



DOSSIER PROFESSIONNEL (DP)

Nom de naissance

- Betterki

Nom d'usage

- Betterki

Prénom

- Adam

Adresse

- 26 Rue de la Pochette, Bobigny 93000

Titre professionnel visé

Développeur Web et Web mobile

MODALITÉ D'ACCÈS :

- Parcours de formation
- Validation des Acquis de l'Expérience (VAE)

DOSSIER PROFESSIONNEL (DP)

Présentation du dossier

Le dossier professionnel (DP) constitue un élément du système de validation du titre professionnel.
Ce titre est délivré par le Ministère chargé de l'emploi.

Le DP appartient au candidat. Il le conserve, l'actualise durant son parcours et le présente **obligatoirement à chaque session d'examen.**

Pour rédiger le DP, le candidat peut être aidé par un formateur ou par un accompagnateur VAE. Il est consulté par le jury au moment de la session d'examen.

Pour prendre sa décision, le jury dispose :

1. des résultats de la mise en situation professionnelle complétés, éventuellement, du questionnaire professionnel ou de l'entretien professionnel ou de l'entretien technique ou du questionnement à partir de productions.
2. du **Dossier Professionnel (DP)** dans lequel le candidat a consigné les preuves de sa pratique professionnelle.
3. des résultats des évaluations passées en cours de formation lorsque le candidat évalué est issu d'un parcours de formation
4. de l'entretien final (dans le cadre de la session titre).

[Arrêté du 22 décembre 2015, relatif aux conditions de délivrance des titres professionnels du ministère chargé de l'Emploi]

Ce dossier comporte :

- pour chaque activité-type du titre visé, un à trois exemples de pratique professionnelle ;
- un tableau à renseigner si le candidat souhaite porter à la connaissance du jury la détention d'un titre, d'un diplôme, d'un certificat de qualification professionnelle (CQP) ou des attestations de formation ;
- une déclaration sur l'honneur à compléter et à signer ;
- des documents illustrant la pratique professionnelle du candidat (facultatif)
- des annexes, si nécessaire.

DOSSIER PROFESSIONNEL (DP)

Pour compléter ce dossier, le candidat dispose d'un site web en accès libre sur le site.

 <http://travail-emploi.gouv.fr/titres-professionnels>

Sommaire

Exemples de pratique professionnelle

Activité n° 1 : Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité.

- Compétence n° 1 : Maquetter une application	p. p.	6
- Compétence n° 2 : Réaliser une interface utilisateur web statique et adaptable	p. p.	10
- Compétence n° 3 : Développer une interface utilisateur web dynamique	p. p.	15
- Compétence n° 4 : Réaliser une interface utilisateur avec une solution de gestion de contenu ou e-commerce	p. p.	20

Activité n° 2 : Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité.

- Compétence n° 5 : Créer une base de données	p. p.	25
- Compétence n° 6 : Développer les composants d'accès aux données	p. p.	36
- Compétence n° 7 : Développer la partie back-end d'une application web ou web mobile	p. p.	41
- Compétence n° 8 : Elaborer et mettre en œuvre des composants dans une application de gestion de contenu ou e-commerce	p. p.	54

Intitulé de l'activité-type n° 3

- Intitulé de l'exemple n° 1	p. p.
- Intitulé de l'exemple n° 2	p. p.
- Intitulé de l'exemple n° 3	p. p.

DOSSIER PROFESSIONNEL^(DP)

Intitulé de l'activité-type n° 4

- Intitulé de l'exemple n° 1
- Intitulé de l'exemple n° 2
- Intitulé de l'exemple n° 3

p.

p. p.

p. p.

p. p.

Titres, diplômes, CQP, attestations de formation (*facultatif*)

p.

Déclaration sur l'honneur

p.

66

Documents illustrant la pratique professionnelle (*facultatif*)

p.

67

Annexes (*Si le RC le prévoit*)

p.

68

DOSSIER PROFESSIONNEL^(DP)

EXEMPLES DE PRATIQUE

PROFESSIONNELLE

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité

Compétence n°1 - Maquetter une application

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

J'ai commencé par créer une **planche de tendance (Moodboard)**, avec le logiciel **Canva**, où l'on retrouve le logo, les couleurs, le slogan et quelques illustrations qui seront intégrées au projet.

Ce qui permettra aux clients(es) d'avoir un rendu visuel à la hauteur de ses attentes sur les actifs du projet.

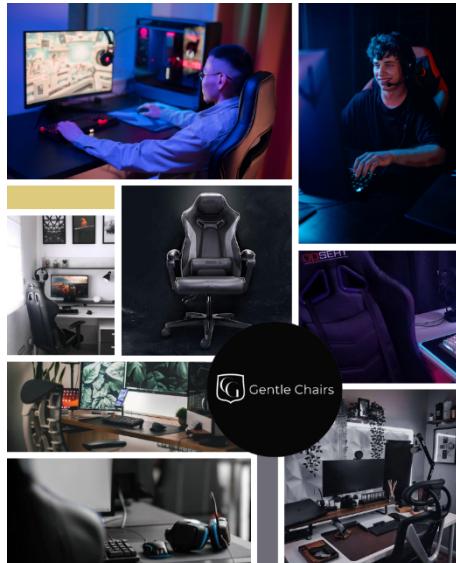


Figure 1 : le MoodBoard

Ensuite j'ai créé une **charte graphique** avec le logiciel **Figma** ce qui consiste à détailler tous les points à respecter qui regroupe l'ensemble des codes définissant l'identité visuelle du projet.

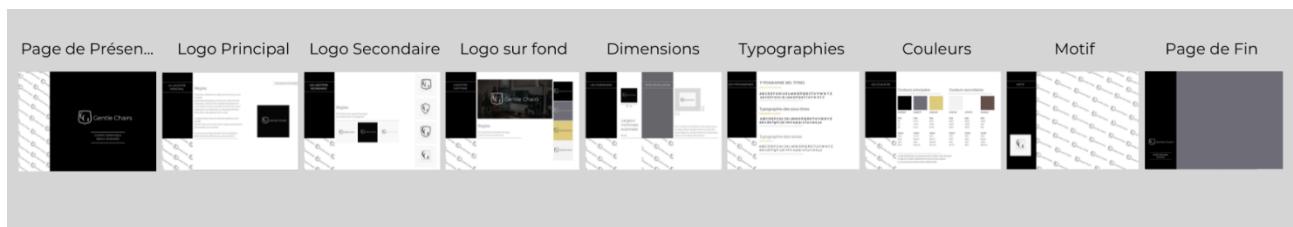


Figure 2 : La charte graphique

L'objectif du projet est de créer un site e-commerce de chaises gaming & bureautique haut de gamme avec l'utilisation du SEO comme canal de vente principal, la boutique a pour but de faciliter les heures passées devant l'écran à jouer ou à travailler, en leurs proposant différentes 2 grandes catégories :

DOSSIER PROFESSIONNEL (DP)

- Mettre en place une collection de chaises haut de gamme propre à la bureautique, peut être vendu en gros en B2B ou aux particuliers.
- Mettre en place une seconde collection de chaises haut de gamme propre à la pratique des jeux vidéo, peut être vendu sur plusieurs marchés à des particuliers.

A la suite j'ai présenté le logo principal du projet et ses bonnes pratiques ainsi que les logos secondaires simple puis sur fond avec leurs dimensions, leurs couleurs, et puis j'ai indiqué la typographie rencontrée au fil du projet, la palette de couleurs avec ses codes du type : RGB, HEXADECIMAL et CMJN et j'ai terminé par le motif, et enfin par une page de fin basique.

Et puis j'ai créé la **maquette** de la page d'accueil du projet en me servant du logiciel **Figma** pour la mise en page, et de **Remove.bg** pour supprimer les arrières plans des images et des icônes pour mieux les intégrer, tout en respectant les codes définis dans la planche de tendance et la charte graphique.

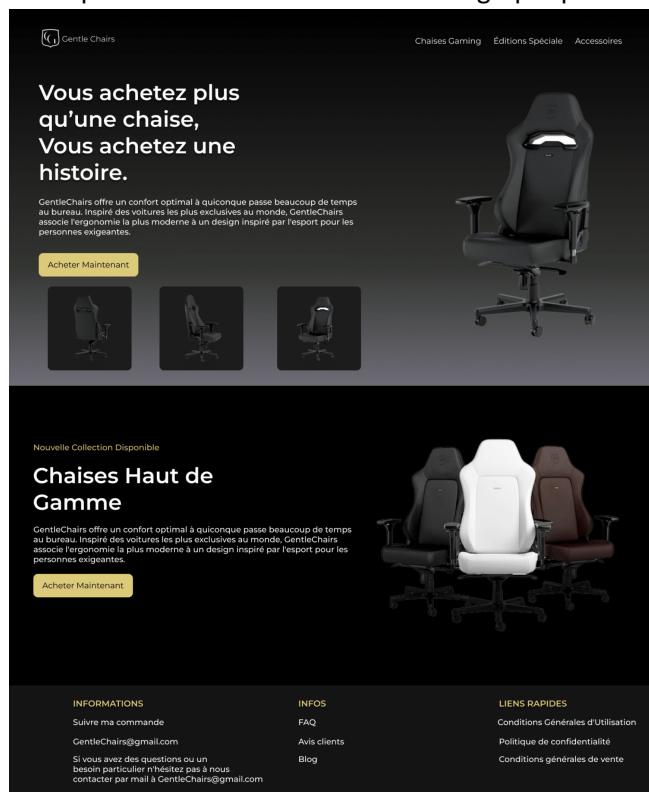


Figure 3 : Maquette de la page d'accueil de mon projet GentleChairs

Pour finir avec cette compétence j'ai commencé à coder mon projet en **HTML 5 (HyperText Markup Language)** et en **CSS 3 (Cascading Style Sheets)** avec l'éditeur de code **Visual Studio Code** et pour sauvegarder l'avancée de mon travail en créant des dépôts j'ai utilisé **Github Desktop**.

DOSSIER PROFESSIONNEL (DP)

```
[DOCTYPE html>
html lang="fr">
head>
    
    
    
    <title>GentleChairs</title>
    <link rel="stylesheet" href="GentleChairs.css">
    <link rel="icon" type="image/png" href="logodiz.png">

```

```
</head>
<body>
    <header>
        <nav>
            <div>GentleChairs</div>
            <div class="right">
                <p>Chaises Gaming</p>
                <p>Editions Spéciale</p>
                <p>Accessoires</p>
            </div>
        </nav>
        <section class="présentation first">
            <div class="text">
                <p>GentleChairs offre un confort optimal à quiconque passe beaucoup de temps au bureau. Inspire des voitures les plus exclusives au monde, GentleChairs associe l'ergonomie la plus moderne à un design inspiré par l'esport pour les personnes exigeantes.</p>
                <button>Acheter Maintenant</button>
            </div>
            <div class="small-images">
                <div></div>
                <div></div>
                <div></div>
            </div>
            <div class="image-présentation">
                
            </div>
        </section>
    </header>
```

Figure 4 : Code HTML

```
@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@300;400&display=swap');
:root {
    --gold: #DDCA7C;
}

body {
    margin: 0;
    font-family: 'Montserrat', sans-serif;
    color: white;
}

header {
    background: linear-gradient( #000000, #181717, #333531, #6E6D77);
}

header nav {
    padding: 30px 5% 0 5%;
    display: flex;
    flex-wrap: wrap;
    justify-content: space-between;
    align-items: center;
}

header nav .left{
    margin: 0;
    text-transform: uppercase;
    color: var(--gold);
}

header nav .right {
    display: flex;
    align-items: center;
}

header nav .right p {
    margin: 0 15px 0 0;
    cursor: pointer;
}
```

Figure 5 : Code CSS

DOSSIER PROFESSIONNEL (DP)

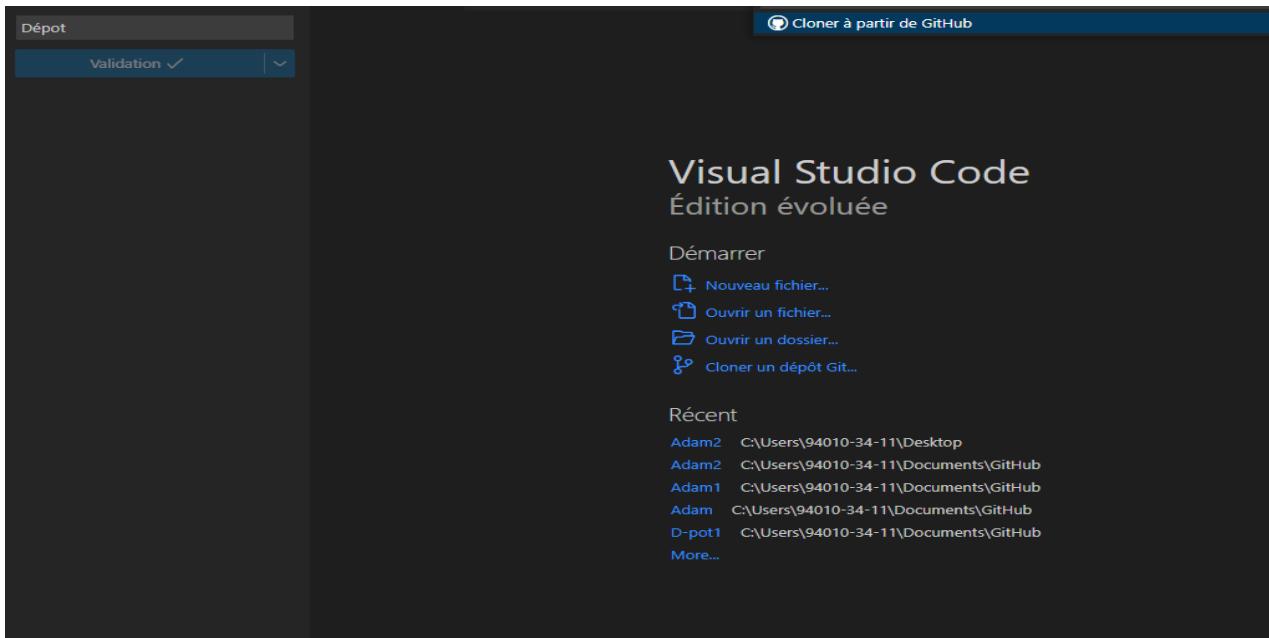


Figure 6 : Dépôt de Visual Studio Code vers Github Desktop

2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisé pour cette première compétence :

Logiciel: Visual studio code.

Site web: Github desktop, Canva, Remove.bg

Application: Figma.

Langage: Html5, Css3.

3. Avec qui avez-vous travaillé ?

Pour les compétences de 1 à 8 j'ai travaillé seul en me documentant sur internet

4. Contexte

Nom de l'entreprise, organisme ou association ➤ Formation AFPA

Chantier, atelier, service ➤ Formation DWWM

Période d'exercice ➤ Du : 05/09/2022 au : 26/04/2022

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité

Compétence n°2 - *Réalisation d'une interface utilisateur web statique et adaptable*

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Pour cette deuxième compétence j'ai commencé par maquetter mon projet en format **mobile, tablette et desktop** toujours avec le logiciel **Figma**.

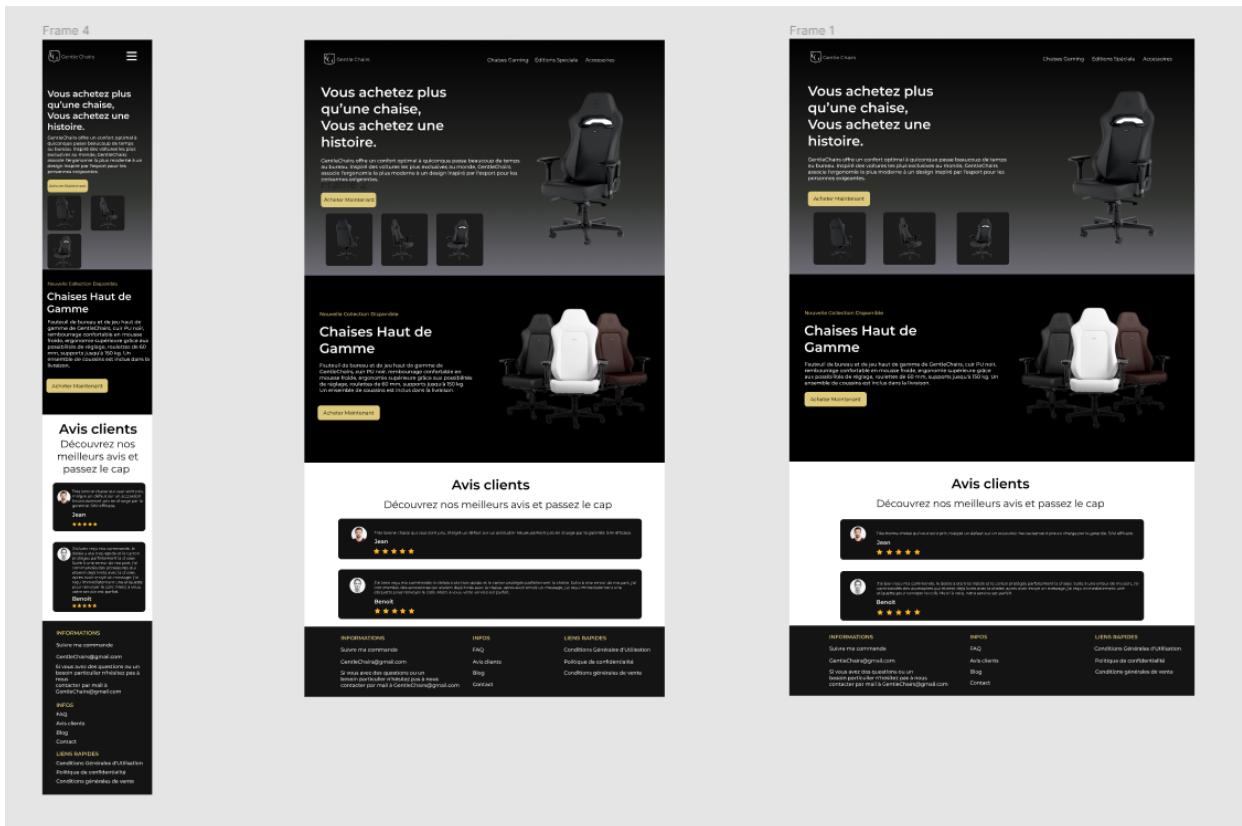


Figure 7 : La maquette au format Mobile, Tablette et Desktop

DOSSIER PROFESSIONNEL (DP)

Et par la suite avec l'éditeur de code **Visual Studio Code**, j'ai continué le codage de la page d'accueil de mon projet, en y ajoutant les **média queries** pour le rendre adaptable sur tous les types de supports : Mobile, Tablette et Desktop.

```
<style>
/*MEDIA QUERIES*/
@media screen and (min-width: 320px) and (max-width: 374px) {
    /* API BOITE METEO PANORAMIQUE */
    .boitemeteo3,.boitemeteo6,.boitemeteo9 {
        border: 1px solid #f1f1f1;
        background: linear-gradient(45deg, #1069b659, #36a33262);
        border-radius: 25px 25px 225px 25px/25px 225px 25px 255px;
        box-shadow: 0 5rem 1rem rgba(0,0,0,.15) !important;
        display: grid;
        grid-template: 70px 30px 30px / repeat(7, 37px);
        height: 72px;
        margin-top: 765px;
        margin-bottom: 800px;
        margin-left: -75px;
        opacity: 0;
        position: relative;
        width: 264px;
    }

    .bloc-logo-info.pano {
        align-items: center;
        display: flex;
        grid-area: 1 / 1 / 2 / -1;
        justify-content: center;
    }
}

@media screen and (min-width: 768px) and (max-width: 991px) {
    h1 {
        margin-top: 350px;
        margin-bottom: 80px;
        font-size: 28px;
    }

    h2 {
        margin-top: 24px;
        font-size: 24px;
    }

    .col-12 {
        margin-bottom: 25px;
    }

    .pano {
        margin-bottom: 25px;
    }

    .espace {
        height: 5px;
        margin-bottom: 5px;
    }
}

@media screen and (min-width: 1024px) and (max-width: 1439px) {
    .container {
        max-width: 1010px;
    }

    h1 {
        margin-top: 400px;
        margin-bottom: 80px;
        font-size: 32px;
    }

    h2 {
        display: flex;
        position: relative;
        justify-content: center;
        margin-top: 24px;
        font-size: 25px;
    }

    .col-12 {
        margin-bottom: 22px;
    }
}
```

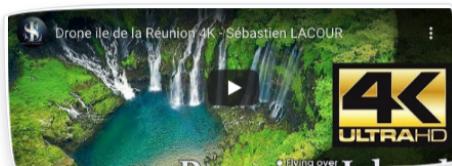
Figure 8 : Les média quéries

J'ai intégré du contenu multimédia, exemple une Vidéo provenant de "**youtube**" pour accentuer la visibilité de mon site par les utilisateurs.(pas encore fait)



" Le DoDo Y Guide A Ou ! "

" Le DoDo Vous Guide ! "



Avec DODO LE GUIDE Vous Serez Votre Propre Guide.



