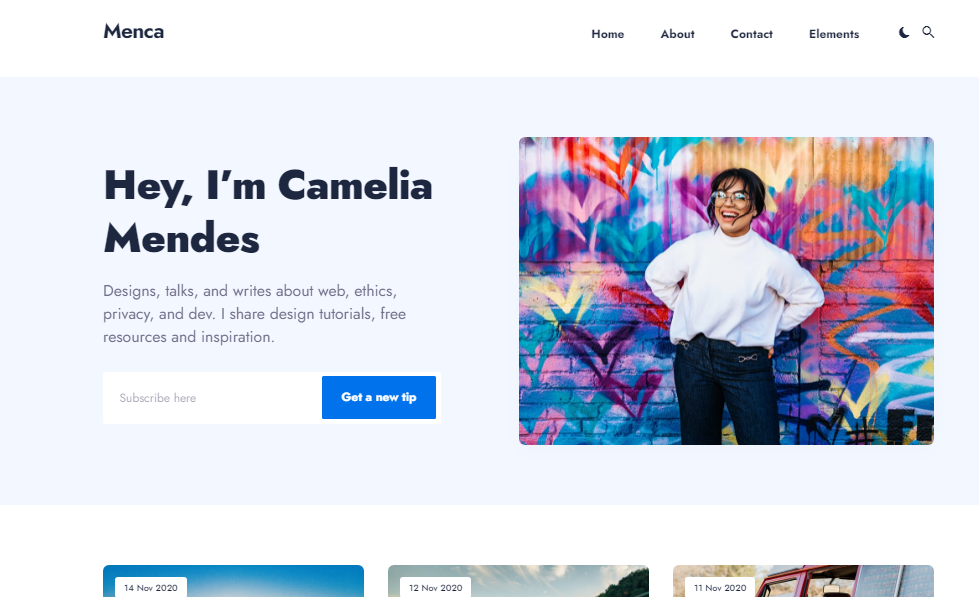
bundle exec jekyll serve

Accédez à [http://localhost:4000](http://localhost:4000/)

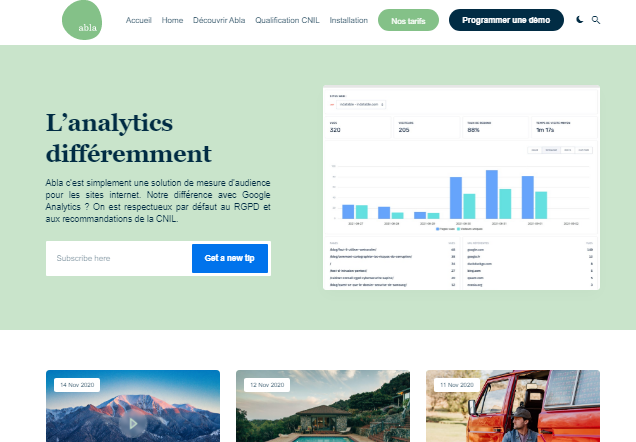
Apres quand j’ai fais tous les instructions j’ai suivi les tache apartire de Trello



**Image N°2: dépôt dans github**



**Image N°3 : Thème avant modification**

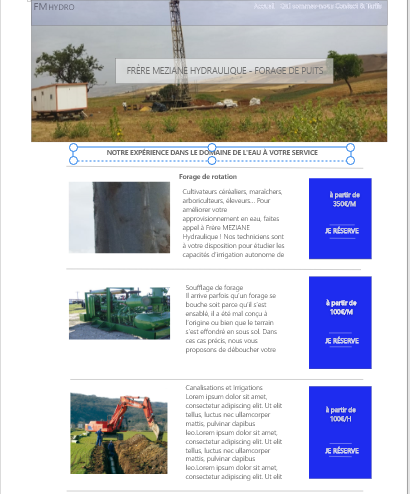


**Image N°4 :Thème après modification « Blog.abla »**

# 3. Les projets personnels

## 3.1 Activités type n°1 : Développer la partie front-end d’une applications web/web mobile en intégrant les recommandations de sécurité « Site vitrine de société FMHYDRO »

### 3.1.1. Compétence N° 01 Maquetter une application

La réalisation des maquettes a été effectué avec ADOBE XD qui est un outille de conception qui permet de maquetter des prototypes des applications web





**Image N°1 : Maquette de site « fmhydro »**

### 3.1.2Compétence N° 02 : Réaliser une interface utilisateur web statique et adaptable

##### **a. Expression de besoins :**

J’ai réalisé ce site pour répondre au besoin de la société de forage pour la visibilité sur internet en tant que entreprise de forage des puits.

**\*-Le cahier des charges adéquat à ce projet est le suivant :**

* **Objectifs du site**

Rendre visible sur internet un format web de site de l’entreprise, en présentant donc les différent services de l’entreprise, travaux réalisés dans tous le territoire nationale, et présentation de l’entreprise.

* **Combien de pages et combien de services le site va contenir ?**

Le site dispose de trois pages, ou les renseignements concernant l’entreprise.

* **Périmètre du projet :**

Le site sera uniquement en français, et responsive. Entant donné que ce sera une interface statique, il ne contiendra aucune base de données.

Concernant la sécurité, elle sera moins importante car aucune base de données ne sera présente, cependant l’hébergeur empêchera la modification du contenu des pages, ce qui assurera une sécurité minimale.

* **Maquettage :**

Le site contiendra trois pages réalisé avec adobe xd afin de disposer d’une premières visualisation de l’interface du site.

* **contraintes techniques :**

L’hébergement du site et l’entretien sera effectué par le client.