Figure 9 : Une vidéo youtube

J'ai aussi intégré un **menu hamburger** à mon bandeau de navigation (*nav-bar*), pour faciliter l'accès aux informations disponibles sur le site.

DOSSIER PROFESSIONNEL (DP)

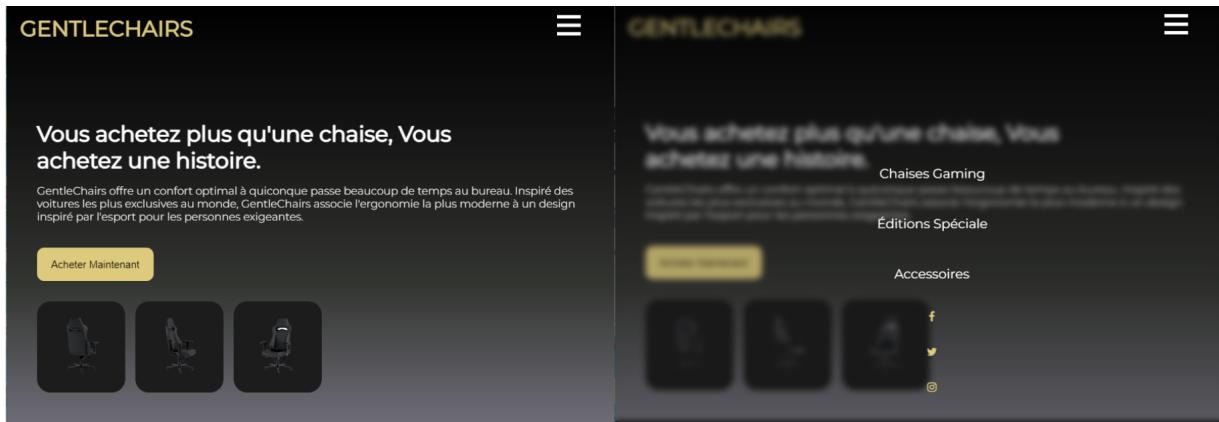


Figure 10 : Menu hamburger

J'ai également intégré des icônes cliquables dans mon bandeau de pied de page (*footer*), qui redirigent vers les réseaux sociaux du type : Twitter, Instagram et Facebook pour offrir une plus grande visibilité à ce projet sur le web.



Figure 11 : Icône réseaux sociaux

Puis j'ai créé des **personas** avec le logiciel **Figma** correspondant aux profils d'utilisateurs qui pourraient se rendre sur le site.

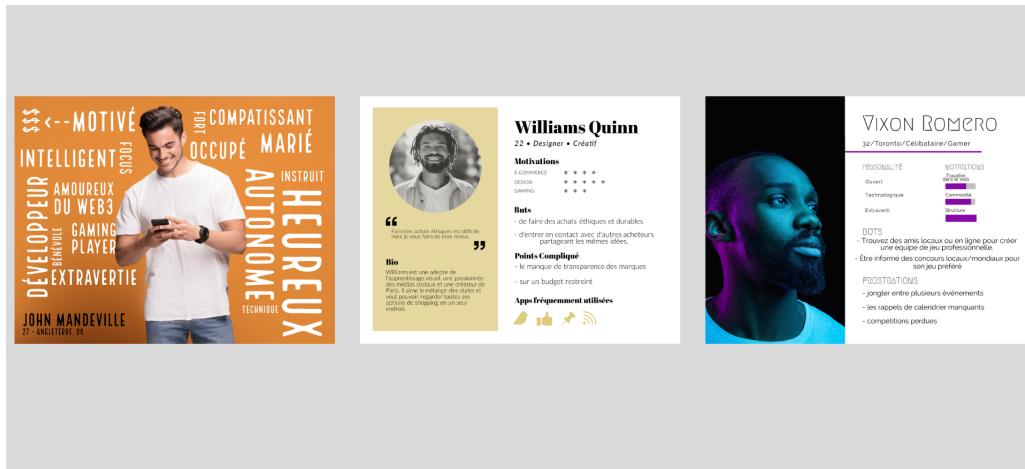


Figure 12 : Les personas

J'ai poursuivi le codage du site en intégrant du code **SCSS**. Avec **SASS** (*Syntactically awesome stylesheets*) on peut imiter la structure du html il faut commencer par intégrer un flux de travail de compilateur Sass à votre projet. Ce qui permet d'obtenir des goodies exemple : conception modulaire, fichiers inclus, mixins, héritage, et de pouvoir mieux se repérer dans le code en ayant moins de lignes en introduisant des variables. (pas encore fait)

DOSSIER PROFESSIONNEL (DP)

The image shows a code editor interface with two files open. The left file is named 'Sass' and contains SCSS code. The right file is named 'CSS' and contains the generated CSS code. Both files have a light gray background with syntax highlighting.

```
SCSS   Sass
$font-stack: Helvetica, sans-serif;
$primary-color: #333;

body {
  font: 100% $font-stack;
  color: $primary-color;
}

CSS
body {
  font: 100% Helvetica, sans-serif;
  color: #333;
}

SCSS   Sass
nav {
  ul {
    margin: 0;
    padding: 0;
    list-style: none;
  }

  li { display: inline-block; }

  a {
    display: block;
    padding: 6px 12px;
    text-decoration: none;
  }
}

CSS
nav ul {
  margin: 0;
  padding: 0;
  list-style: none;
}
nav li {
  display: inline-block;
}
nav a {
  display: block;
  padding: 6px 12px;
  text-decoration: none;
}
```

Figure 13 : Code Scss

2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisé pour cette deuxième compétence:

Logiciel: Visual studio code.

Site web: Github desktop.

Application: Figma.

Langage: Html5, Scss3.

3. Avec qui avez-vous travaillé ?

Pour les Compétences de 1 à 8 j'ai travaillé seul en me documentant sur internet

4. Contexte

Nom de l'entreprise, organisme ou association ➤ Formation AFPA

Chantier, atelier, service ➤ Formation DWWM

Période d'exercice ➤ Du : 05/09/2022 au : 26/04/2022

DOSSIER PROFESSIONNEL ^(DP)

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité

Compétence n°3 - Développer une interface utilisateur web dynamique

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Pour la première partie de cette compétence(3A), toujours à partir de l'éditeur de code **Visual Studio Code** j'ai utilisé **Bootstrap** qui est un framework qui permet de simplifier l'action du développeur web en lui donnant accès à une multitude de codes déjà prêts à être intégrés à son projet pour la navigation et les éléments interactifs de celui-ci.

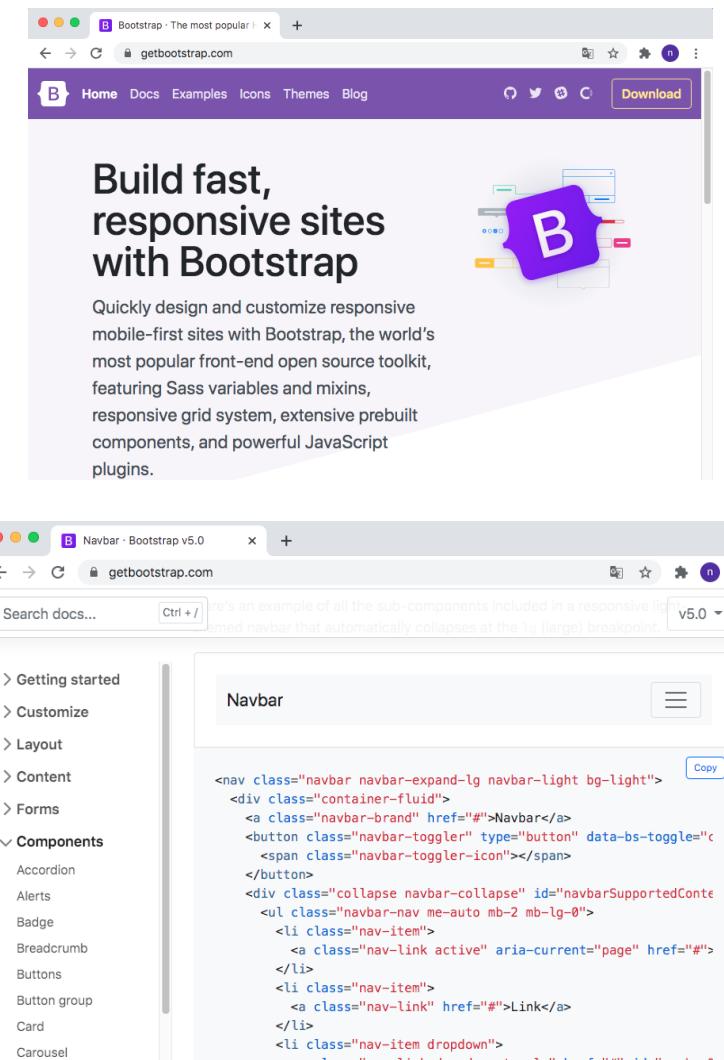
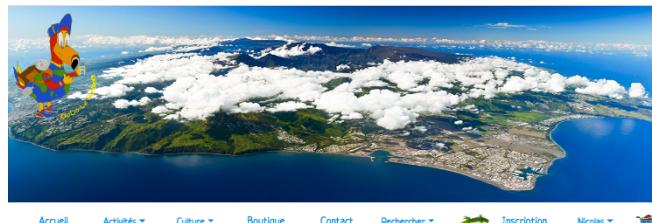


Figure 14 : Page d'accueil de la documentation Bootstrap, et, un exemple de code pour une nav-bar avec menu-hamburger

DOSSIER PROFESSIONNEL (DP)

Cela m'a permis de rendre mon projet dynamique en y intégrant un formulaire de contact interactif à partir de **Bootstrap** ainsi que du **Jquery** qui est une librairie écrite en **Javascript** qui fournit au développeur web de nombreuses fonctions.



The screenshot shows a contact form integrated into a website. At the top, there's a navigation bar with links for Accueil, Activités, Culture, Boutique, Contact, Rechercher, Inscription, and Nicolas. A shopping cart icon is also present. Below the navigation, a cartoon character is visible on the left, and a large image of a tropical island with a city at its base is on the right. The main form area has fields for Nom d'utilisateur, Adresse email, and Message, each with placeholder text. To the right of these fields is a small image of a traditional hut. Below the message field is a file input field labeled "Choisir un fichier" with the message "Aucun fichier choisi". A green "envoyer" button is at the bottom. At the very bottom of the page is a footer banner with social media icons (Facebook, Twitter, etc.) and a blue arrow pointing upwards.

Figure 15 : Un formulaire de contact interactif

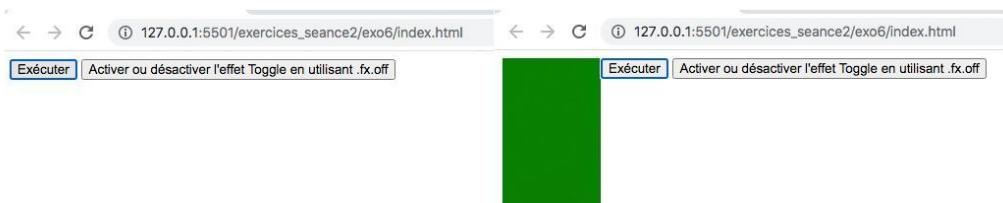


Figure 16 : Le Jquery

Ensuite pour la seconde partie de cette compétence(3B), toujours à partir de l'éditeur de code **Visual Studio Code** j'ai repris mon projet précédent et je l'ai recréé en projet **Argon** dans un environnement **Vue.Js**, également avec du format **.Json** (*JavaScript Object Notation*) qui permet de stocker, lire et partager tous les langages de programmations disponibles.

DOSSIER PROFESSIONNEL (DP)

EXPLORER

OPEN EDITORS

3-DODO LE GUIDE PROJET ARGON_V...

- layout
- starter
- AppFooter.vue
- AppHeader.vue
- plugins
- views
- components
- Accueil.vue

Components.vue

Contact.vue

Landing.vue

Login.vue

Profile.vue

Rando.vue

Register.vue

Starter.vue

App.vue

main.js

registerServiceWorker.js

router.js

starterRouter.js

.gitattributes

.gitignore

babel.config.js

CHANGELOG.md

package-lock.json

package.json

README.md

vue.config.js

yarn.lock

OUTLINE

TIMELINE

Accueil.vue X

```
src > views > Accueil.vue > style > {} @media screen and (min-width: 2056px)
1  <template>
2    <!--<section class="section section-lg section-shaped overflow-hidden my-0">-->
3    <section>
4      <section id="titre">
5        <article id="guide">
6          <h1 id="ancrel1"> LE DODO Y GUIDE A OU </h1>
7        </article>
8        <article id="video">
9          <iframe
10            width="560"
11            height="200"
12            src="https://www.youtube.com/embed/UoTtNd3lu0A?autoplay=1&mute=1&enablejsapi=1&controls=0&loop=1&start=56"
13            frameborder="0"
14            allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture"
15            allowfullscreen
16          ></iframe>
17        </article>
18        <section>
19          <p></p>
20        </section>
21        <section id="présentation">
22          <article id="présentation">
23            <p>
24              Avec <span id="span1">DODO LE GUIDE</span> Vous Serez Votre Propre
25              Guide.
26            </p>
27            <p class="p2">
28              Voici Nos <span>Conseils</span> Et Notre <span>Selections</span> De
29              Lieux Et D'activités Incontournables à Découvrir Lors De Votre
30              Séjour.
31            </p>
32            <div class="d-flex justify-content-center">
33              
34            </div>
35          </article>
36        </section>
37      </section>
38      <section>
39        <p class="espace"></p>
40      </section>
41      <section id="détail">
42        <h3>Le Dodo Y Guide à Ou</h3>
```

Figure 17 : Vue.js

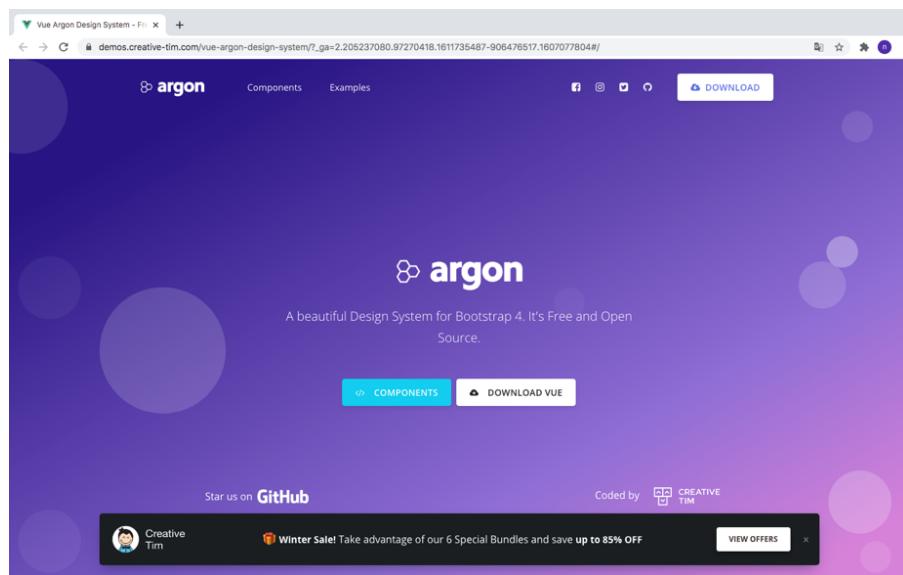


Figure 18 : Page d'accueil Argon

Voici le rendu de ma page d'accueil avec mon projet **Argon Vue.js**.

DOSSIER PROFESSIONNEL (DP)

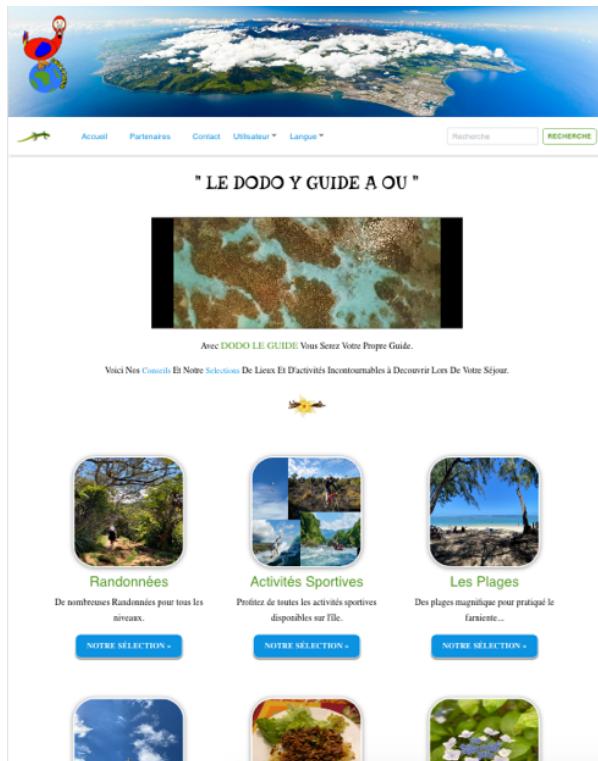


Figure 19 : Visuel de mon projet Vue.js-Argon

2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisé pour cette troisième compétence:

Documentation: Site de Bootstrap et site de creative-tim VueArgon.