##### **b. Documentation de spécifications fonctionnelles :**

Afin de pouvoir compléter le cahier des charges, un document des spécifications fonctionnelles peut être réalisé sous forme de schéma explicatif comme un **impact-mapping** par exemple.

Un impact mapping est un schéma résumant le produit correspondant au besoin du client en répondant aux questions Pourquoi ? Qui ? Comment ? Quoi ?

|  |  |  |  |
| --- | --- | --- | --- |
| Pourquoi ? | Qui ? | Comment ? | Quoi ? |
| Augmenter visibilité du client sur internet | -Développeur web junior  -Divers utilisateurs & entreprises | - Mettre en valeur les compétences de  -Pouvoir consulter le différent service donné par l’entreprise | -Afficher une page de renseignement & plusieurs onglets pour les services proposés et les projets réalisés  -proposé les prix des services  - consulter des renseignements, compétences & expérience pour un éventuel contact  - contacte l’entreprise pour connaitre le faire le devis |

**Tableau : impact-mapping**

**Ici,** l’impact mapping répond aux questions suivantes :

* **Pourquoi** le client souhaite obtenir un site web ?

Pour augmenter sa visibilité sur internet

* **Qui** seront les acteurs de ce site ?

L’entreprise de forage, ainsi que les personnes physiques et morales qui cherche des services concernant le forage et travaux public.

* **Comment** le site augmentera la visibilité du client ?

En mettant ses compétences de l’entreprise ainsi que ses expériences en valeur, et en proposant aux visiteurs de consulter

* Le site va faire **Quoi** pour augmenter la visibilité du client ?

Il va afficher un page d’accueil, il ya des défirent prix proposé par l’entreprise, et aussi des renseignements concernant les défirent projet réalisé.

Une page qui somme nous ? et page contacte pour les devis.

**\*.Document de spécifications techniques**

**Enfin,** pour finaliser le cahier des charges de se projet, un document de spécifications technique est nécessaires pour préciser quels outils vont êtres utilisés, comment ils le seront, pour mener à bien le développent du site web.

Voici les différents outils ayant servi pour la création de l’interface :

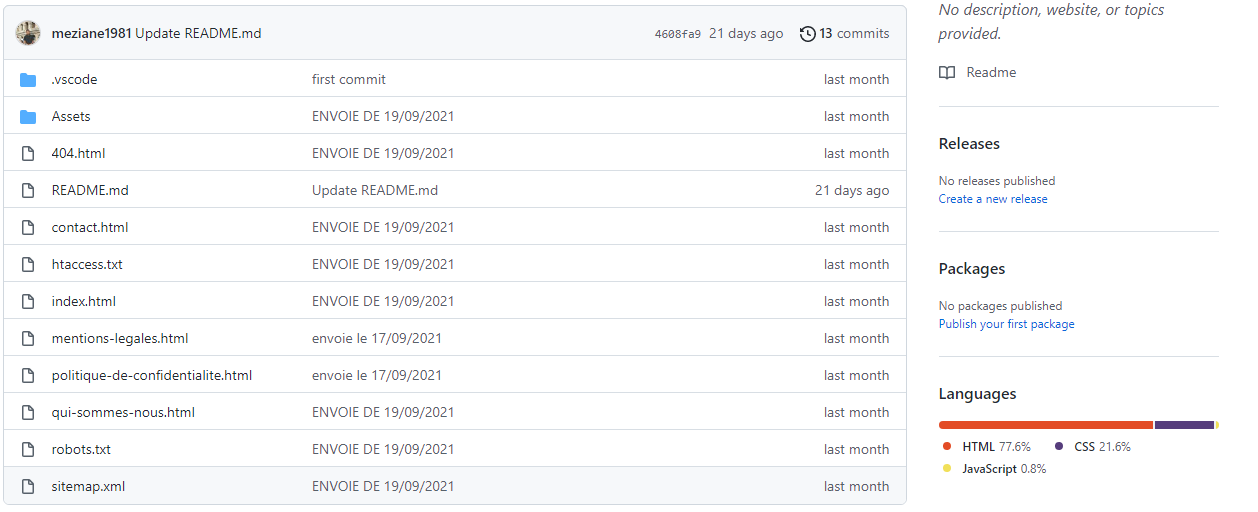
* Visual studio code en tant qu’éditeur de texte
* HTML5 : langage de balisage pour définir la structure du site
* CSS 3 : langage de balisage pour mettre en page le site
* JavaScripts : langage de script utilisé ici pour créer le menu burger

Pour des facilites de compréhension et de lecture, la suite du document de spécification techniques comportera des illustrations argumentatives au déroulé du projet.

Maintenant que les différents outils ont été énoncés, il est temps de vous expliquer leur impact dans le développent du projet.

Pour commencer, mon arborescence est structurée de la manière suivant :

* la racine contient mes différents fichiers HTML, qui vont coder la structure de mon site, ainsi les fichiers robots.txt[[1]](#footnote-1), sitemap.xml[[2]](#footnote-2) et .htaccess.txt[[3]](#footnote-3).
* contient le dossier **Assets** qui contient les dossiers images, CSS et JS.



**Image N°1 : dépôt de projet sur Github « site vitrine de fmhydr»**

Il est responsive ce qui facilite grandement le développement à destination de multi support, téléphone, tablette, ordinateur et grand écran, j’ai du intégrer les fichiers JS pour le menu burger.