Logiciel: Visual studio code.

Site web: Github desktop.

Application: Figma.

Framework: Bootstrap, Argon Vue.js.

Bibliothèque: Jquery.

Langage: Html5, Scss3, Javascript, Json.

3. Avec qui avez-vous travaillé ?

Pour les Compétences de 1 à 8 j'ai travaillé seul en me documentant sur internet

4. Contexte

DOSSIER PROFESSIONNEL (DP)

Nom de l'entreprise, organisme ou association ➤ *Formation AFPA*

Chantier, atelier, service ➤ *Cliquez ici pour taper du texte.*

Période d'exercice ➤ Du : *05/10/2020* au : *11/06/2021*

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité

Compétence n°4 - Réaliser une interface utilisateur avec une solution de gestion de contenu ou e-commerce

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Pour cette quatrième compétence j'ai utilisé **Wordpress** qui se trouve dans la catégorie des **Cms (Content Management Système)**.



Figure 20 : Page d'accueil WordPress

J'ai donc commencé par installer (**Mamp (Mac)**, **Xampp (Universel)**, ou **Wamp(Windows)**) qui me permet de créer une **base de données** en interprétant des scripts, et d'avoir accès au **PhpmyAdmin** pour l'administration Web des bases **MySql** (*système de gestion de base de données relationnelle open source*) et de pouvoir développer et exécuter des sites Web dynamiques, et de mettre en place un serveur Web local, un serveur FTP ou un serveur de messagerie électronique.

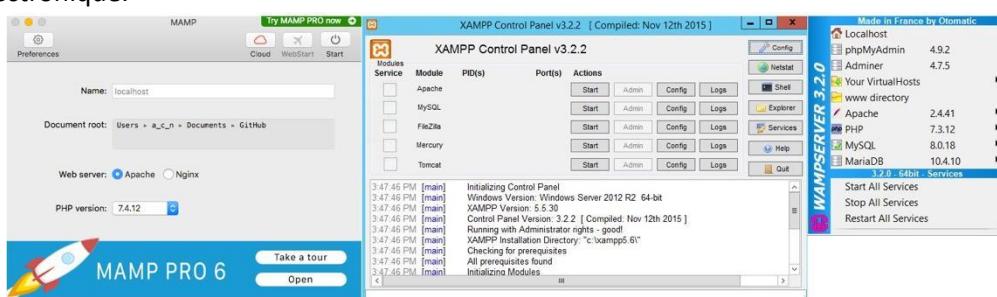


Figure 21 : Mamp, Xampp et Wamp

DOSSIER PROFESSIONNEL (DP)

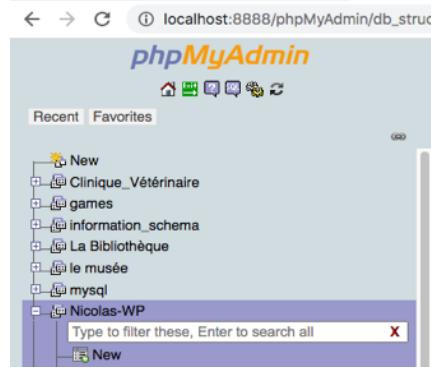


Figure 22 : Création d'une base de données

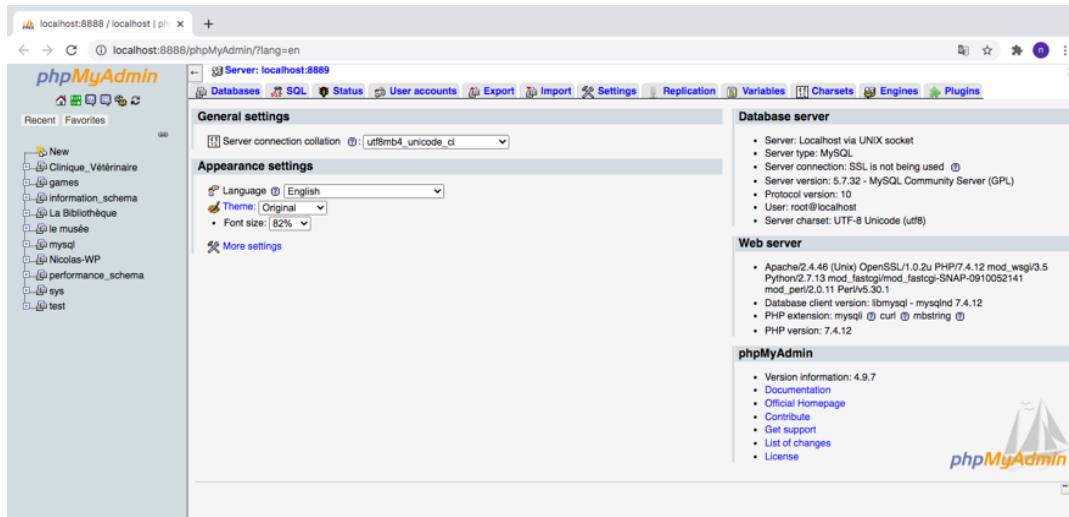


Figure 23 : Page du server local Php My Admin

Après avoir créée une base de données j'ai démarré mon **localhost** sur le **webserver Apache**, ce qui m'a permis d'accéder à Wordpress et tous ses outils.

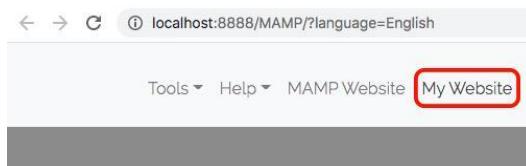


Figure 24 : Lancement de la base de données

DOSSIER PROFESSIONNEL (DP)

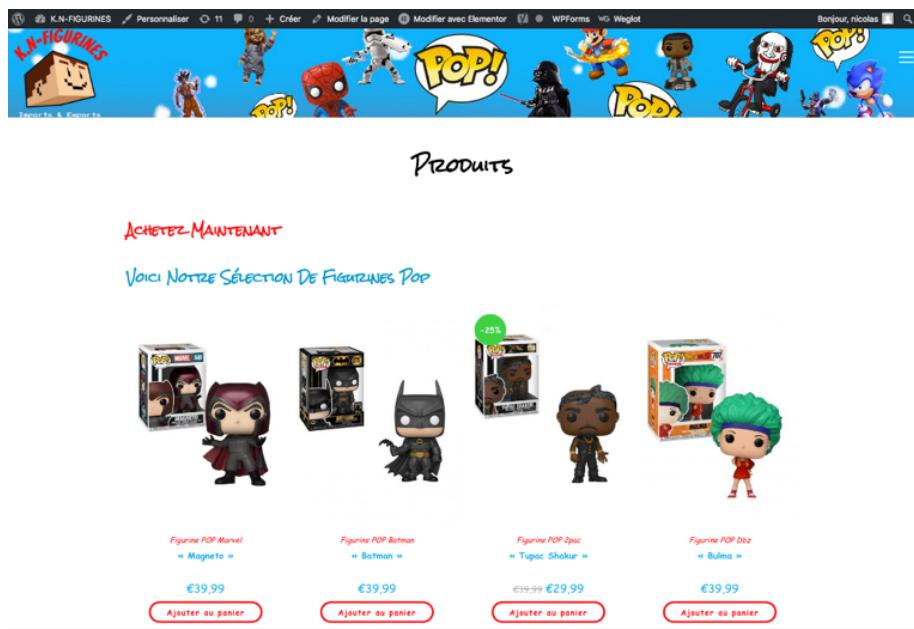


Figure 25 : Mon projet WordPress

A screenshot of the WordPress dashboard showing the "Extensions" section. The sidebar lists "Extensions installées" including Fenêtre modale, Commentaires, WooCommerce, Produits, Statistiques, Marketing, Elementor, Modèles, WPForms, Apparence, and Extensions (with 8 items). The main area shows four active plugins: Weglot Translate, WooCommerce, and WPForms Lite, each with a status bar indicating compatibility with WooCommerce and Easy Digital Downloads, version numbers, and developer information.

Figure 26 : Tableau de bord de mon projet WordPress

J'ai poursuivi en créant un **thème enfant** à partir du **thème parent OcénWP** ce qui me permet de ne pas subir une mise à jour du thème parent ce qui engendrerait des modifications irréversibles sur mon projet, et, qui serait une perte de temps.

DOSSIER PROFESSIONNEL (DP)

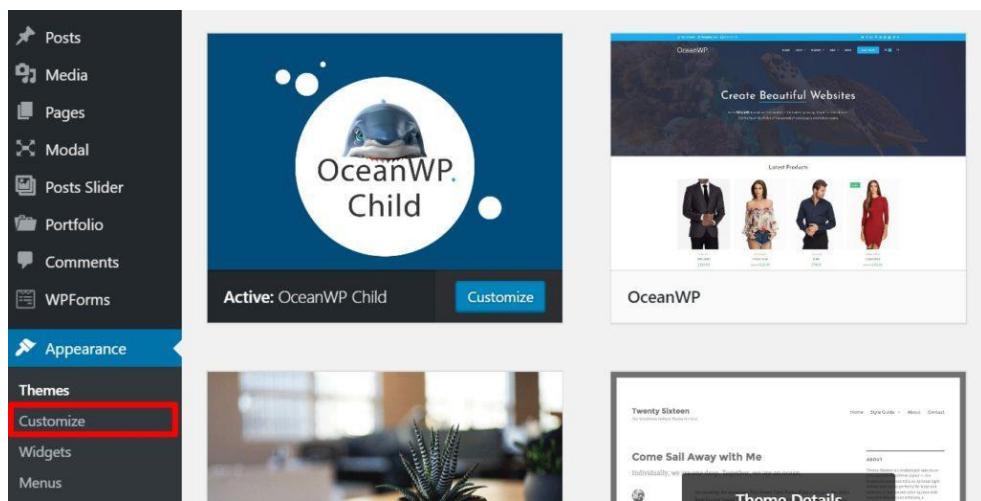


Figure 27 : Crédit du thème enfant

A la suite je me suis lancé dans la création de mon interface Ecommerce en installant des plugins :

- **WooCommerce** qui permet de faciliter le commerce en ligne en apportant des outils indispensables.
- **Weglot** qui permet la traduction de mon site web dans toutes les langues.
- **WPForms** qui permet de créer des formulaires de contact ou autre....

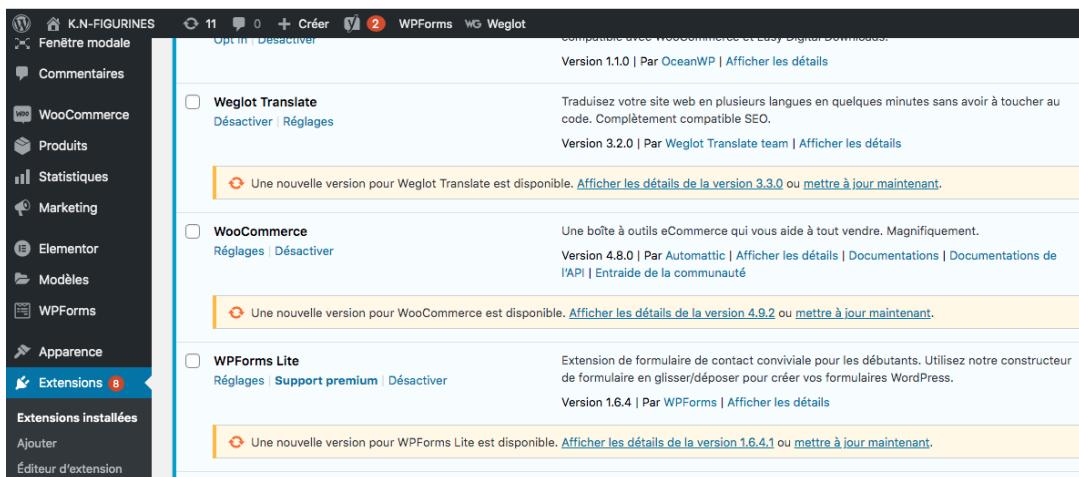


Figure 28 : Les différents plugins utilisés dans mon projet

Et je me suis servi de nombreux autres plugins déjà intégrés dans Wordpress du type **OcéanWP/Elementor** qui permet de modifier pratiquement tout ce que l'on souhaite sur notre interface d'une manière simplifiée.

2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisé pour cette quatrième compétence :

Documentation: Site de WordPress.

Logiciel: Visual studio code, Mamp, Mysql.

DOSSIER PROFESSIONNEL (DP)

Site web: *Github desktop.*

Cms: *WordPress.*

Application: *PhpMyAdmin.*

Framework: *Bootstrap, Argon Vue.js.*

Bibliothèque: *Jquery.*

Langage: *Html5, Scss3.*

3. Avec qui avez-vous travaillé ?

Pour les Compétences de 1 à 8 j'ai travaillé seul en me documentant sur internet

4. Contexte

Nom de l'entreprise, organisme ou association ➤ *Formation AFPA*

Chantier, atelier, service ➤ *Cliquez ici pour taper du texte.*

Période d'exercice ➤ Du : *05/10/2020* au : *11/06/2021*

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 2

Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité.

Compétence n°5 - Créer une base de données

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Pour cette première compétence de la partie **Back-end**, j'ai commencé par comprendre le fonctionnement d'un site web et savoir faire la différence entre la partie clients et la partie serveurs, j'ai également appris à différencier un site **Statique** d'un site **Dynamique**.

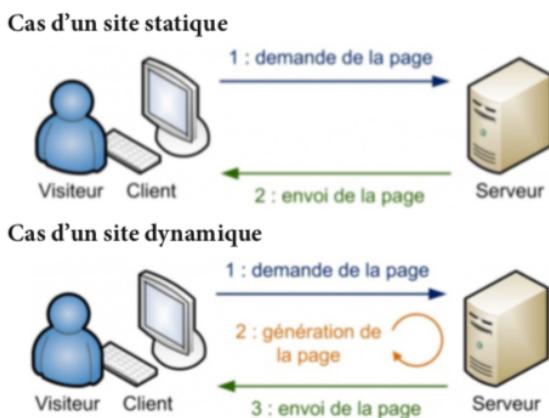


Figure 29 : La différence entre un site statique et un site dynamique

Sur un site Statique le serveur stocke des pages web et les envoie aux clients qui les demandent sans les modifier. Alors que sur un site Dynamique la page web est générée à chaque fois qu'un client la réclame, cela rend son contenu vivant il peut changer d'un instant à l'autre. Tout cela est possible grâce au langage **Php** et au Système de gestion de bases données relationnel (*SGBDr*) **MySQL**.

Ensuite j'ai installé **Mamp** (mac), **Xampp** (universel) ou **Wamp** (windows) pour accéder à **MyPhpAdmin** ce qui m'a permis de créer une base de données pour faire fonctionner mon code Php.

DOSSIER PROFESSIONNEL (DP)

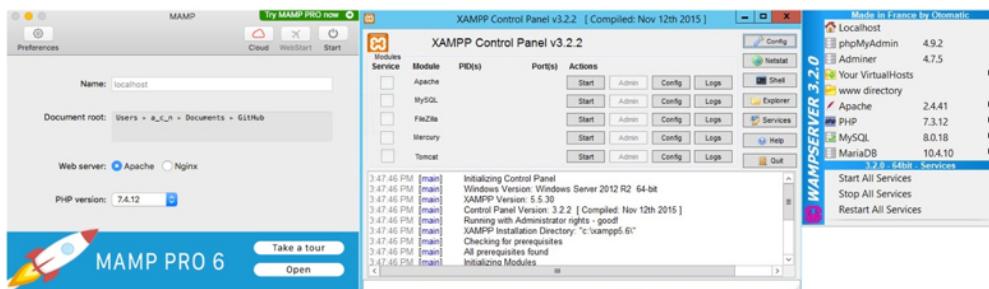


Figure 30 : Mamp, Xampp et Wamp

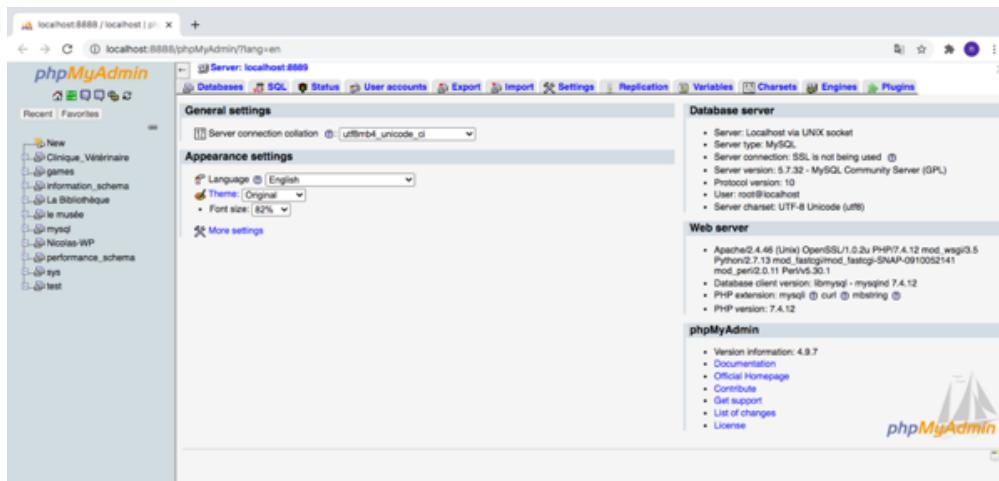


Figure 31 : Page serveur local Php My Admin

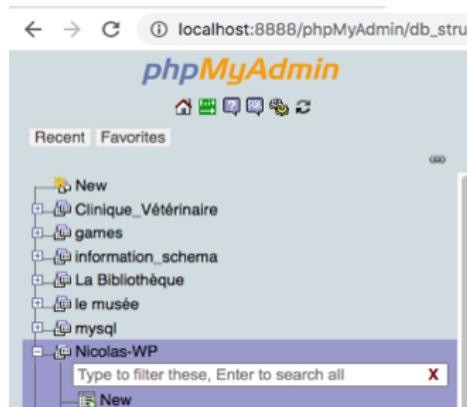
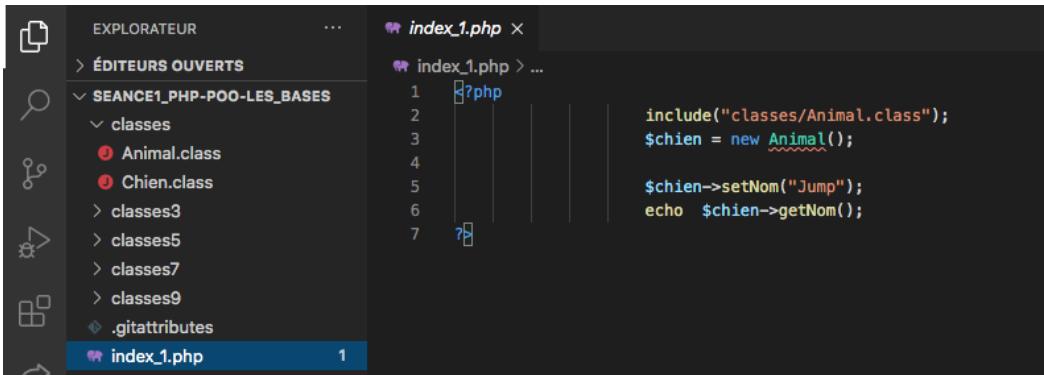


Figure 32 : Création base de données

DOSSIER PROFESSIONNEL (DP)



```
index_1.php
index_1.php > ...
1 <?php
2
3 include("classes/Animal.class");
4 $chien = new Animal();
5
6 $chien->setNom("Jump");
7 echo $chien->getNom();
```

Figure 33 : Code Php

A la suite j'ai appris à transmettre des données de page en page avec l'**Url** (*Uniform Ressource Locator*).