### 3.1.3 Compétence N° 03 : Réaliser une interface utilisateur avec une solution de gestion de contenu -CMS

J’ai réalisé la même site vitrine pour l’entreprise Fmhydro, j’ai développé avec plusieurs pages web et aussi avec coté vente de matériel de forage bien détailler, j’ai publier en ligne  «‘https://fmhydro.alwaysdata.net’», j’ai utilisé WordPress , j’ai installer manuellement. Dans ce cas la marche à suivre se décompose en 4 étapes :

-Téléchargé Wordpress.

-Créer une base de données.

-Envoyer WordPress sur le serveur (j’ai utilisé ALWAYSDATA.NET).

-Initialiser WordPress grâce à un script d’installation.

Pour paramétrer WordPress, je vais dans l’administration de mon site «https://fmhydro.alwaysdata.net/wp-dmin» :

- Régler les «permaliens»

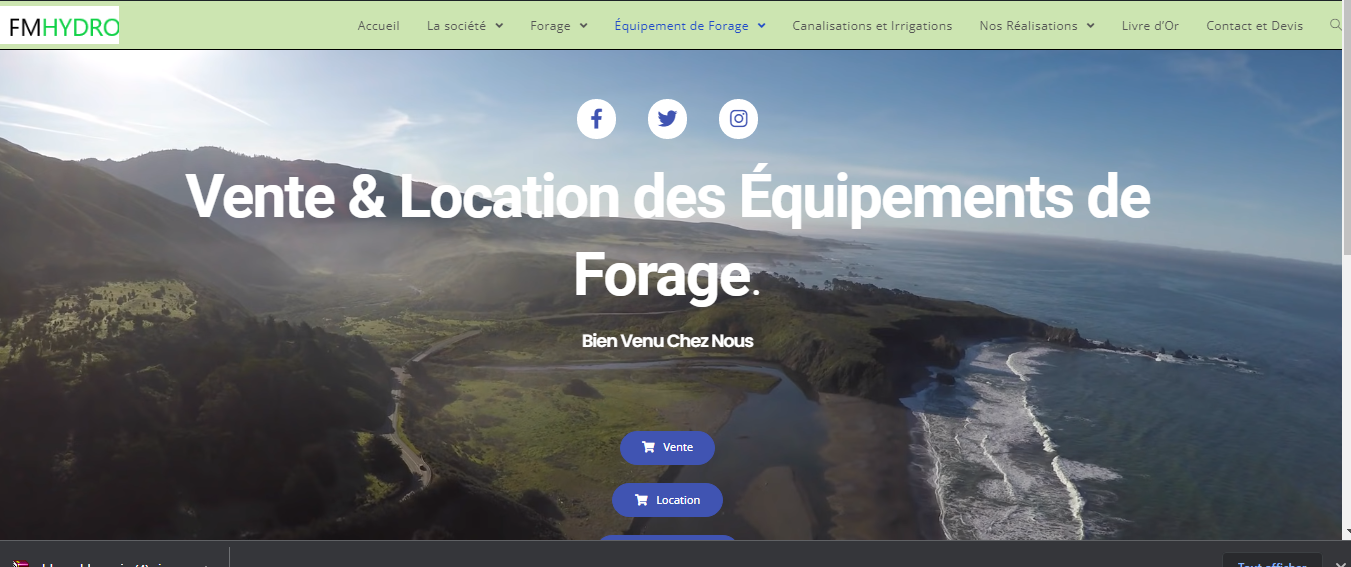
- Paramétrer le menu principal

- Gérer les «widgets»

-Choisir un thème, après j’ai installé « Elementor », et j’ai développer toutes mes pages web.



**Image N°1 : page d’accueil avec cms « fmhydro »**



**Image N°2 : page Vent & Location de site vitrine « fmhydro »**

## 3.2. Activités type N°2 : Développer la partie Back-End d’une applications web/web mobile en intégrant les recommandations de sécurité

## Projet blog Meziane dev

Suite au recueil des informations et la réalisation des cahiers des charges, J’ai développé sur serveur local un blog avec des articles et une section de commentaire pour chaque article. Dans cette application j’ai utilisé le langage **PHP**, avec **Framework Materialize** j’ai utilisé **MAILDEV** pour teste la connexion avec validation des mots de passe. Mon blog a été versionné pour permettre une sauvegarde avec traçabilitéé grâce à Github. Aussi il est possible, seulement pour l’administrateur du site, de créer, modifier et supprimé des articles, il ya également un système d’inscription /connexion pour ajouter des modérateurs.

### 3.2.1. Compétence N° 05 Crée une base de données (blog)

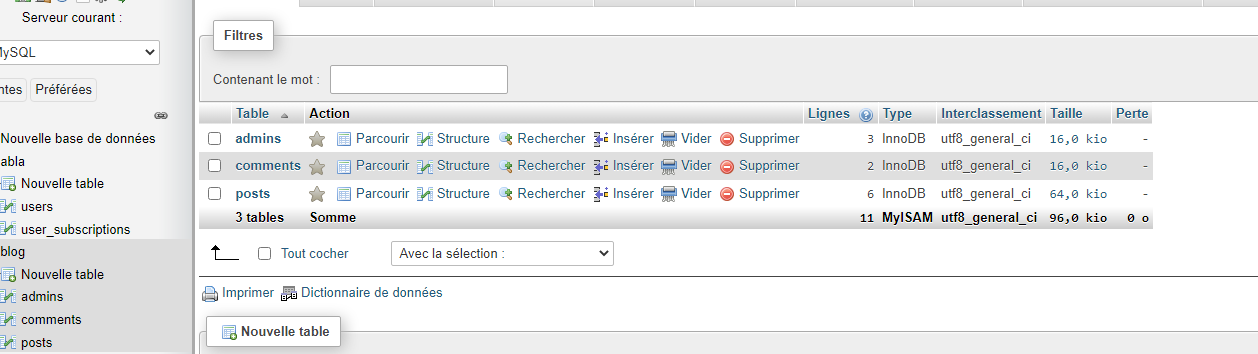
Pour les besoins de mon site, j’ai du crée une base de données, mais avant de réaliser cette tache j’ai utilisé le système de gestion de bases de données relationnelles SQL open source développé et supporté par Oracle.

Pour créer ma base de données j’ai d’abord commencé par rédiger nos idées sur un document, Une fois que j’ai finalisé ma base de données.

Pour ce faire, l’interface **phpMyAdmin** sera sollicitée, après avoir accéder à **phpMyAdmin (à travers WAMP),** j’ai créé une base de données que j’ai nomé **« Blog ».**

Suite à cela, j’ai dû créer chaque table individuellement avec les colonnes la composant, en respectant certains paramètres, tel que le type de données qui y serait insérée dans la table (du texte, des chiffres, une date etc), la longueur de la donnée, si elle était une clé primaire ou étrangère etc.

Voici une illustration des différentes tables composant la base de données :



**Image N°1 : La base de Données de Blog Meziane DEV**

Ainsi que l’illustration d’une structure d’une table, avec différents colonnes :



**Image N°2 : La Table de Posts Blog Meziane DEV**

Ici, c’est la table des articles qui représentée, avec les différentes colonnes nécessaires pour l’ajout d’un article.

Par exemple, la table **POSTS** contient un ***Identifiant unique***, qui auto-incrémenté, un **title** (titre de l’article), un **content** ( le contenu de l’article), **writer** (l’auteur de l’article), **image** (l’image pour l’article) **date** (date de poster de l’article).

Enfin, la dernière colonne est un booléenne pour indiquer si l’article est posté ou non.

Une fois les tables établies, il reste une fonctionnalité importante concernant la base de données en MySQL : les **liaisons**

En effet, les liaisons ont pour but d’éviter les redondances, d’optimiser le travail des bases de données et d’en faciliter la structure et la compréhension.

Le but de la liaison est de relier un article avec son auteur et sa date de publication, avec sont émail.

Pour ce faire, j’ai fait une requête **SQL** suivant :

  SELECT       posts.id,

                posts.title,

                posts.image,

                posts.date,

                posts.content,

                admins.name

     FROM posts

    JOIN admins

    ON posts.writer=admins.email

    WHERE posted='1'

    ORDER BY date DESC

    LIMIT 0,2

L’ordre du requête est vraiment très important perce que si par exemple je choisis limite 02, il regardera que dans les deux premiers éléments de la table et puis il verra dans les deux premiers éléments la ou **posted = 1** alors ici d’abord il regarde ses **posted =1** donc il regarde pour tous et puis il les ordonne, il les mets dans l’ordre décroissant et finalement il ne choisit que les deux derniers articles.

Ensuite pour récupérer un seul l'article et établir le lien avec la table de **posts** et la table **comments,** j’ai fait une requête **SQL** suivant :

SELECT   posts.id,

                    posts.title,

                    posts.image,

                    posts.content,

                    posts.date,

                    admins.name

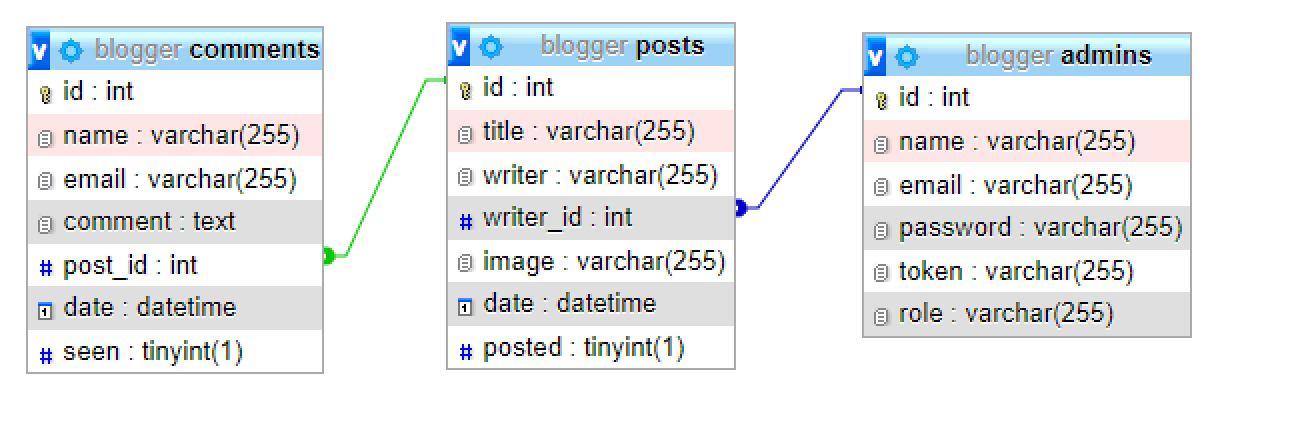
            FROM posts

            JOIN admins

            ON posts.writer = admins.email

            WHERE posts.id='{$\_GET['id']}'

            AND posts.posted = '1'

C’est donc dans ce procedé-ci que la base de données a été réalisée, et que les liaisons ont été établies. En voici un schéma résumant la base de données ainsi que les liaisons entre les tables.