Imaginons que votre site s'appelle monsite.com et que vous avez une page PHP intitulée bonjour.php.

Pour accéder à cette page, vous devez aller à l'URL suivante :

<http://www.monsite.com/bonjour.php>

Si on veut maintenant envoyer des paramètres à la page PHP il suffit d'ajouter à cette URL après un point d'interrogation les informations suivantes telles que :

<http://www.monsite.com/bonjour.php?nom=Dupont&prenom=Jean>

Figure 34 : La transmission de données avec l'URL

J'ai également appris à transmettre des données avec les formulaires et les méthodes **GET** : (*les données transiteront par l'URL*) et **POST** : (*les données ne transiteront pas par l'URL*) ainsi que Les variables **superglobales** et également les **Sessions** et les **Cookies**.



```
index.php
index.php > ...
1 <?php
2
3 <!DOCTYPE html>
4 <html lang="en">
5 <head>
6   <meta charset="UTF-8">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8   <title>Document</title>
9 </head>
10 <body>
11 <a href="bonjour.php?nom=Dupont&prenom=Jean">Dis-moi bonjour !</a>
12 </body>
13 </html>
```

Figure 35 : HTML pour la méthode GET



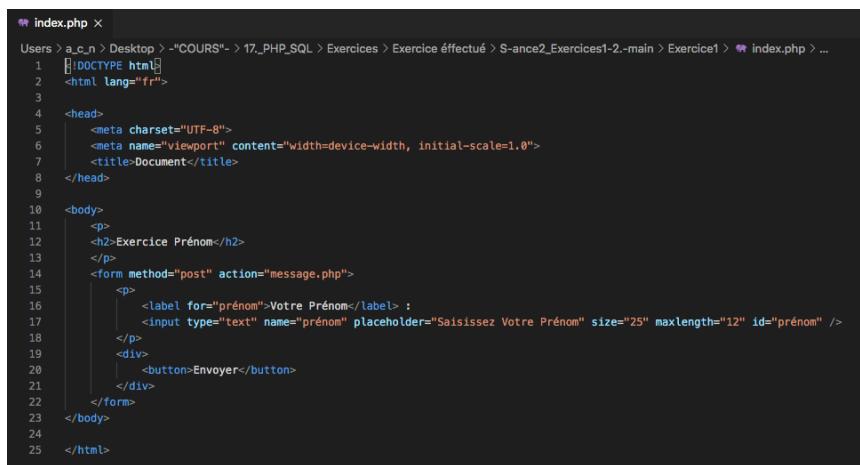
```
bonjour.php
bonjour.php > ...
1 <p>Bonjour <?php echo $_GET['prenom'] . ' ' . $_GET['nom']; ?> !</p>
```

DOSSIER PROFESSIONNEL (DP)

Figure 36 : La méthode GET

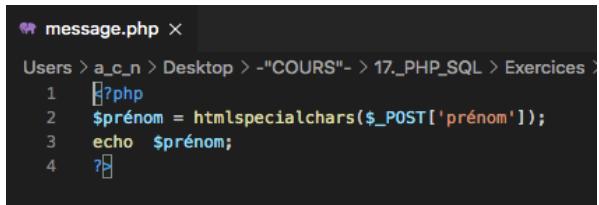
Dis-moi bonjour !

Figure 37 : L'écho du résultat de la méthode GET



```
index.php
1 <!DOCTYPE html>
2 <html lang="fr">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <p><b>Exercice Prénom</b></p>
12   <form method="post" action="message.php">
13     <p>
14       <label for="prénom">Votre Prénom</label> :
15       <input type="text" name="prénom" placeholder="Saisissez Votre Prénom" size="25" maxlength="12" id="prénom" />
16     </p>
17     <div>
18       <button>Envoyer</button>
19     </div>
20   </form>
21 </body>
22
23 </html>
```

Figure 38 : HTML pour la méthode POST



```
message.php
1 <?php
2 $prénom = htmlspecialchars($_POST['prénom']);
3 echo $prénom;
4 ?>
```

Figure 39 : La méthode POST

Exercice Prénom

Votre Prénom :

Figure 40 : Résultat de la méthode POST

DOSSIER PROFESSIONNEL (DP)

nicolas

Figure 41 : L'écho du résultat de la méthode POST

Et j'ai poursuivi mon apprentissage en apprenant à stocker des informations dans une base de données avec le langage **SQL** en créant des **Tables** et en modifiant leurs contenus.

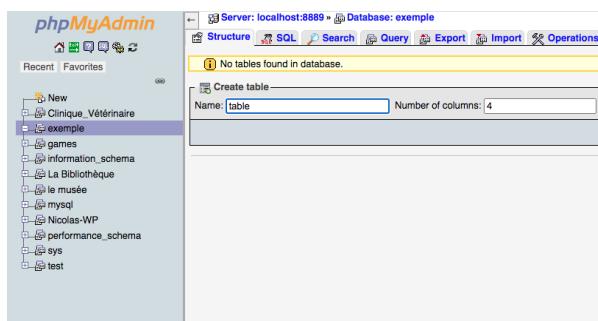


Figure 42 : Création d'une table

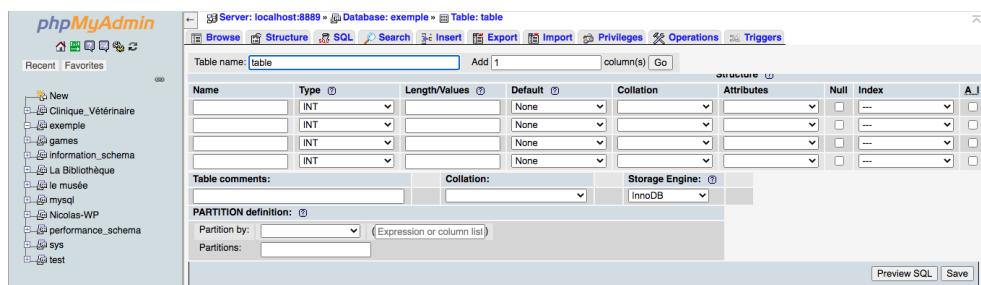


Figure 43 : La modification d'une table

J'ai aussi appris à lire des données en autorisant l'accès de PHP aux bases de données MySQL via le **PDO** : (*PHP Data Objects*). Ainsi pour les récupérer grâce aux **requêtes**, et les écrire avec la requête **INSERT INTO** qui permet d'ajouter une entrée.

```
<?php
try
{
    $pdo_options[PDO::ATTR_ERRMODE] = PDO::ERRMODE_EXCEPTION;

    $db = new PDO('mysql:host=localhost;dbname=games', 'root', 'root', $pdo_options);
```

Figure 44 : Connexion à MySQL via PDO

DOSSIER PROFESSIONNEL (DP)

Votre première requête SQL

Code : SQL

```
SELECT * FROM 'news'
```

```
$requete = $db->query('SELECT UPPER(nom) AS nom_maj FROM jeux_video');
```

Figure 45 : Les requêtes

Ecrire des données

INSERT : ajouter des données

La requête INSERT INTO permet d'ajouter une entrée

Code : SQL

```
INSERT INTO news(id, auteur, titre, contenu, dateAjout, dateModif) VALUES(",'Patrick', 'Ma première info', 'Bonjour le monde !', NOW(), NOW())
```

Figure 46 : La requête INSERT INTO

Et puis j'ai continué avec les **Fonctions Scalaires SQL**. Cela crée un "**champ virtuel**" qui n'existe que le temps de la requête, et j'ai poursuivi sur les différents types de dates que peut stocker MySQL et j'ai également travaillé sur les **jointures** entre les tables ce qui permet de pouvoir créer des **relations** entre elles et de les liées.

Les fonctions SQL

Les fonctions scalaires

Nous allons d'abord découvrir le mode d'emploi d'une fonction SQL de type scalaire : la fonction UPPER.

Utiliser une fonction scalaire SQL

Code : SQL

```
SELECT UPPER(nom) FROM jeux_video
```

Figure 47 : Les fonctions scalaires SQL

Les dates en SQL

Les champs de type date

Les différents types de dates

Voici les différents types de dates que peut stocker MySQL :

- DATE : stocke une date au format AAAA-MM-JJ (Année-Mois-Jour).
- TIME : stocke un moment de la journée au format HH:MM:SS (Heures:Minutes:Secondes).
- DATETIME : stocke une combinaison d'une date et d'un moment de la journée au format AAAA-MM-JJ HH:MM:SS. Ce type de champ est donc plus précis.
- TIMESTAMP : stocke une date et un moment de la journée sous le format AAAAMMJJHHMMSS.
- YEAR : stocke une année, soit au format AA, soit au format AAAA.

Utilisation des champs de date en SQL

Si le champ est de type DATETIME, il faut indiquer précisément les heures, minutes et secondes :

Code : SQL

```
SELECT pseudo, message, date FROM minichat WHERE date = '2017-24-01' 14:18:45'
```

Pour récupérer la liste des messages postés entre le 24/01/2017 et le 27/01/2017 :

Code : SQL

```
SELECT pseudo, message, date FROM minichat WHERE date BETWEEN '2017-24-01 00:00:00' AND '2017-27-01 00:00:00'
```

Figure 48 : Les différents types de dates que peut stocker MySQL

DOSSIER PROFESSIONNEL (DP)

Le nouveau champ `ID_proprietaire` est de type `INT`. Il permet de faire référence à une entrée précise de la table `proprietaires`.

On peut maintenant considérer que les tables sont reliées à travers ces ID de propriétaires, comme le suggère la figure suivante.

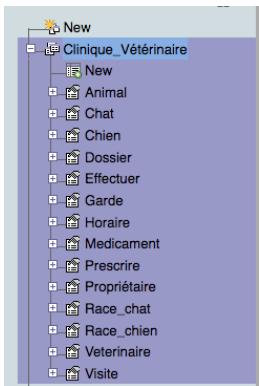
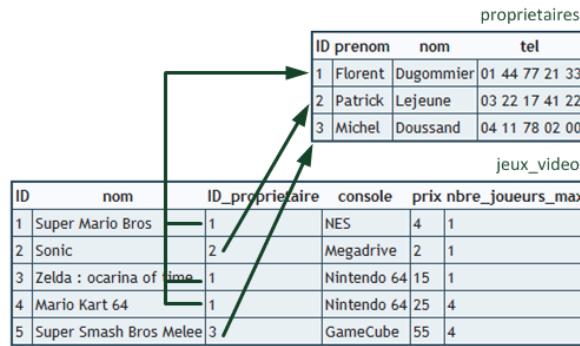
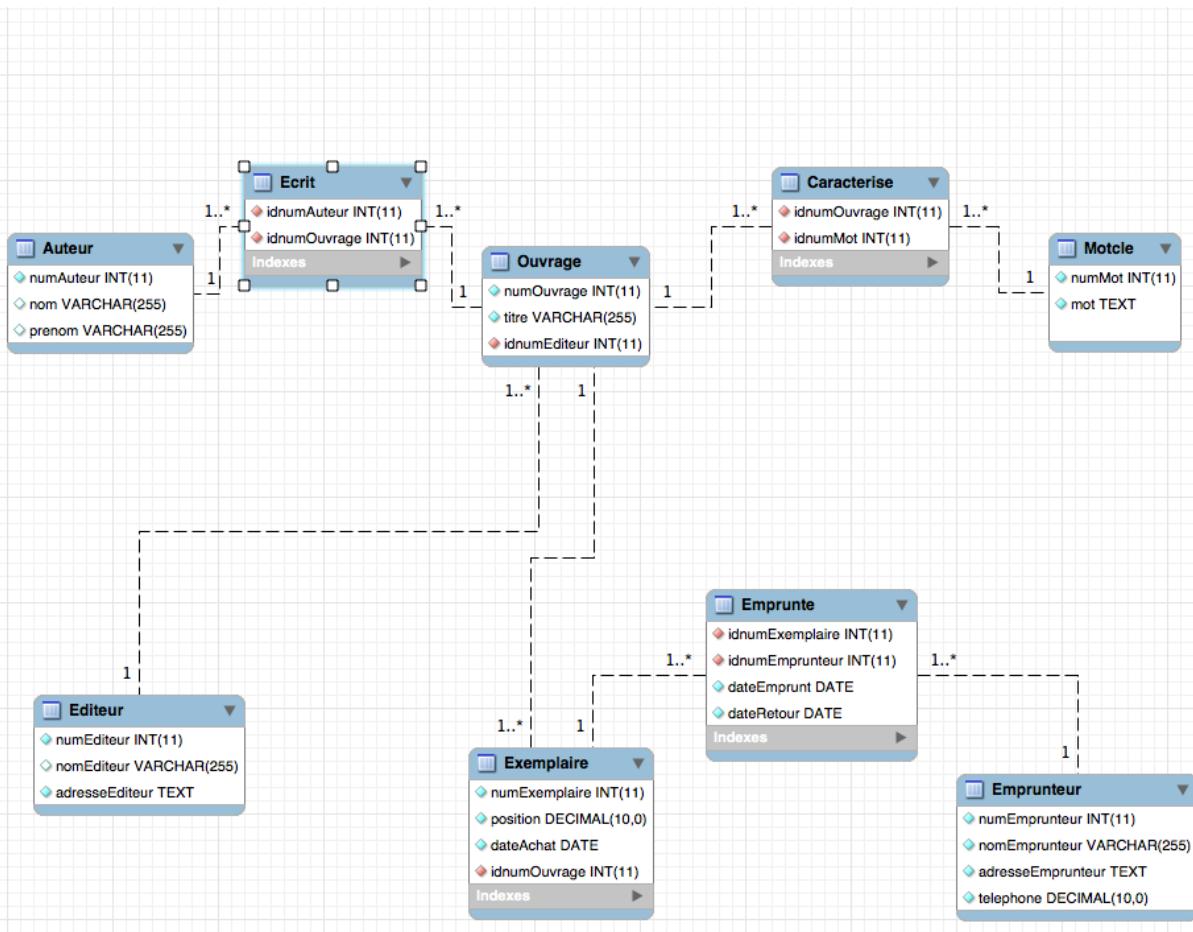


Figure 49 : Les jointures entre les tables

Figure 50 : La base de données

J'ai poursuivi avec le langage de modélisation UML (*Langage de Modélisation Unifié*). Il permet, grâce à un ensemble de diagrammes très explicites, de représenter l'architecture et le fonctionnement des systèmes informatiques complexes en tenant compte des relations entre les concepts utilisés et l'implémentation qui en découle.

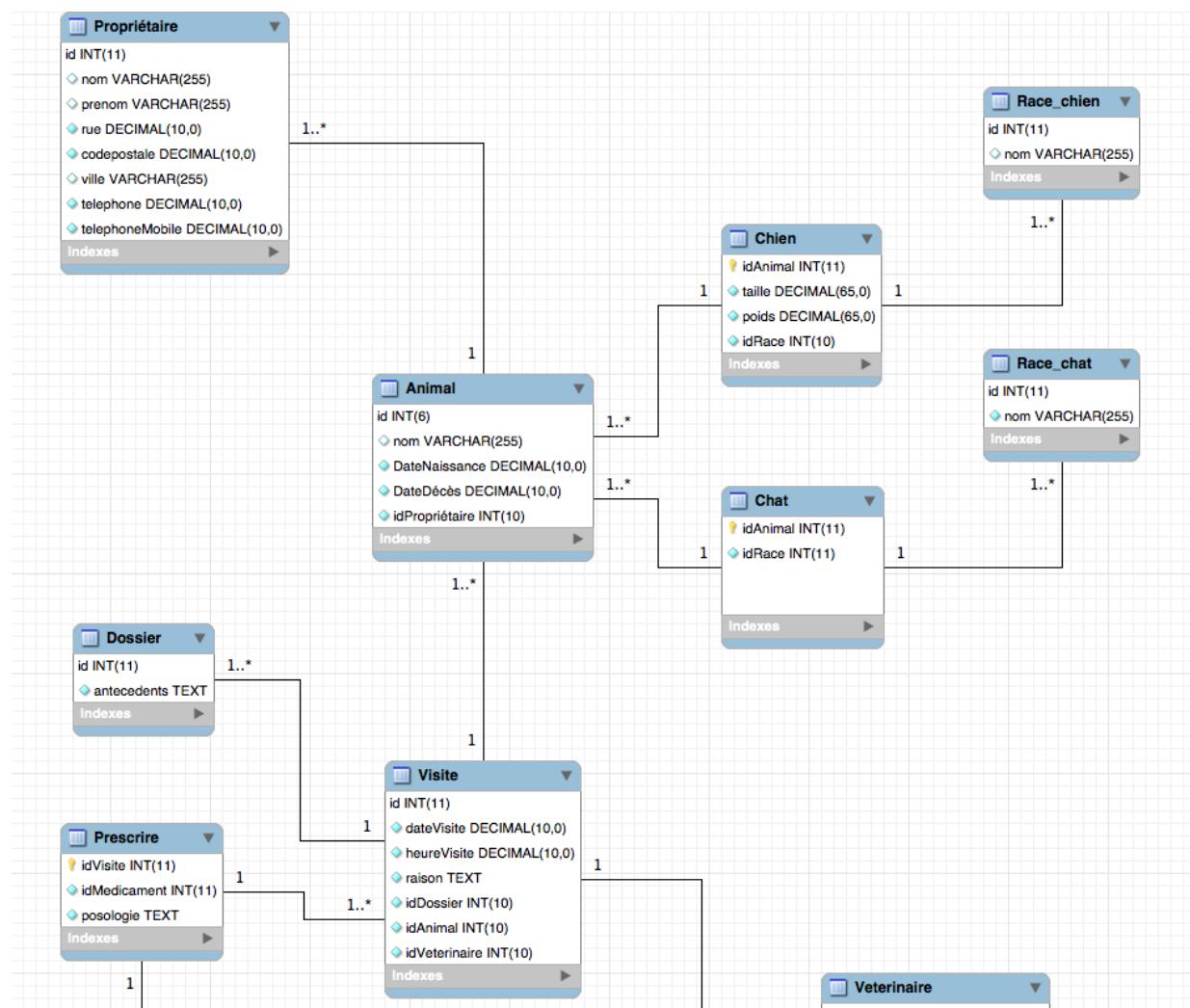
DOSSIER PROFESSIONNEL (DP)



terminer cette compétence j'ai créé deux bases de données avec Php MySql, les thèmes des deux projets était : (*la Bibliothèque et le Musée*). Pour lesquelles je me suis occupé de la création des **diagrammes** avec le **MCD** (*Modèle Conceptuel de Données*) en y ajoutant les **contraintes d'intégrité** souhaitées dans les exercices tout cela créé via **MYSQL WorkBench**. Le **MCD** permet d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités.