**Image N°3 : schéma des Table de BDD**

### 3.2.2. Compétence N° 06Développer les composants d’accès aux données (blog)

Après avoir créé la base de données on accède à celle-ci via des requêtes, pour pouvoir utiliser les données ou en ajouter. Pour commencer nous devons établir une connexion.

|  |
| --- |
| session\_start();  *//définir les variables de la base de donnée*      $dbhost = 'localhost';      $dbname = 'blog';      $dbuser = 'root';      $dbpswd = '';  *//connection à la base de donnée*   try{$db=new PDO('mysql:host='.$dbhost.';dbname='.$dbname,$dbuser,$dbpswd,array(PDO::MYSQL\_ATTR\_INIT\_COMMAND => 'SET NAMES utf8', PDO::ATTR\_ERRMODE => PDO::ERRMODE\_WARNING));      }catch(PDOexception $e){          die("Une erreur est survenue lors de la connexion à la base de données");      } |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

La connexion est définie par 4 paramètres :

1. L’adresse de la base de données.
2. Le nom de la base de données correspondant.
3. Le login pour se connecter a phpmyadmin.
4. Le MDP s'il y en a un.

Il est aussi possible, comme le vous le voyez, de créer une fonction de

connexion pour faciliter la répétition de l’appel à la connexion sur de multiples pages.

Il y a plusieurs requêtes nécessaires pour qu’un utilisateur puisse se connecter au site.

?php

*//  la connection a la base de données*

    include 'functions/main-functions.php';

*// récupérer les pages par la fonction "scandir qui permet de scanner les pages*

    $pages = **scandir**('pages/');

    if(**isset**($\_GET['page']) && !**empty**($\_GET['page'])){

        if(**in\_array**($\_GET['page'].'.php',$pages)){

            $page = $\_GET['page'];

        }else{

            $page = "error";

        }

    }else{

        $page = "home";

    }

Ici par exemple, en commence par inclure la fonction main pour connecté a la BDD

Dans mon application je ne fait pas un fichier par page, mais tout les pages passe par index et puis index va les rechercher les bons fichiers pour composer la page .

J’ai crée un dossier qui s’apple pages, contient toutes les pages, pour récupérer j’ai créé une variable $pages = **scandir**('pages/'); qui permet de scanner le contenu d’un dossier, pour ca premier temps vérifier que cette page a bien été écrite, je vais créer une condition si jamais il y a bien cette page if(**isset**($\_GET['page']) et cette page n’est pas vide !**empty**($\_GET['page']), dans ce cas la je vais commencer a travailler avec cette page je vais voir si elle se trouve bien dans le dossier pages avec     if(**isset**($\_GET['page']) && !**empty**($\_GET['page'])){

        if(**in\_array**($\_GET['page'].'.php',$pages)){

            $page = $\_GET['page'];}

Par contre c’est l’utilisateur a saisie une page mais cette page n’existe pas dans ce cas la page retourner soit la page d’erreur

else{

            $page = "error";

        }

Si la page est vide nous renvoyait a la page home

}else{

        $page = "home";

    }

### 3.2.3. Compétence N° 07  Développer la partie Back-End d’une application web ou web mobile (blog)



**Image N°1 : interface de connexion d’amine au modé**

Pour ce projet, pour accéder au panneau d’administration en tapons le **URL** **admin**.

**http://localhost/GitHub/meziane-TIC-BLOG/admin/index.php?page=login**

Nous n’avons pas de bouton sur le site car seuls les administrateurs peuvent y accéder dans la page de connexion je mets mon adresse email et mots de passe de l’administrateur et je suis directement connecté.

Dans le tableau de bord il ya trois petite aperçu **publication** **commentaires** et **administrateur** j’ai juste en dessous les commentaires non lu donc ici je peux valider le commentaire ou supprimé alors il disparait de la liste des commentaires non lus



**Image N°2 : Tableau de Bord**

Dans cette page (Image2), l’administrateur pourra alors consulter les différentes articles donc ici je peux écrire des articles soit je valide pour publier au bien non ici il apparaitre avec un petit cadenas signifie qu’il n’est pas publier et qui n’est pas rendu public perce que n’est pas terminée, ainsi les commentaires pour validé au supprimé.

Il pourra également dans partie paramètres qui permet de gérer les modérateurs dans le formulaire **Nouveau modérateur**, je peux ajouter un autre modo **(Image N°1)**, j’ai ’intégré MAILDEV, je reçois un émail en local avec un identifier et un TOKEN après je connecte dans l’espace administrateur et la on me demande de choisir un mots de passe, quand un modo est connecté il peut uniquement que valider soit supprimé soit visionner **les commentaires.**

Pour ajouter un modérateur, elle s’effectuera dans le la page « settings.php »

<?php

            if(**isset**($\_POST['submit'])){

                $name = **htmlspecialchars**(**trim**($\_POST['name']));

                $email = **htmlspecialchars**(**trim**($\_POST['email']));

                $email\_again = **htmlspecialchars**(**trim**($\_POST['email\_again']));

                $role = **htmlspecialchars**(**trim**($\_POST['role']));

                $token = **token**(30);

                $errors = [];

                if(**empty**($name) || **empty**($email) || **empty**($email\_again)){

                    $errors['empty'] = "Veuillez remplier tous les champs";

                }

*//correspondance de l'adresse email*

                if($email != $email\_again){

                    $errors['different'] = "Les adresses email ne correspondent pas";

                }

                if(**email\_taken**($email)){

                    $errors['taken'] = "L'adresse email est déjà assignée à un modérateur";

                }

                if(!**empty**($errors)){

                    ?>

                        <div *class*="card red">

                            <div *class*="card-content white-text">

                                <?php

                                foreach($errors as $error){

**echo** $error."<br/>";

                                }

                                ?>

                            </div>

                        </div>

                    <?php

*//ajoute le modo ds la bdd*

                }else{

**add\_modo**($name,$email,$role,$token);

                }

            }

        ?>

Le modérateur et sont mots de passe sera haché grâce à la fonction **« password\_hach »,** et vérifie grâce a la fonction **« password\_verify »** qui vérifie si le hachage du mots de passe inscrit dans le champ. Cela permet de renforcer la sécurité.

function **update\_password**($password){

    global $db;

    $p = [

        'password'  =>  **password\_hash**($password,PASSWORD\_DEFAULT),

        'session'   =>  $\_SESSION['admin']

    ];

    $sql = "UPDATE admins SET password = :password WHERE email=:session";

    $req = $db->**prepare**($sql);

    $req->**execute**($p);

}

Concernant la connexion, elle s’effectuera dans la page « login.php »

   <?php

*// on récupére les données des inputs*

                if(**isset**($\_POST['submit'])){

                    $email = **htmlspecialchars**(**trim**($\_POST['email']));

                    $password = **htmlspecialchars**(**trim**($\_POST['password']));

                    $errors = [];

                    if(**empty**($email) || **empty**($password)){

                        $errors['empty'] = "Tous les champs n'ont pas été remplis!";

                    }else if (**is\_admin**($email,$password) == 0){

                        $errors['exist']  = "Cet administrateur n'existe pas";}

                    if(!**empty**($errors)){

                        ?>

                        <div *class*="card red">

                            <div *class*="card-content white-text">

                                <?php

                                    foreach($errors as $error){

**echo** $error."<br/>";

                                    }

                                ?>

                            </div>

                        </div>

                        <?php

                    }else{

                        $\_SESSION['admin'] = $email;

**header**("Location: index.php?page=dashboard");

                    }

                }

            ?>

Ici, si le formulaire de connexion est validé, on stock les champs du pseudo et du mot de passe de l’administrateur, puis on enclencher la méthode de l’objet **« is\_admin »** afin de retrouver l’administrateur dans le champ du pseudo.

L’administrateur et sont mots de passe sera vérifier grâce à la fonction « password\_verify » , qui vérifie si le hachage du mots de passe inscrit dans le champ. Cela permet de renforcer la sécurité.

S’il y’a correspondance, alors le rôle de l’administrateur sera vérifier, pour savoir c’est bien un administrateur au modérateur.

Maintenant que l’administrateur est connecté il pourra alors gérer la plupart des données de son application.

function **is\_admin**($email, $password): bool {

*// connection a la bdd*

    global $db;

*//créer requête de recupération pour la connection soit admin au modo*

    $req = $db->**prepare**("SELECT \* FROM admins WHERE `email` = ? AND `role` = 'admin' || `role` = 'modo' ");

    $req->**execute**([$email]);

    $result = $req->**fetch**();

    if ($result) {

        return **password\_verify**($password, $result['password']);

    } else {

        return false;

    }

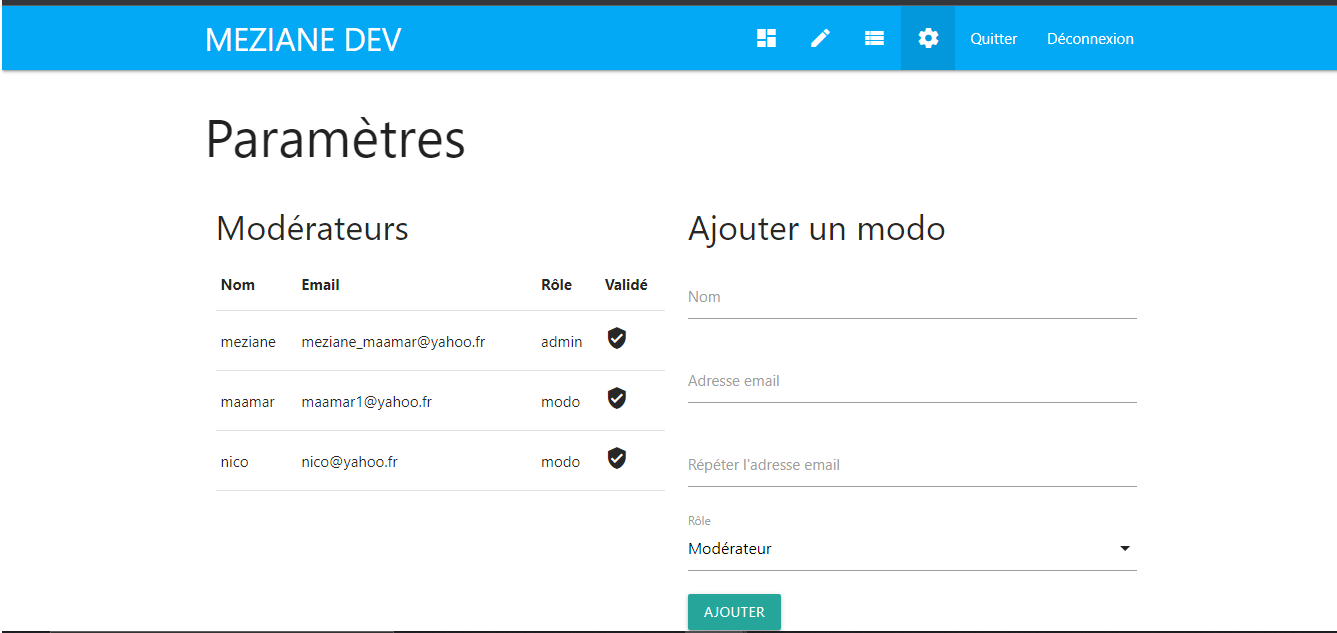
}

# 4. Jeu d’essai

Pour le jeu d’essai j’ai choisi le formulaire d’ajoute un administrateur ou modérateur de mon projet personnel (Blog). Je vais tester trois choses :