Figure 51 : Le modèle MCD

DOSSIER PROFESSIONNEL (DP)



Pour l'exercices demandés dans cette compétence j'ai commencé par créer une base de données qui correspond à la gestion d'une clinique vétérinaire ensuite je me suis servi du **MLD** (*modèle logique de données*) qui est la représentation des données d'un système d'information. Les données sont représentées en prenant en compte le modèle technologique qui sera utilisée pour leur gestion.

Figure 52 : Le modèle MLD

2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisé pour cette cinquième compétence :

Logiciel: Visual studio code, Mamp, Mysql, Mysql-WorkBench.

Site web: Github desktop.

DOSSIER PROFESSIONNEL (DP)

Application: *PhpMyAdmin.*

Langage: *Html5, Css3, Sql, Uml.*

3. Avec qui avez-vous travaillé ?

Pour les Compétences de 1 à 8 j'ai travaillé seul en me documentant sur internet

4. Contexte

Nom de l'entreprise, organisme ou association ➤ *Formation AFPA*

Chantier, atelier, service ➤ *Cliquez ici pour taper du texte.*

Période d'exercice ➤ Du : *05/10/2020* au : *11/06/2021*

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Activité-type 2

Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité.

Compétence n°6 - Développer les composants d'accès aux données

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Pour cette sixième compétence J'ai commencé par réaliser la page d'accueil d'un blog en PHP.

The image shows a code editor window with the file 'menu.php' open. The code is a PHP script with HTML and CSS embedded. It defines a navigation bar with links to Accueil, Blog, CV, Contact, and a Contact section. A sidebar labeled 'Aside' is also present. Below the code editor is a screenshot of the blog's homepage. The header contains the title 'Ma page' and a navigation bar with links to Accueil, Blog, CV, Contact, and Contact. The main content area is labeled 'Article'. At the bottom right of the page is a red footer bar with the text 'Copyright © 2016'.

```
<?php
...
<div class="pos-f-t">
  <div class="collapse" id="navbarToggleExternalContent">
    <div class="bg-dark p-4">
      <h4 class="text-black">
        <ul>
          <li><a href="#">Accueil </a> </li>
          <li><a href="#">Blog </a> </li>
          <li><a href="#">CV </a> </li>
          <li><a href="#">Contact </a> </li>
          <li><a href="#">Contact </a> </li>
        </ul>
      </h4>
    </div>
  </div>
<nav class="navbar navbar-dark bg-dark">
  <button class="navbar-toggler" type="button" data-toggle="collapse"
    <span class="navbar-toggler-icon">
      <span id="menut"></span>
      <span id="menut"></span>
      <span id="menut"></span>
    </span>
  </button>
</nav>
</div>
```

Figure 53 : Le code Php et La page d'accueil de mon blog en Php

J'ai continué sur la réalisation d'un blog avec Le design pattern **MVC** (Modèle Views Controller) administré **POO** (Programmation Orientée Objet).

DOSSIER PROFESSIONNEL (DP)

Figure 54 : Le modèle MVC

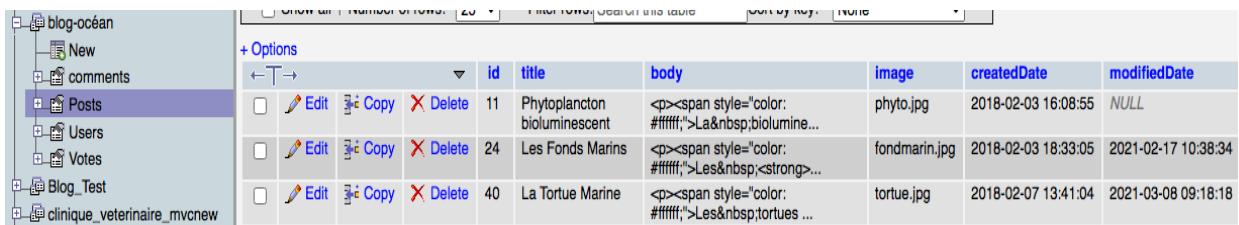
Titre	Date	Action
La Tortue Marine	Le 07/02/2018 à 13:41	MODIFIER SUPPRIMER
Les Fonds Marins	Le 03/02/2018 à 18:33	MODIFIER SUPPRIMER
Phytoplancton bioluminescent	Le 03/02/2018 à 16:08	MODIFIER SUPPRIMER

Dans le Modèle **MVC** on peut y intégrer le **CRUD** : (*create, read, update, delete*) qui nous permet de pouvoir : Créer – Lire – Mettre à jour et supprimer. Ce sont les quatre opérations de base pour la persistance des données, en particulier le stockage d'informations en base de données.

Figure 55 : Le CRUD

Voilà ce qui se passe dans la base de données au moment de l'ajout, la modification, la mise à jour ou la suppression de données grâce au **CRUD** intégré.

DOSSIER PROFESSIONNEL (DP)



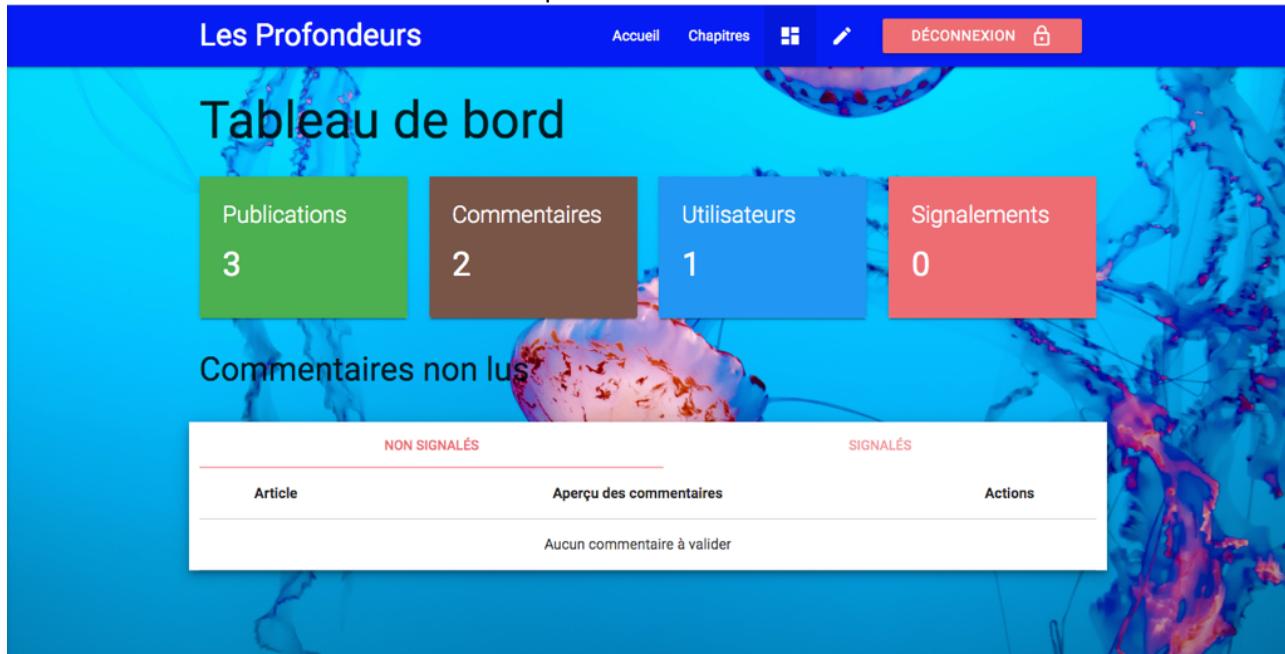
A screenshot of a database management interface. On the left, there's a sidebar with a tree view of databases: 'blog-ocean' (selected), 'New', 'comments', 'Posts' (selected), 'Users', 'Votes', 'Blog_Test', and 'clinique_veterinaire_mvcnew'. The main area shows a table titled '+ Options' with columns: id, title, body, image, createdDate, and modifiedDate. There are three rows of data:

	<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	id	title	body	image	createdDate	modifiedDate
		11	Phytoplancton bioluminescent	<p>La biolumine...	phyto.jpg	2018-02-03 16:08:55	NULL
		24	Les Fonds Marins	<p>Les ...	fondmarin.jpg	2018-02-03 18:33:05	2021-02-17 10:38:34
		40	La Tortue Marine	<p>Les tortues ...	tortue.jpg	2018-02-07 13:41:04	2021-03-08 09:18:18

Figure 56 : La base de données au moment de sa modification

Je termine cette compétence en présentant mon Blog **MVC** personnalisé.

Voici le Dashboard (*Tableau de bord*). C'est un outil d'évaluation constitué de plusieurs indicateurs de performance à des moments donnés ou sur des périodes données.



A screenshot of a personalized MVC blog dashboard titled 'Les Profondeurs'. The top navigation bar includes 'Accueil', 'Chapitres', a settings icon, and 'DÉCONNEXION'. The main section is titled 'Tableau de bord' and features four colored boxes with statistics: 'Publications' (3, green), 'Commentaires' (2, brown), 'Utilisateurs' (1, blue), and 'Signalements' (0, red). Below this, a section titled 'Commentaires non lus' shows a table with two columns: 'NON SIGNALÉS' and 'SIGNALÉS'. The 'NON SIGNALÉS' column has headers 'Article', 'Aperçu des commentaires', and 'Actions', with a note 'Aucun commentaire à valider'. The 'SIGNALÉS' column is empty.

Figure 57 : Le dashboard(tableau de bord)

DOSSIER PROFESSIONNEL (DP)

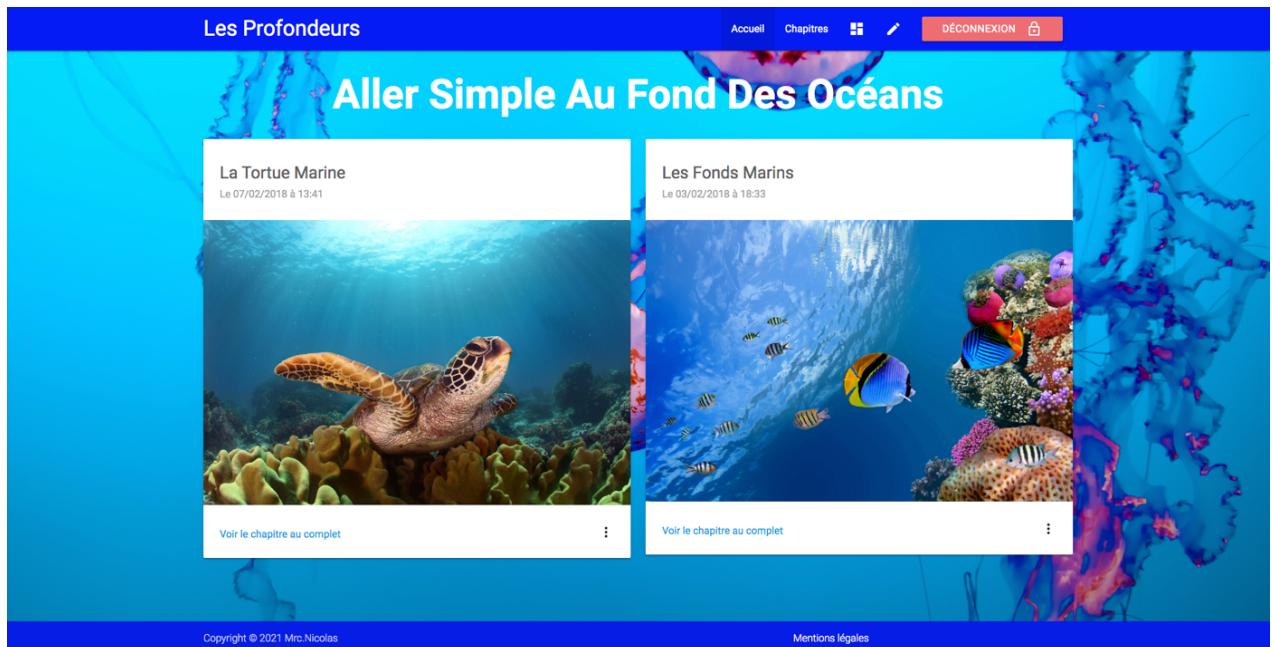
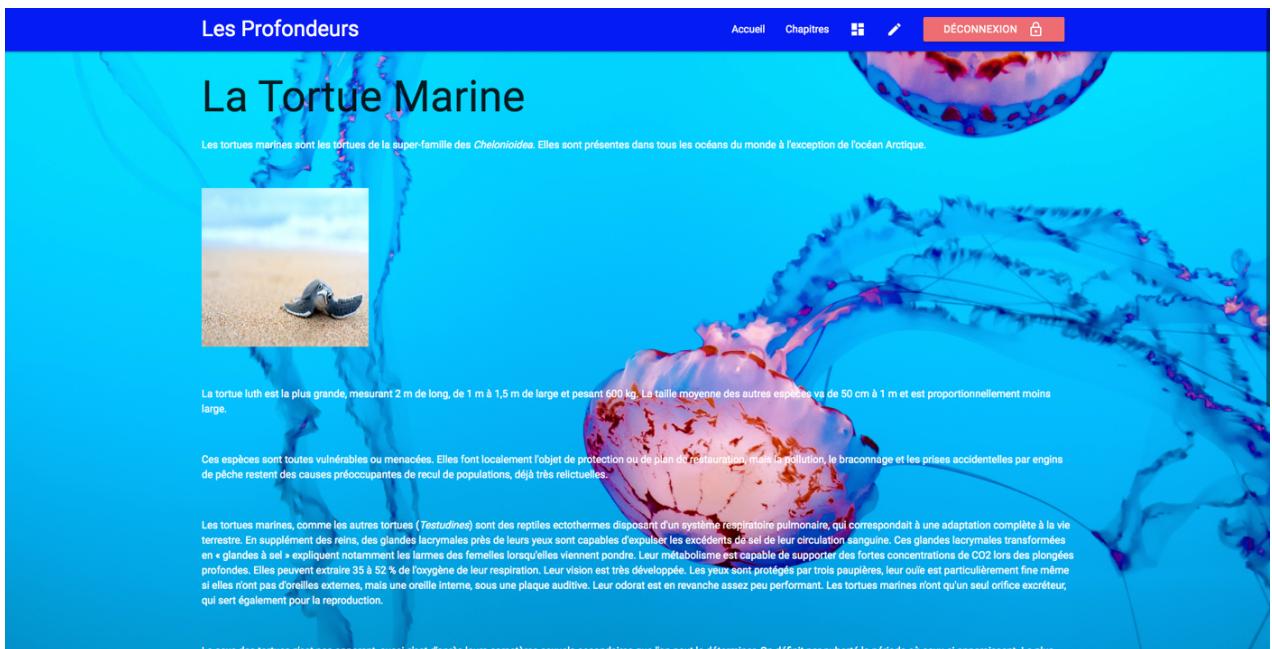


Figure 58 : Mon blog MVC



2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisé pour cette sixième compétence :

Logiciel: Visual studio code, Mamp, Mysql.

Site web: Github desktop.

Application: PhpMyAdmin.

Langage: Html5, Css3, Php.

DOSSIER PROFESSIONNEL (DP)

3. Avec qui avez-vous travaillé ?

Pour les Compétences de 1 à 8 j'ai travaillé seul en me documentant sur internet

4. Contexte

Nom de l'entreprise, organisme ou association ➤ *Formation AFPA*

Chantier, atelier, service ➤ *Cliquez ici pour taper du texte.*

Période d'exercice ➤ Du : *05/10/2020* au : *11/06/2021*

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Activité-type 2

Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité.

Compétence n°7 - Développer la partie back-end d'une application web ou web mobile

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Pour cette septième compétence j'ai commencé par lancer l'éditeur de code **Visual Studio Code**, ouvrir le terminal et y installer **Symfony** : (*Symfony est un ensemble de composants PHP ainsi qu'un framework MVC libre écrit en PHP Il fournit des fonctionnalités modulables et adaptables qui permettent de faciliter et d'accélérer le développement d'un site web*). En exécutant cette commande illustrée ci-dessous.

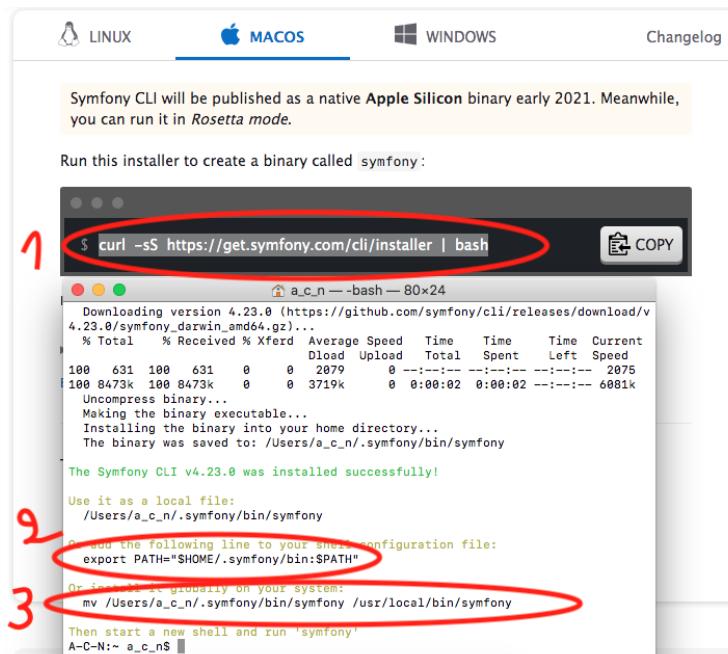


Figure 59 : Installation de Symfony

A la suite j'ai également installé **Composer** : (*Composer est un logiciel gestionnaire de dépendances libre écrit en PHP. Il permet à ses utilisateurs de déclarer et d'installer les bibliothèques dont le projet principal a besoin*). En exécutant les commandes illustrées ci-dessous.

Pour installer rapidement Composer dans le répertoire courant, exécutez le script suivant dans votre terminal. Pour automatiser l'installation, utilisez le guide d'installation de Composer par programmation .

```
php -r "copie ('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file ('sha384', 'composer-setup.php') === '756890a4488ce9024fc62c56153228907f1545c22'
php composer-setup.php
php -r "unlink ('composer-setup.php');"
```

Figure 60 : Installation de composer

Après avoir vérifié les versions de **Symfony**, **Composer** et **PHP**, J'ai poursuivi sur la création du projet **Symfony** en tapant dans le terminal la commande : (*composer create-project symfony/website-skeleton*).

DOSSIER PROFESSIONNEL (DP)

Ce qui nous permet d'avoir cette structure illustrée ci-dessous.

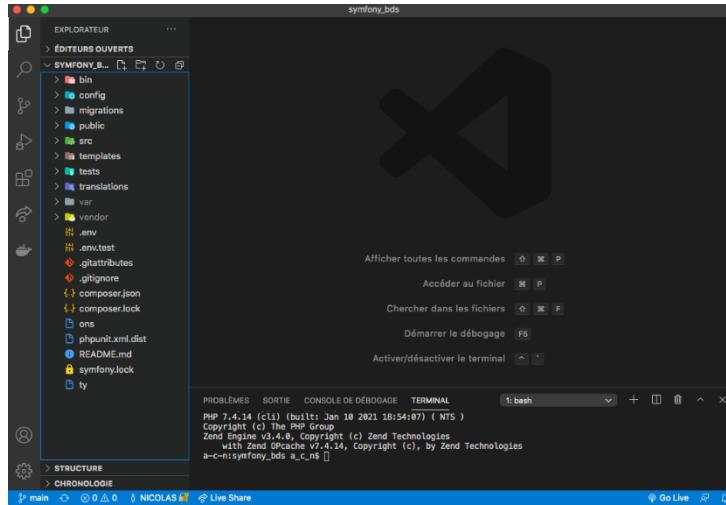


Figure 61 : La structure de Symfony obtenue dans Visual studio code

Ensuite je lance mon projet **Symfony** en tapant en ligne de commande : `symfony server -d`.

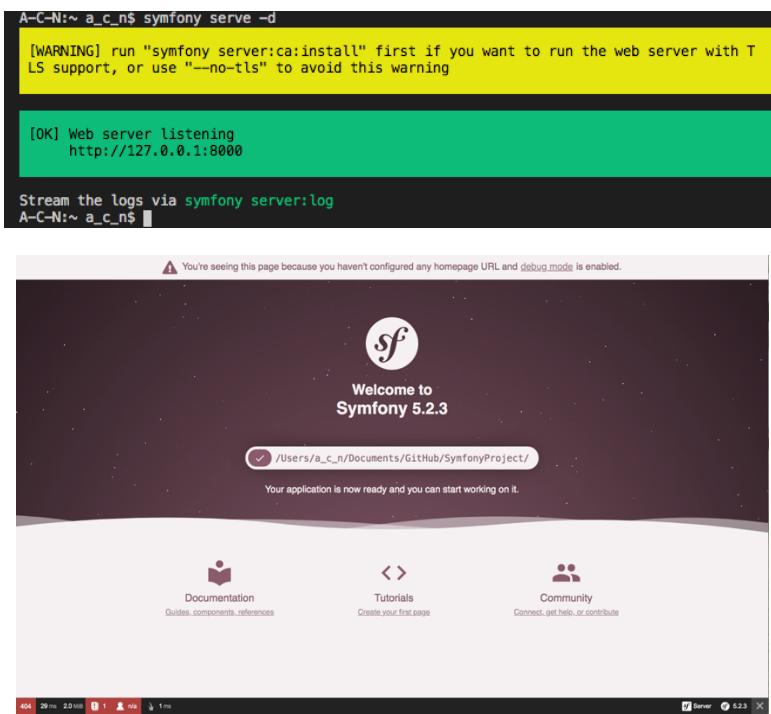


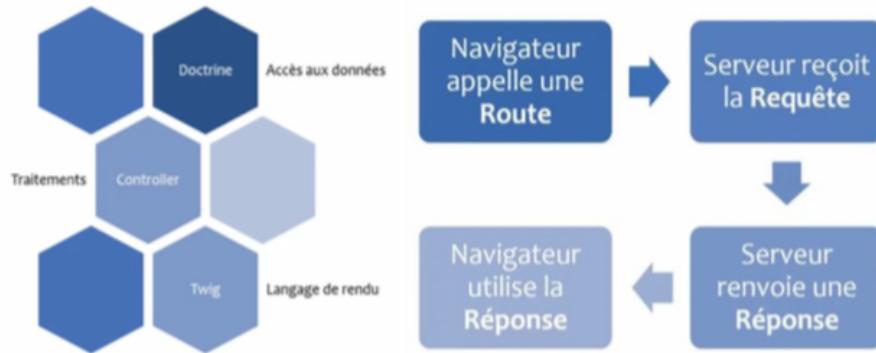
Figure 62 : Le démarrage du projet Symfony

Je continue sur la création d'une application web en créant une première page, pour ça, il faut créer un **controller** avec ces **routes** en installant le **maker-bundle** en tapant en ligne de commande : (`composer require symfony/maker-bundle --dev`) et (`composer require doctrine/annotations`). Ensuite je tape en ligne de commande : (`php bin/console make:controller`) pour la création du controller et je relance le serveur.

DOSSIER PROFESSIONNEL (DP)

Le Controller permet de créer des fonctionnalités dans symfony voici un exemple de son fonctionnement ci-dessous ainsi que la définition de ces routes.

Le cœur de Symfony La logique du Controller

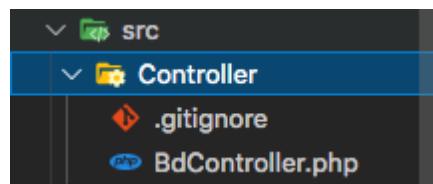


Les routes

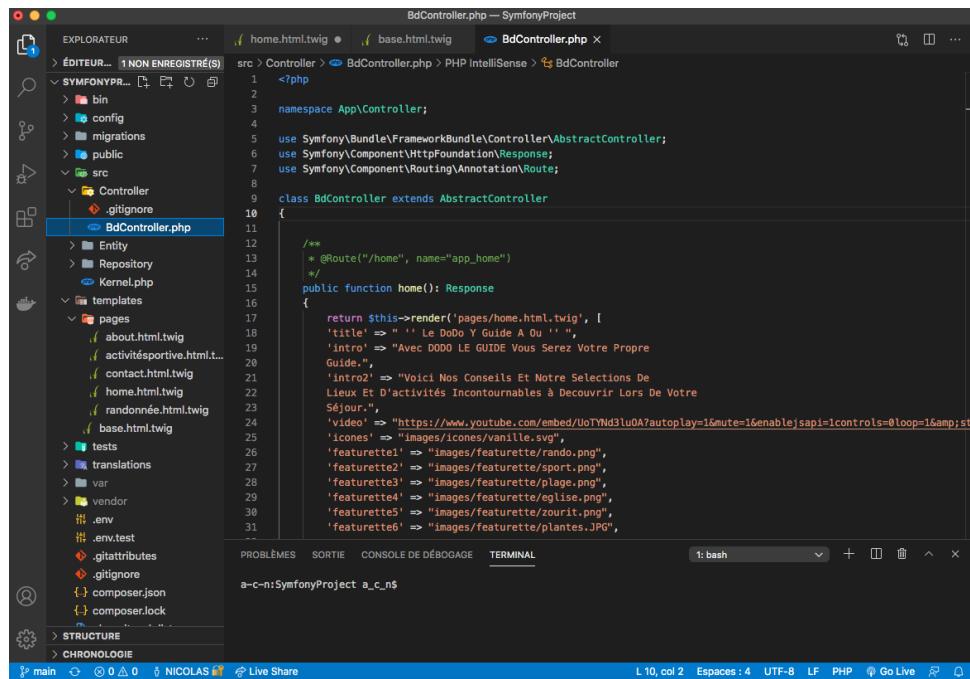
Une route est une correspondance entre une URL et une fonction. Celle-ci se trouve dans le controller. Une route peut être configurée par une annotation (mais aussi dans un fichier yaml).

Figure 63 : Le controller et ces routes

Voici mon controller créé.



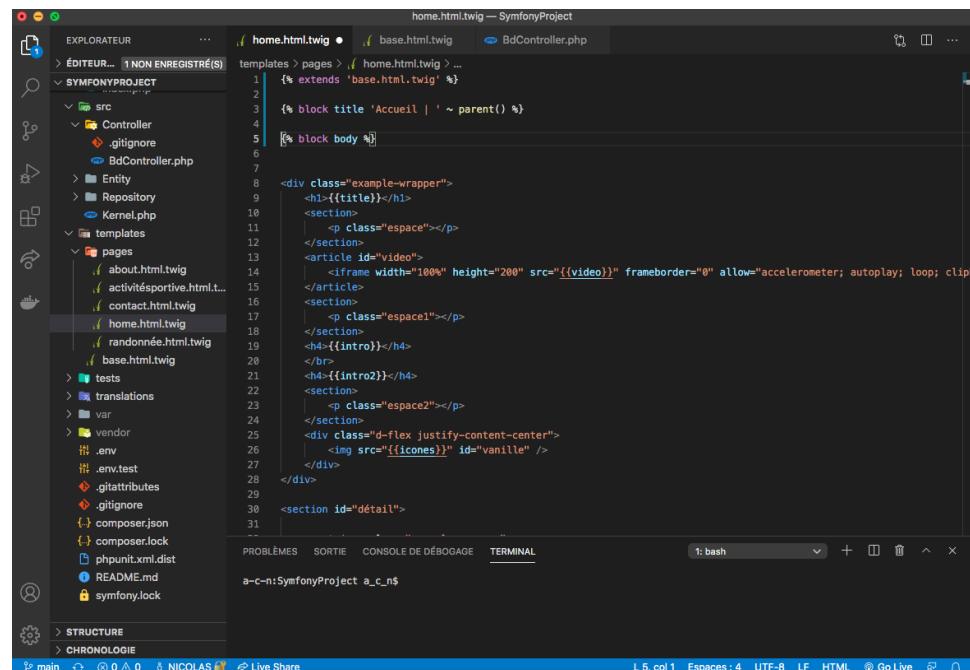
DOSSIER PROFESSIONNEL (DP)



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "SYMFONYPROJECT".
- Editor:** Displays the code for `BdController.php`. The code defines a controller class `BdController` extending `AbstractController`. It contains a `home()` method that returns a response with a template named `home.html.twig`.
- Terminal:** Shows the command `a-c-n:SymfonyProject a_c_n$` being run.
- Status Bar:** Shows the current file is `BdController.php`, the line count is L 10, column count is col 2, and encoding is UTF-8.

Figure 64 : Le controller créé



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "SYMFONYPROJECT".
- Editor:** Displays the code for `home.html.twig`. The template extends `base.html.twig` and contains sections for title and body. It includes HTML for a video player and two paragraphs of text.
- Terminal:** Shows the command `a-c-n:SymfonyProject a_c_n$` being run.
- Status Bar:** Shows the current file is `home.html.twig`, the line count is L 5, column count is col 1, and encoding is UTF-8.

Figure 65 : Code de la première page de mon application web sur VisualStudioCode avec Symfony

DOSSIER PROFESSIONNEL (DP)

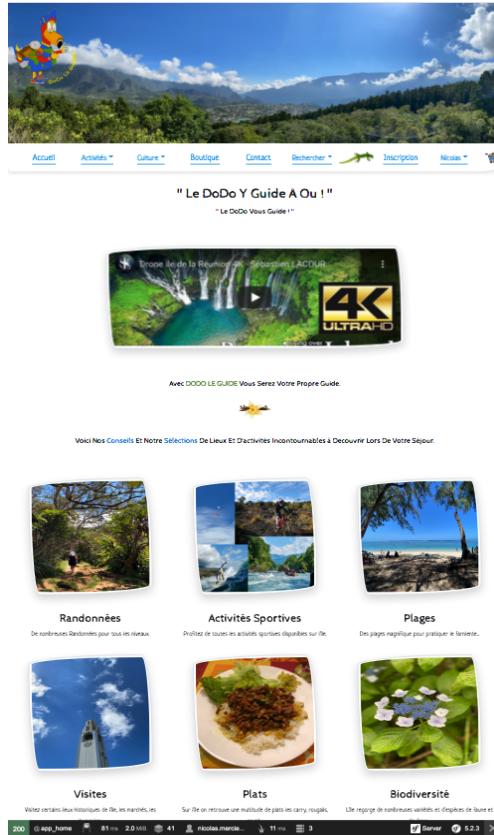


Figure 66 : Résultat sur le navigateur du code ci-dessus

Symfony propose un langage de **templates** puissant, appelé **Twig**.

The screenshot shows the Twig documentation homepage. At the top, there's a header with the SymfonyHub logo, the SymfonyCloud logo, and the text "SymfonyCloud: The best way to host your Symfony project". Below the header, there's a large green banner with the word "Twig" in white and the text "The flexible, fast, and secure template engine for PHP". To the left of the banner is a stylized green plant icon. To the right is a vertical bar with three colored segments: light green, medium green, and dark blue. The main content area has a white background. It starts with a section titled "Twig is a modern template engine for PHP" with a bulleted list of features: "Fast", "Secure", and "Flexible". Below this is a section titled "Why yet another template engine?" with a paragraph of text and a code block. The code block contains two lines of PHP-like syntax:

```
<?php echo $var ?>
<?php echo htmlspecialchars($var, ENT_QUOTES, 'UTF-8') ?>
```

DOSSIER PROFESSIONNEL (DP)

Langage de rendu Twig

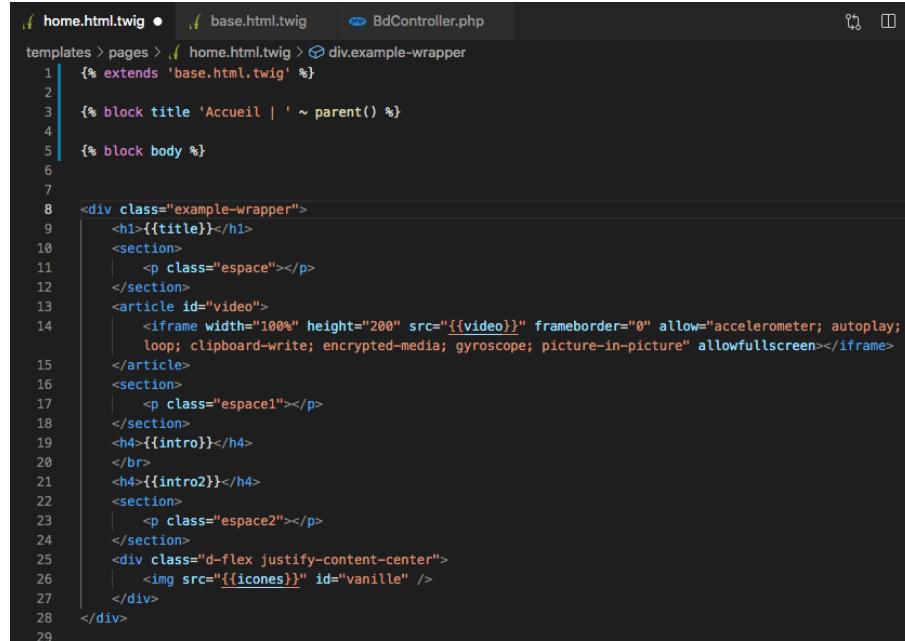
Simplicité

- Facilite l'écriture des affichages
- Apporte beaucoup de fonctionnalités

Absence de PHP

- Permet d'abstraire les affichages de balises PHP
- Plus simple pour un intégrateur

Figure 67 : Page d'accueil Twig et sa documentation

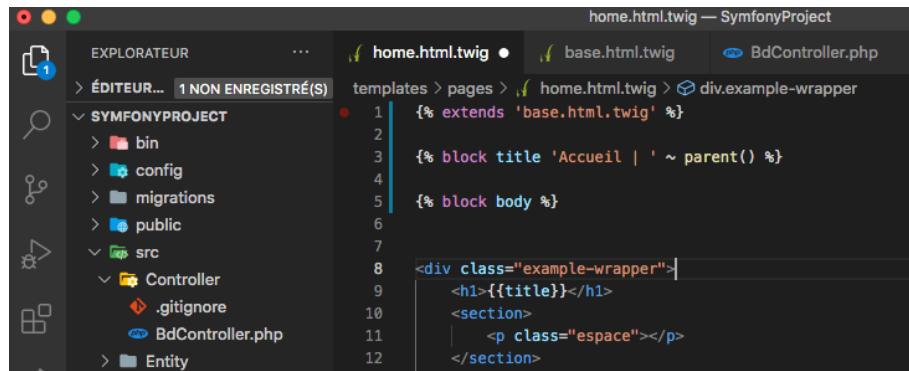


```
home.html.twig • base.html.twig BdController.php
templates > pages > home.html.twig > div.example-wrapper
1  {% extends 'base.html.twig' %}
2
3  {% block title 'Accueil | ' ~ parent() %}
4
5  {% block body %}
6
7
8  <div class="example-wrapper">
9    <h1>{{title}}</h1>
10   <section>
11     <p class="espace"></p>
12   </section>
13   <article id="video">
14     <iframe width="100%" height="200" src="{{video}}" frameborder="0" allow="accelerometer; autoplay; loop; clipboard-write; encrypted-media; gyroscope; picture-in-picture" allowfullscreen></iframe>
15   </article>
16   <section>
17     <p class="espace1"></p>
18   </section>
19   <h4>{{intro}}</h4>
20   <br>
21   <h4>{{intro2}}</h4>
22   <section>
23     <p class="espace2"></p>
24   </section>
25   <div class="d-flex justify-content-center">
26     
27   </div>
28 </div>
29
```

Figure 68 : L'écriture Twig dans VisualStudioCode

Les principales fonctionnalité sont qu'elles peuvent afficher le contenu d'une **variable** exemple avec :
L'interpolation **{} var {}** on utilise les **doubles-accolades** pour afficher le contenu d'une variable.