* La vérification des champs vides
* Le format du champ de type email
* La concordance entre les deux champs mot de passe

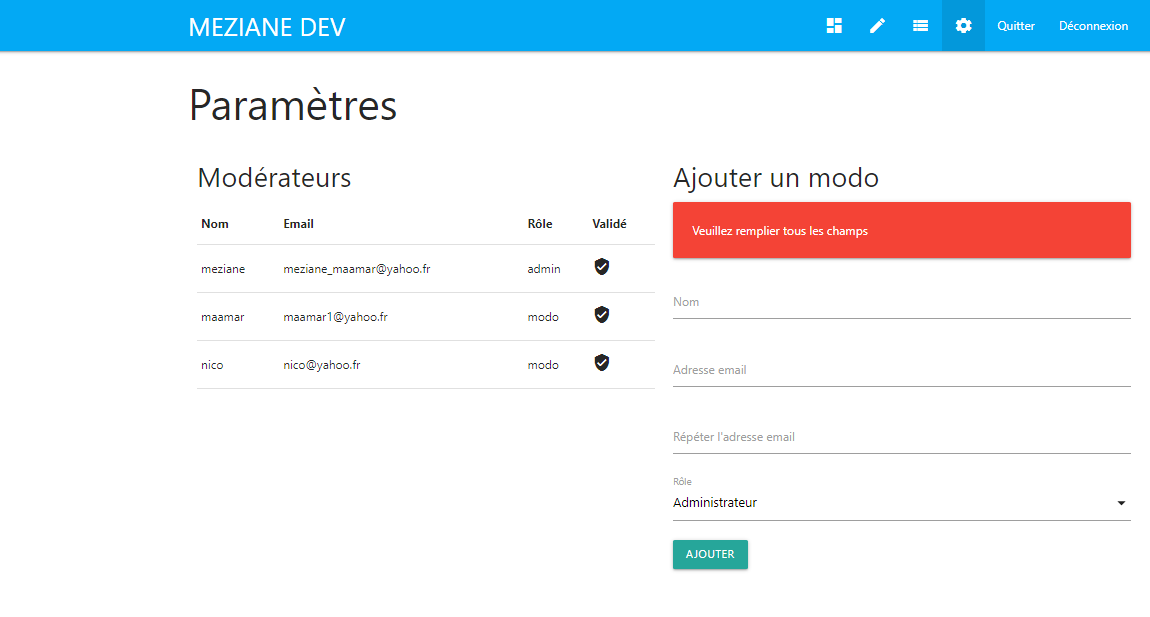
Voici le formulaire d’ajoute un administrateur ou modérateur



**Image N°1 formulaire d’ajoute un administrateur ou modérateur**

## 4.1 La vérification des champs vides

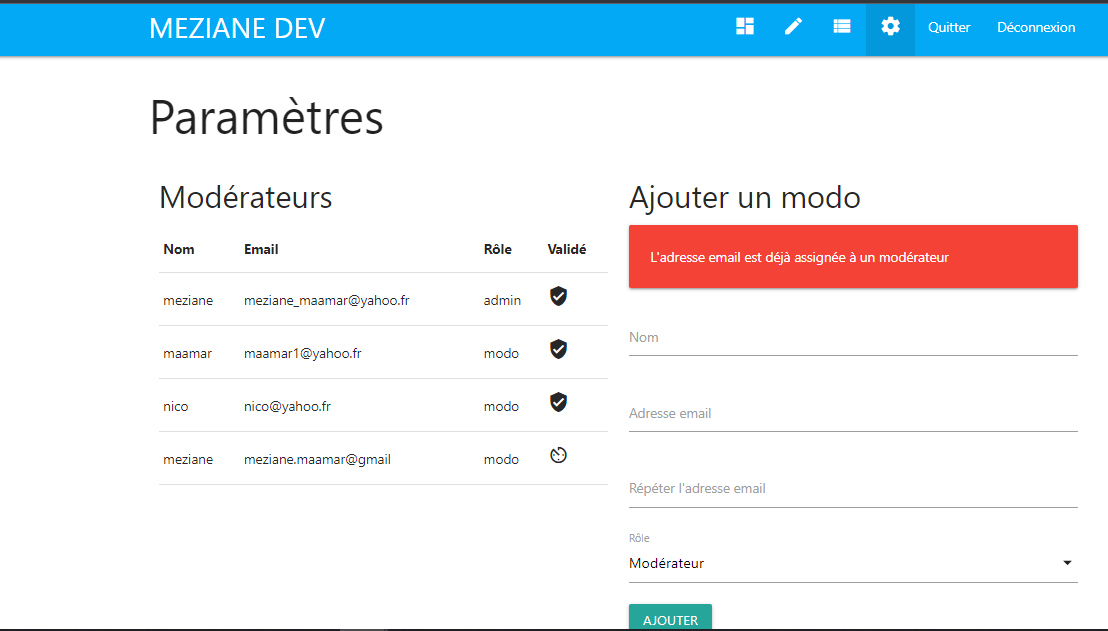
Le résultat du premier test, à savoir les champs vide. Ici le script PHP vérifie que chaque champ est rempli, si ce n’est pas le cas un message d’erreur apparait dans le formulaire :