DOSSIER PROFESSIONNEL (DP)



The screenshot shows the Symfony Project Explorer interface. The left sidebar lists project files: bin, config, migrations, public, src (Controller, .gitignore, BdController.php), and Entity. The main area displays the content of the file `home.html.twig`:

```
home.html.twig — SymfonyProject
templates > pages > home.html.twig > div.example-wrapper
1  {% extends 'base.html.twig' %}
2
3  {% block title 'Accueil | ' ~ parent() %}
4
5  {% block body %}
6
7
8  <div class="example-wrapper">
9    <h1>{{title}}</h1>
10   <section>
11     <p class="espace"></p>
12   </section>
```

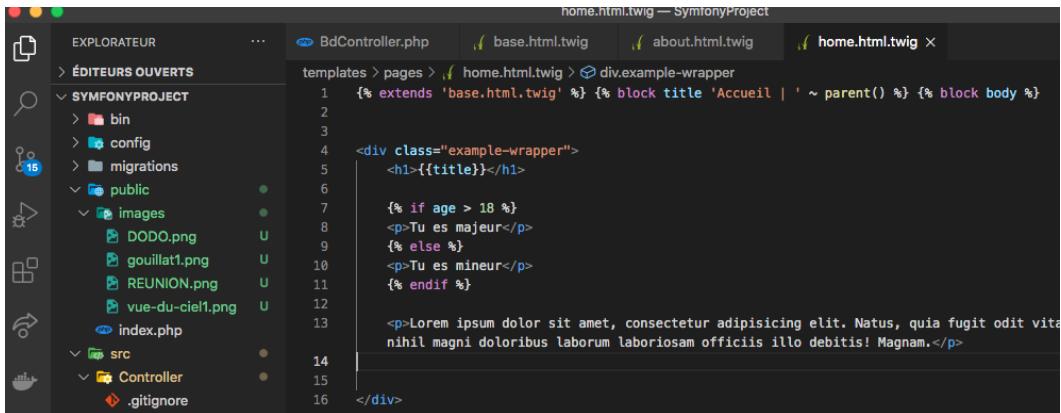
Figure 69 : Fonctionnalité : afficher le contenu d'une variable



" Le DoDo Y Guide A Ou "

Figure 70 : Résultat de la fonctionnalité de l'exemple ci-dessus

On peut aussi appeler une structure de contrôle avec la commande : `{% cmd %}` et pour se faire on utilise l'**accolade pourcentage** pour appeler.

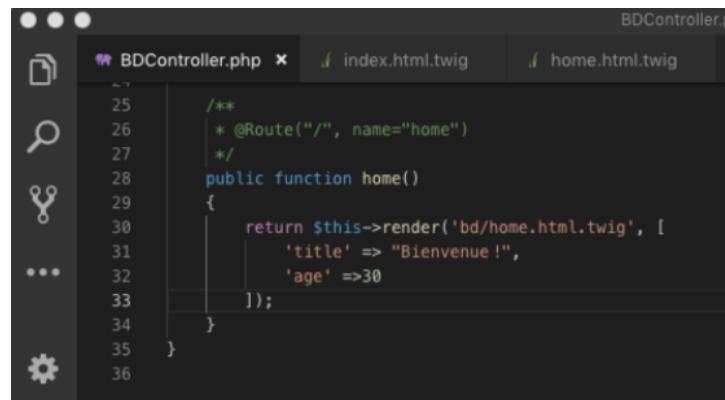


The screenshot shows the Symfony Project Explorer interface. The left sidebar lists project files: bin, config, migrations, public (images folder containing DODO.png, gouillat1.png, REUNION.png, vue-du-ciel1.png), index.php, src, and Controller (.gitignore). The main area displays the content of the file `home.html.twig`:

```
home.html.twig — SymfonyProject
templates > pages > home.html.twig > div.example-wrapper
1  {% extends 'base.html.twig' %}
2
3  {% block title 'Accueil | ' ~ parent() %}
4
5  {% block body %}
6
7
8  <div class="example-wrapper">
9    <h1>{{title}}</h1>
10
11    {% if age > 18 %}
12      <p>Tu es majeur</p>
13    {% else %}
14      <p>Tu es mineur</p>
15    {% endif %}
16
17    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Natus, quia fugit odit vita
18    nihil magni doloribus laborum laboriosam officiis illo debitis! Magnam.</p>
19
20  </div>
```

Figure 71 : Fonctionnalité appelle d'une structure de contrôle avec les accolades pourcentage

DOSSIER PROFESSIONNEL (DP)



```
BDController.php
index.html.twig
home.html.twig

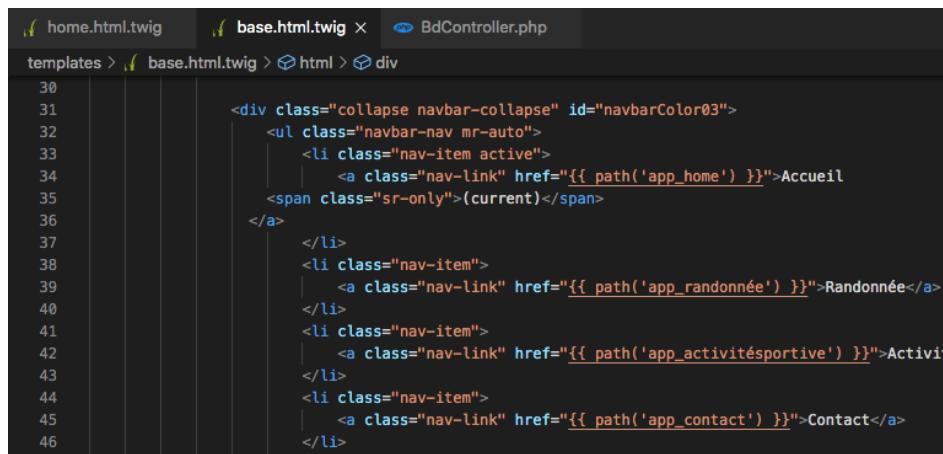
25     /**
26      * @Route("/", name="home")
27      */
28     public function home()
29     {
30         return $this->render('bd/home.html.twig', [
31             'title' => "Bienvenue !",
32             'age' => 30
33         ]);
34     }
35
36 }
```

Figure 72 : Le controller qui contient la route pour traiter le code de la page .Twig et le renvoyer sur la page home du navigateur



Figure 73 : Résultat de la fonctionnalité de l'exemple ci-dessus

Il existe aussi la fonction **Path**, twig prend 2 paramètres : 1 obligatoire le nom de la route et 1 optionnel. J'utilise cette fonction pour créer un lien de cette manière : {{ path() }}.



```
home.html.twig
base.html.twig
BdController.php

templates > base.html.twig > html > div

30     <div class="collapse navbar-collapse" id="navbarColor03">
31         <ul class="navbar-nav mr-auto">
32             <li class="nav-item active">
33                 <a class="nav-link" href="{{ path('app_home') }}>Accueil
34                 <span class="sr-only">(current)</span>
35             </a>
36         </li>
37         <li class="nav-item">
38             <a class="nav-link" href="{{ path('app_randonnée') }}>Randonnée</a>
39         </li>
40         <li class="nav-item">
41             <a class="nav-link" href="{{ path('app_activitésportive') }}>Activité
42         </li>
43         <li class="nav-item">
44             <a class="nav-link" href="{{ path('app_contact') }}>Contact</a>
45         </li>
46     </ul>
```

Figure 74 : Fonctionnalité Path

Accueil Activités Culture Boutique

DOSSIER PROFESSIONNEL (DP)

Figure 75 : Résultat sur le navigateur de la fonctionnalité Path

Symfony utilise également **Doctrine**, c'est un **ORM**: (*Object Relational Mapping*).

L'ORM est une classe qui permet de manipuler les tables d'une base de données comme des objets (*Entities*) et leurs relations.

Doctrine ajoute des fonctionnalités à **PDO**. (*PHP Data Objects*).

L'ORM de Symfony : Doctrine

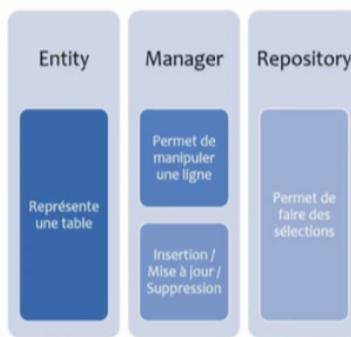


Figure 76 : L'ORM de Symfony

Je poursuis avec Les **migrations** dans **Symfony** qui permettent de faire passer une base de données d'un état A à un état B.

Les migrations dans Symfony

- Migration #1**
 - Je créé 2 tables
- Migration #2**
 - Je modifie les champs d'une table
 - J'en créé une autre
 - J'en supprime une autre
- Migration #3**
 - Je supprime un champ d'une table
 - J'ajoute une relation entre deux tables

Figure 77 : Les migrations

On peut parler aussi des **fixtures** ou données de tests, qui permettent au développeur d'initialiser les données de la base de données dans un état connu.

DOSSIER PROFESSIONNEL (DP)

Les Fixtures

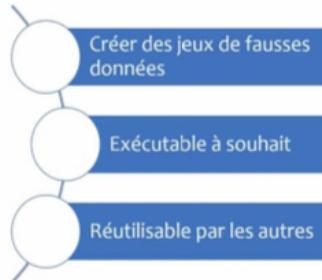


Figure 78 : Les fixtures

Pour relier ma base de donnée à mon projet **Symfony** avec l'**ORM Doctrine**, j'ouvre le fichier **.env** comme dans l'exemple ci-dessous et je modifie la ligne 31 en y ajoutant le nom de ma base de donnée et en vérifiant le port du serveur MySql qui est le **8889**.

```
env - SymfonyProject
explorateur
ÉDITEURS OUVERTS
SYMPONYPROJECT
> contact
> order
> order_cancel
> order_success
pages
> about.html.twig
> activitésportive.html.twig
> biodiversité.html.twig
> histoire.html.twig
> home.html.twig
> plage.html.twig
> plats.html.twig
> randonnée.html.twig
> visite.html.twig
product
> security
base.html.twig
flash_messages.html.twig
tests
translations
var
vendor
.env
.env.test
.gitattributes
.gitignore
composer.json
composer.lock
phpunit.xml.dist
README.md
symfony.lock
STRUCTURE
CHRONOLOGIE
main* 0 0 0 Live Share
Li 31, Col 1 (72 sélectionné) Espaces : 4 UTF-8 LF Texte brut ⌂ Go Live ⌂ ⌂
```

```
# _ENV_ = &ENV|_ENV_CLOUD|COMMITTED_ENVIRONMENT_SPECIFIC_OVERRIDE
#
# Real environment variables win over .env files.
#
# DO NOT DEFINE PRODUCTION SECRETS IN THIS FILE NOR IN ANY OTHER COMMITTED FILES.
#
# Run "composer dump-env prod" to compile .env files for production use (requires symfony/flex >=1.2).
# https://symfony.com/doc/current/best_practices.html#use-environment-variables-for-infrastructure-configuration
#
##> symfony/framework-bundle ##
# MAILER_DSN=smtplib://localhost
MAILER_DSN=smtplib://127.0.0.1:1025
##< symfony/mailer ##
#
##> doctrine/doctrine-bundle ##
# Format described at https://www.doctrine-project.org/projects/doctrine-dbal/en/latest/reference/configuration.html#connecting-using-a-url
# IMPORTANT: You MUST configure your server version, either here or in config/packages/doctrine.yaml
#
# DATABASE_URL="sqlite:///kernel.project_dir/var/data.db"
DATABASE_URL="mysql://root@127.0.0.1:8889/bd_digserverVersion=5.7"
# DATABASE_URL="postgresql://db_user:db_password@127.0.0.1:5432/db_name?serverVersion=13&charset=utf8"
##< doctrine/doctrine-bundle ##
#
##> nelmio/cors-bundle ##
CORS_ALLOW_ORIGIN='^https://(localhost|127.0.0.1)(:[0-9]+)$'
##< nelmio/cors-bundle ##
```

Figure 79 : Modification du fichier .env

Je peux désormais créer des **entités** avec la commande :(*php bin/console make :entity.*)

DOSSIER PROFESSIONNEL (DP)

```
A-C-N:SymfonyProject a_c_n$ php bin/console make:entity
Class name of the entity to create or update (e.g. FierceKangaroo):
> Users

created: src/Entity/Users.php
created: src/Repository/UsersRepository.php

Entity generated! Now let's add some fields!
You can always add more fields later manually or by re-running this command.

New property name (press <return> to stop adding fields):
> []
```

Figure 80 : Création d'une class (Users) qui représente les utilisateurs du site

La ligne de commande a créé deux nouveaux fichiers une **entité Users** (qui représente la table des Utilisateurs inscrit sur le site) et un **repository UsersRepository** (qui permet de faire des sélections sur les données de la table des utilisateurs).

On continue avec un exemple de relation entre deux entités (Produits et Auteur) avec la commande : (*php bin/console make :entity Produit.*)

```
A-C-N:SymfonyProject a_c_n$ php bin/console make:entity Produit
Your entity already exists! So let's add some new fields!

New property name (press <return> to stop adding fields):
> auteur

Field type (enter ? to see all types) [string]:
> relation
```

What class should this entity be related to?:
> Auteur

What type of relationship is this?

Type	Description
ManyToOne	Each Produit relates to (has) one Auteur. Each Auteur can relate to (can have) many Produit objects
OneToMany	Each Produit can relate to (can have) many Auteur objects. Each Auteur relates to (has) one Produit
ManyToMany	Each Produit can relate to (can have) many Auteur objects. Each Auteur can also relate to (can also have) many Produit objects
OneToOne	Each Produit relates to (has) exactly one Auteur. Each Auteur also relates to (has) exactly one Produit.

DOSSIER PROFESSIONNEL (DP)

The screenshot shows a terminal window with a file tree on the left and command output on the right. The file tree includes .gitignore, AuteurReposit..., ProduitReposit..., Kernel.php, templates (containing pages like about.html.twig, activitesportive.html.twig, contact.html.twig, home.html.twig, randonnee.html.twig, base.html.twig), tests, translations, var, vendor (.env, .env.test), .gitattributes, .gitignore, composer.json, composer.lock, and phunit.xml.dist. The terminal output shows the creation of a relationship between Auteur and Produit:

```
Relation type? [ManyToOne, OneToMany, ManyToMany, OneToOne]: > ManyToOne
Is the Produit.auteur property allowed to be null (nullable)? (yes/no) [yes]: > no
Do you want to add a new property to Auteur so that you can access/update Produit objects from it - e.g. $auteur->getProduits(): > no
A new property will also be added to the Auteur class so that you can access the related Produit objects from it.
New field name inside Auteur [produits]: >
Do you want to activate orphanRemoval on your relationship?
A Produit is "orphaned" when it is removed from its related Auteur.
e.g. $auteur->removeProduit($produit)
NOTE: If a Produit may *change* from one Auteur to another, answer "no".
Do you want to automatically delete orphaned App\Entity\Produit objects (orphanRemoval)? (yes/no) [no]: > yes
updated: src\Entity\Produit.php
updated: src\Entity\Auteur.php
Add another property? Enter the property name (or press <return> to stop adding fields): >
Success!
```

Figure 81 : Création des relations entre entités

Ensuite on effectue une migration vers la base données avec la commande : (*php bin/console make:migration*), qui permet de créer une migration SQL à partir des entités présentes. Et pour faire une mise à jour on se sert de la commande *php bin/console doctrine:migrations:migrate*, qui permet de lancer les scripts de migrations afin de mettre à jour la base de données.

The screenshot shows a terminal window with the following output:

```
Next: When you're ready, create a migration with php bin/console make:migration
A-C-N:SymfonyProject a_c_n$ php bin/console make:migration
Success!

Next: Review the new migration "migrations/Version20210302131936.php"
Then: Run the migration with php bin/console doctrine:migrations:migrate
See https://symfony.com/doc/current/bundles/DoctrineMigrationsBundle/index.html
A-C-N:SymfonyProject a_c_n$
```

Figure 82 : Les migrations

Ensute je vérifie si la requête est celle que je souhaite bien exécuter pour modifier la base de données et je valide (yes).

2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisé pour cette septième compétence :

Documentation: Site de Symfony.

Logiciel: Visual studio code, Mamp, Mysql, Composer.

Site web: Github desktop.

Framework: Symfony5.

Application: PhpMyAdmin.

DOSSIER PROFESSIONNEL (DP)

Langage: *Html5, css3, Php, Sql, Twig.*

3. Avec qui avez-vous travaillé ?

Pour les Compétences de 1 à 8 j'ai travaillé seul en me documentant sur internet

4. Contexte

Nom de l'entreprise, organisme ou association ▶ *Formation AFPA*

Chantier, atelier, service ▶ *Cliquez ici pour taper du texte.*

Période d'exercice ▶ Du : *05/10/2020* au : *11/06/2021*

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 2

Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité.

Compétence n°8 - Elaborer et mettre en œuvre des composants dans une application de gestion de contenu ou e-commerce

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Pour cette huitième compétence j'ai étudié la création d'un **plugin WordPress** pour créer un **livre d'Or**.

Les **extensions** ou **plugins** en anglais sont des fonctionnalités que l'on ajoute à un site WordPress pour réaliser une application web.

On créer un **plugin** pour plusieurs raisons : pour alléger le functions.php du **thème enfant**, pour faire suivre le code, réutiliser le code et également pour prévenir les conflits et incompatibilités.

Je débute par l'installation de **WordPress**.

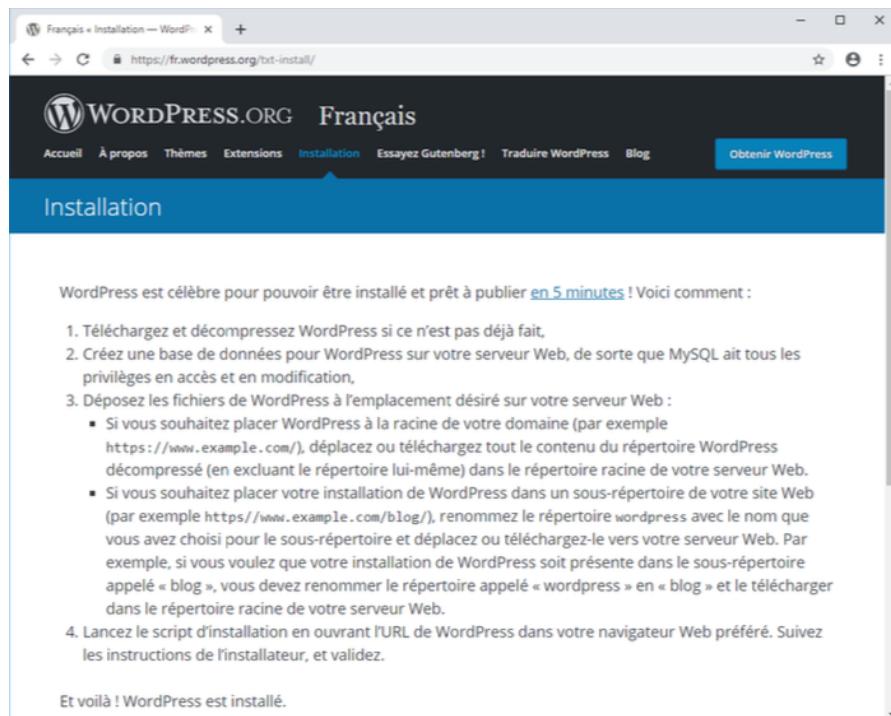


Figure 83 : Page de la documentation de l'installation de WordPress

A la suite je créer la **base de données**.