**Image N°2 La vérification des champs vides**

## 4.2 Le format du champ de type email

La vérification du champ de type émail, dans un premier temps j’ai utilisé un « input » de type email coté HTML. C’est il ya un modé qui inscrit ici le script PHP vérifie l’émail de la BDD. Voici l’exemple de message d’erreur qui apparait lorsque l’administrateur entre une adresse email existe dans la BDD.

~~~~

**Image N°3 Le format du champ de type email**

## 4.3 La concordance entre les deux champs mot de passe

Pour le test de correspondance, le script PHP vérifie si les mots de passe sont égaux. Si ce n’est pas le cas, un message d’erreur apparait dans le formulaire :



**Image N°4 La concordance entre les deux champs mot de passe**

# 5. Veille technologique

## 5.1. Contexte

Que ce soit dans le projet de groupe ou dans le projet perso, on a eu des formulaires à traiter, et un input sur une interface web représente une opportunité d’injection SQL ou JS. J’ai donc effectué quelques recherches sur le sujet et ai trouvé quatre fonctions utiles pour se protéger dans la plupart des cas :

* La fonction PHP « htmlspecialchars » qui converti certains caractères spéciaux en entités HTML
* La fonction PHP « trim » qui suprime les espaces en dédut et fin de chaine de caractères spéciaux en entités HTML

### 5.1.1. Extrait du site anglophone

Pour le texte anglophone j’ai choisi une description de la fonction PHP « htmlspecialchars ».

Certain characters have special significance in HTML, and should be represented by HTML entities if they are to preserve their meanings. This function returns a string with these conversions made. If you require all input substrings that have associated named entities to be translated, use [htmlentities()](https://www.php.net/manual/en/function.htmlentities.php) instead.

If the input string passed to this function and the final document share the same character set, this function is sufficient to prepare input for inclusion in most contexts of an HTML document. If, however, the input can represent characters that are not coded in the final document character set and you wish to retain those characters (as numeric or named entities), both this function and [htmlentities()](https://www.php.net/manual/en/function.htmlentities.php) (which only encodes substrings that have named entity equivalents) may be insufficient. You may have to use [mb\_encode\_numericentity()](https://www.php.net/manual/en/function.mb-encode-numericentity.php) instead.

### 5.1.2. Traduction en Français

Certains caractères ont des significations spéciales en HTML, et doivent être remplacés par des entités HTML pour conserver leurs significations. Cette fonction retourne une chaîne de caractères avec ces modifications. Si vous avez besoin que toutes les sous-chaînes en entrée qui sont associées à des entités nommées soient transformées, utilisez la fonction [htmlentities()](https://www.php.net/manual/fr/function.htmlentities.php).

Si la chaîne en entrée passée à cette fonction et le document final partagent le même jeu de caractères, cette fonction est suffisante pour préparer l'entrée pour une inclusion dans la plupart des contextes d'un document HTML. Si cependant, l'entrée peut présenter des caractères qui ne sont pas codés dans le jeu de caractères du document final, et que vous souhaitez épargner ces caractères (comme des numériques ou des entités nommés), cette fonction et la fonction [htmlentities()](https://www.php.net/manual/fr/function.htmlentities.php) (qui n'encodes que les sous-chaînes qui ont des entités nommés équivalentes) ne sont pas suffisantes. Vous devez utiliser la fonction [mb\_encode\_numericentity()](https://www.php.net/manual/fr/function.mb-encode-numericentity.php) à la place.

# Conclusion

Cette formation m’a permis d’évoluer dans le domaine du développement ou j’ai pu développer certaines affinités.

Comme vous avez pu le constater, ma manière de travailler et de coder a évolué au fur et à mesure des difficultés rencontrées et des connaissances assimilées.

De plus, je me rendais compte, que plus j’assimilais de connaissances, plus il m’en restait à découvrir.

Espérant avoir pu partager mes retours d’expériences de la manière la plus claire possible. Je suis convaincu que grâce à cette formation, aura marqué le début d’une longue aventure dans le web, voire dans d’autres domaines

1. Un fichier **robots.txt** indique aux robots d'exploration d'un moteur de recherche les URL auxquelles il peut accéder sur votre site [↑](#footnote-ref-1)
2. **Sitemaps** est un [protocole](https://fr.wikipedia.org/wiki/Protocole_de_communication) permettant à un [webmestre](https://fr.wikipedia.org/wiki/Webmestre) d'informer les [moteurs de recherche](https://fr.wikipedia.org/wiki/Moteur_de_recherche) des [adresses](https://fr.wikipedia.org/wiki/Uniform_Resource_Locator) d'un [site web](https://fr.wikipedia.org/wiki/Site_web) disponibles pour l'indexation automatique. Proposé initialement par [Google](https://fr.wikipedia.org/wiki/Google). [↑](#footnote-ref-2)
3. Les fichiers **.htaccess** (pour [hypertext](https://fr.wikipedia.org/wiki/Hypertexte" \o "Hypertexte) access) sont des [fichiers de configuration](https://fr.wikipedia.org/wiki/Fichier_de_configuration) des [serveurs HTTP](https://fr.wikipedia.org/wiki/Serveur_web) [Apache](https://fr.wikipedia.org/wiki/Apache_HTTP_Server). Leur particularité est leur emplacement : dans les répertoires de données du [site web](https://fr.wikipedia.org/wiki/Site_web), au lieu du répertoire de configuration d'Apache. [↑](#footnote-ref-3)