DOSSIER PROFESSIONNEL (DP)

Table	Action	Rows	Type	Collation	Size	Overhead
wp_commentmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_520_ci	48.0 Kib	-
wp_comments	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_520_ci	96.0 Kib	-
wp_links	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_520_ci	32.0 Kib	-
wp_options	Browse Structure Search Insert Empty Drop	191	InnoDB	utf8mb4_unicode_520_ci	1.1 Mib	-
wp_postmeta	Browse Structure Search Insert Empty Drop	218	InnoDB	utf8mb4_unicode_520_ci	80.0 Kib	-
wp_posts	Browse Structure Search Insert Empty Drop	94	InnoDB	utf8mb4_unicode_520_ci	224.0 Kib	-
wp_termmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_520_ci	48.0 Kib	-
wp_terms	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_unicode_520_ci	48.0 Kib	-
wp_term_relationships	Browse Structure Search Insert Empty Drop	9	InnoDB	utf8mb4_unicode_520_ci	32.0 Kib	-
wp_term_taxonomy	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_unicode_520_ci	48.0 Kib	-
wp_usermeta	Browse Structure Search Insert Empty Drop	23	InnoDB	utf8mb4_unicode_520_ci	48.0 Kib	-
wp_users	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_520_ci	64.0 Kib	-
12 tables	Sum	543	InnoDB	utf8_general_ci	1.8 Mib	0 B

Figure 84 : Base de données pour WordPress

Je place le dossier **WordPress1** dans le dossier (**htdocs**).

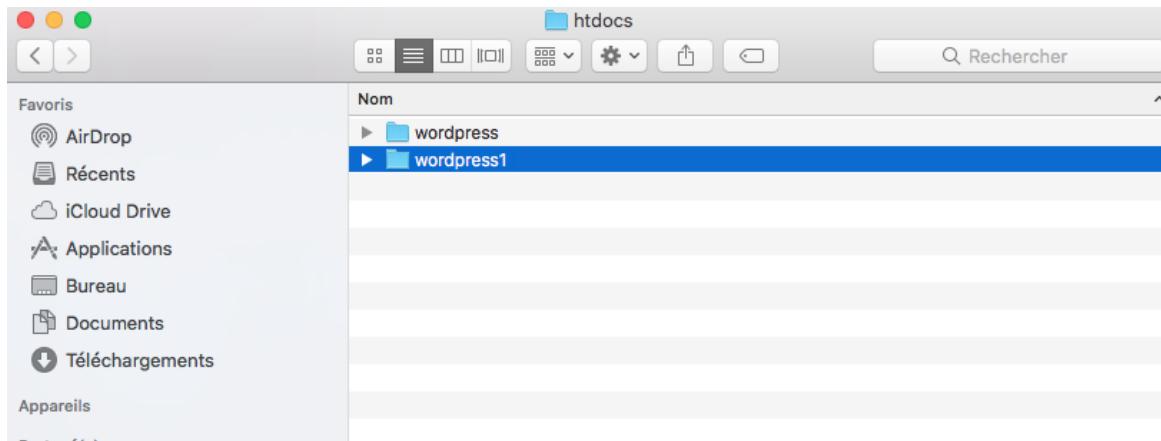


Figure 85 : Le Dossier du projet WordPress

Je lance le script d'installation en ouvrant l'URL de **WordPress** dans mon navigateur Web.

← → ⌛ ⓘ localhost:8888/wordpress1/wp-admin/index.php

DOSSIER PROFESSIONNEL (DP)

Figure 86 : Url de WordPress

The screenshot shows the WordPress dashboard. At the top, there's a header with a back arrow, forward arrow, refresh button, and a URL bar showing "localhost:8888/wordpress1/wp-admin/index.php". The dashboard title is "Tableau de bord". On the left, a sidebar lists "Accueil" (with 1 update), "Articles", "Products", "Comic books", "Médias", "Pages", "Commentaires", "Testimonials", "Apparence", "Extensions" (with 1 update), "Comptes", "Outils", "Règles", "Alecaddd", and "Réduire le menu". The main content area has sections for "Bienvenue sur WordPress!", "Lancez-vous!", "Étapes suivantes", "Plus d'actions", "État de santé du site", "D'un coup d'œil", "Brouillon rapide", and "Événements et nouveautés WordPress".

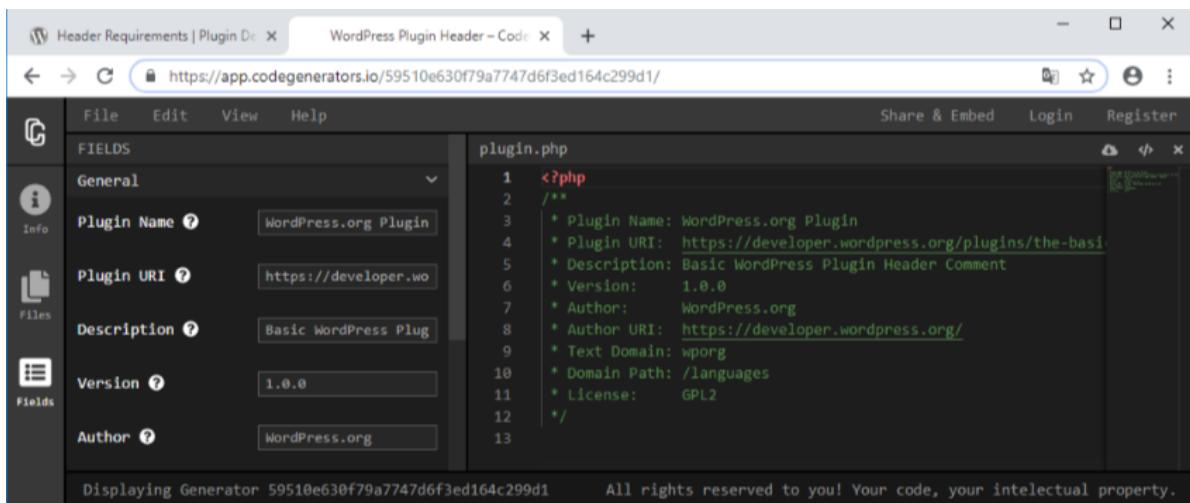
Figure 87 : Tableau de bord de WordPress

```
wp-config-sample.php
1 * For developers: WordPress debugging mode.
2 *
3 * Change this to true to enable the display of notices during development.
4 * It is strongly recommended that plugin and theme developers use WP_DEBUG
5 * in their development environments.
6 *
7 * For information on other constants that can be used for debugging,
8 * visit the Codex.
9 *
10 * @link https://codex.wordpress.org/Debugging_in_WordPress
11 */
12 define( 'WP_DEBUG', true );
```

Figure 88 : Script de l'installation généré par WordPress

Je génère le fichier **plugin.php**.

DOSSIER PROFESSIONNEL (DP)



The screenshot shows a web application interface for generating a WordPress plugin header. On the left, there are tabs for 'Info', 'Files', and 'Fields'. Under 'Fields', there are sections for 'General' (Plugin Name: WordPress.org Plugin, Plugin URI: https://developer.wo, Description: Basic WordPress Plug, Version: 1.0.0, Author: WordPress.org), 'Info' (Info icon), 'Files' (File icon), and 'Fields' (List icon). The main area displays the generated plugin.php code:

```
<?php
/*
 * Plugin Name: WordPress.org Plugin
 * Plugin URI: https://developer.wordpress.org/plugins/the-basi
 * Description: Basic WordPress Plugin Header Comment
 * Version: 1.0.0
 * Author: WordPress.org
 * Author URI: https://developer.wordpress.org/
 * Text Domain: wporg
 * Domain Path: /languages
 * License: GPL2
 */
```

At the bottom, it says "Displaying Generator 59510e630f79a7747d6f3ed164c299d1" and "All rights reserved to you! Your code, your intellectual property."

Figure 89 : Outil pour générer le fichier plugin.php

Je crée un nouveau fichier que je nomme **alecaddd-plugin.php** qui correspond à une fonctionnalité de l'application.

J'enregistre mon fichier et je le range dans un dossier que je nomme par exemple **mon-premier- plugin** qui lui-même se trouve dans le dossier **plugins**.

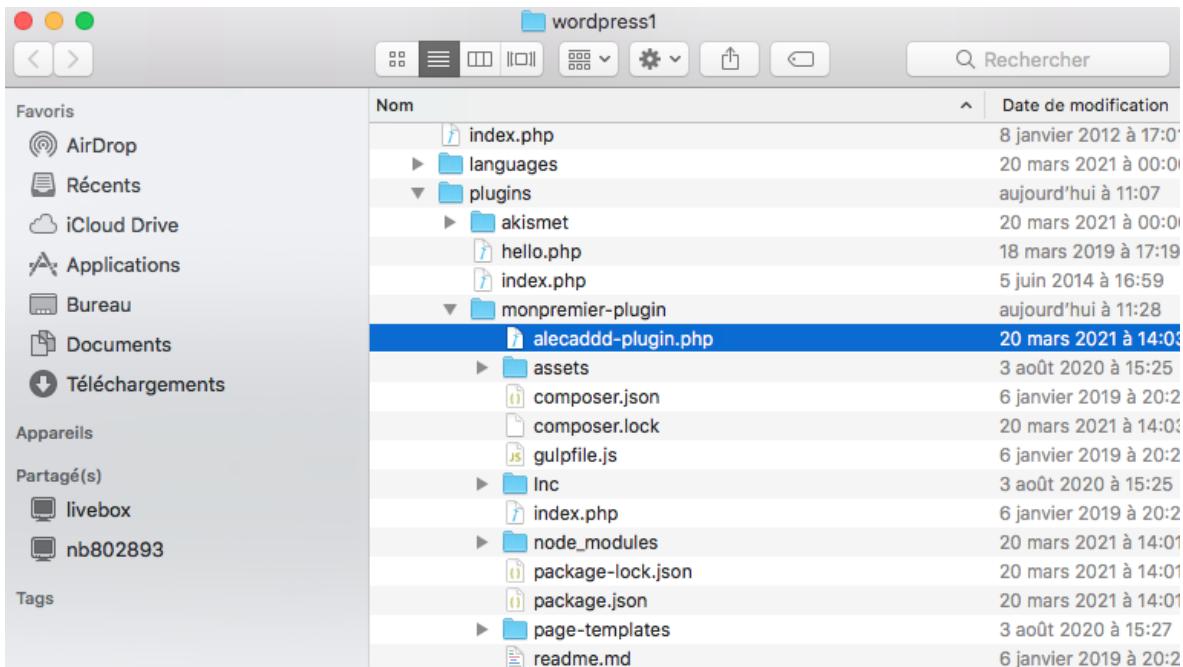


Figure 90 : Fichier alecaddd-plugin.php

Ensuite Je teste mon plugin et je vérifie s'il fonctionne bien une fois activé depuis le **back-office** de mon application **WordPress** en développement.

DOSSIER PROFESSIONNEL (DP)

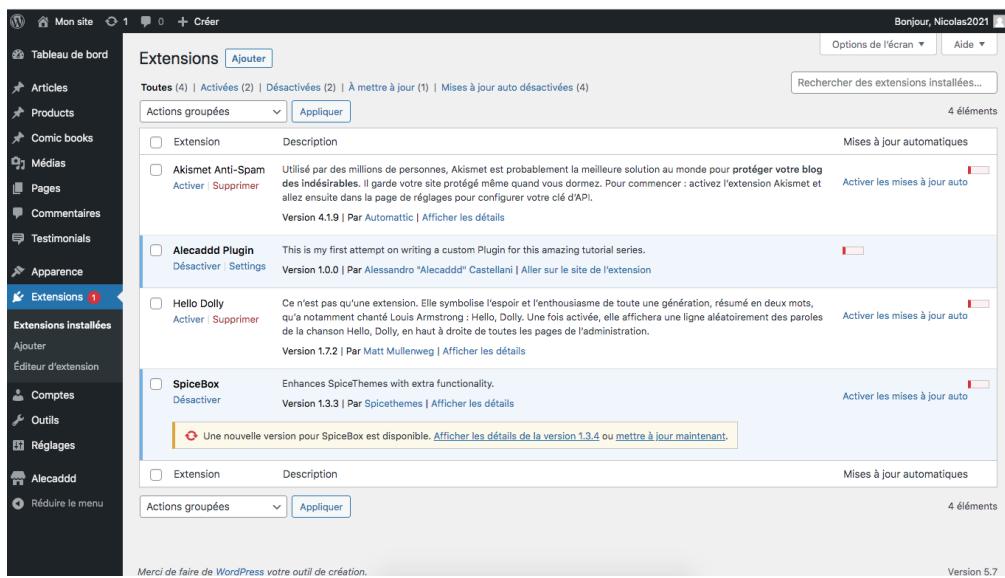
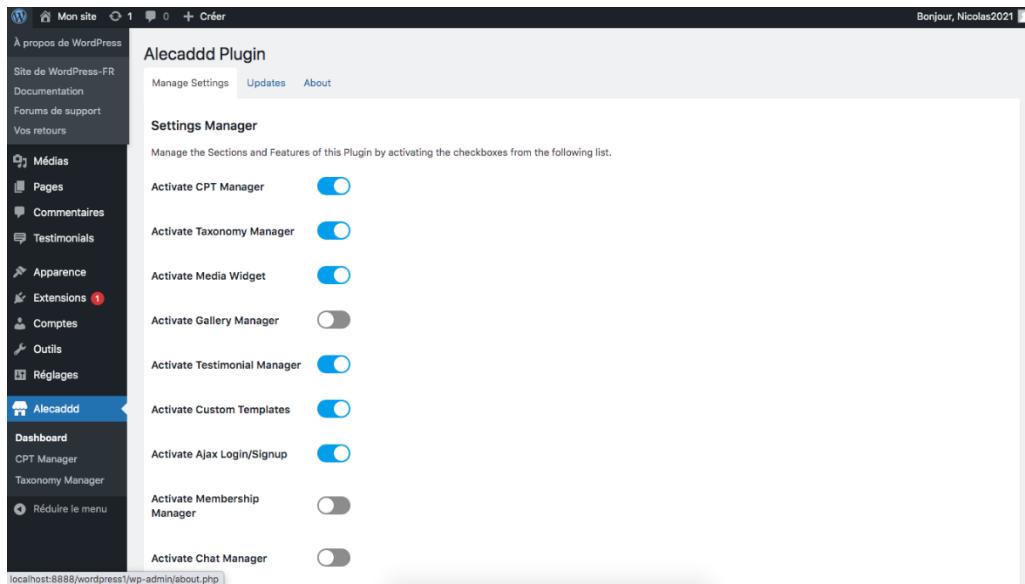


Figure 91 : Activation du plugin Alecaddd Plugin

Par la suite Je peux envoyer mon **plugin** dans mon application **WordPress** en production soit via **FTP**, soit via le **back-office** mais dans ce cas il doit être compressé.

A la suite je commence la configuration du plugin selon mes besoin pour mon projet.

Je créer les **Custom Post Types** qui vont me permettre de créer des types de contenus sur-mesure associant ou non une taxonomie particulière.



DOSSIER PROFESSIONNEL (DP)

The image consists of two vertically stacked screenshots of the CPT Manager plugin's configuration interface within a WordPress dashboard.

Screenshot 1: Adding a new Custom Post Type

This screenshot shows the "Add Custom Post Type" tab of the CPT Manager settings. The "Custom Post Type ID" field contains "product", "Singular Name" is "Product", and "Plural Name" is "Products". The "Public" and "Archive" options are turned off. A blue "Enregistrer les modifications" button is at the bottom.

ID	Singular Name	Plural Name	Public	Archive	Actions
product	Product	Products	TRUE	FALSE	Edit Delete
comicbook	Comic Book	Comic books	FALSE	TRUE	Edit Delete

Screenshot 2: Managing existing Custom Post Types

This screenshot shows the "Manage Your Custom Post Types" section. It lists the two custom post types created: "product" (singular "Product", plural "Products") and "comicbook" (singular "Comic Book", plural "Comic books"). Each row has "Edit" and "Delete" buttons in the "Actions" column.

Figure 92 : Configuration du Plugin

Maintenant que mon **plugin** est configuré comme je le souhaite je peux y ajouter les articles que je souhaite voici un exemple :

DOSSIER PROFESSIONNEL (DP)

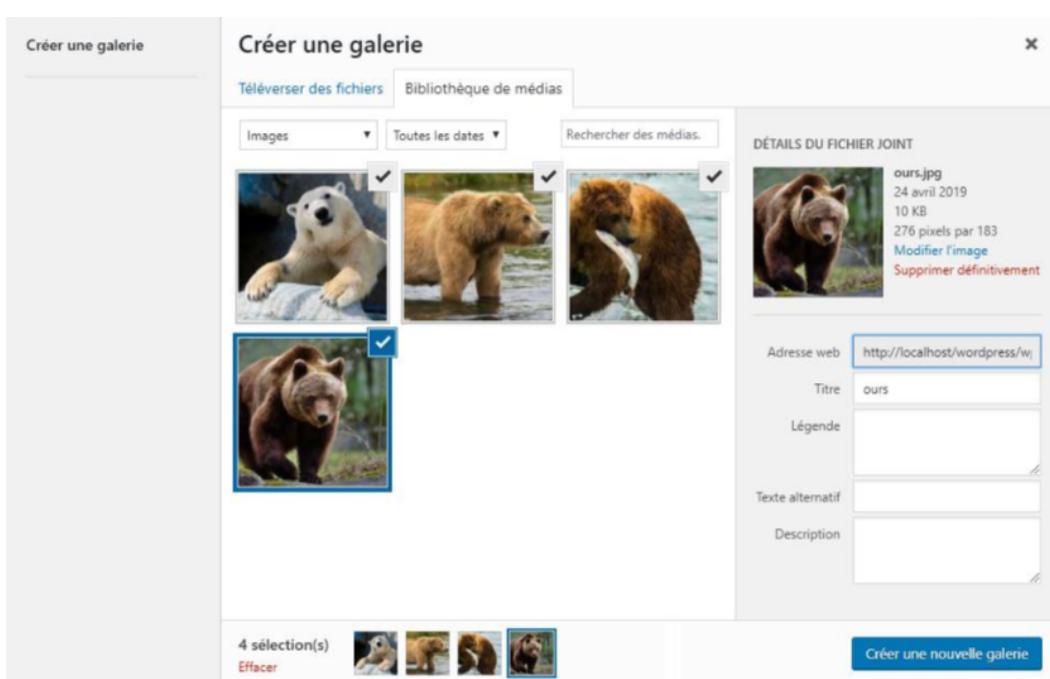


Figure 93 : Ajout des articles

Et je configure ma fonction **Testimonials** en y intégrant un **article Média**.

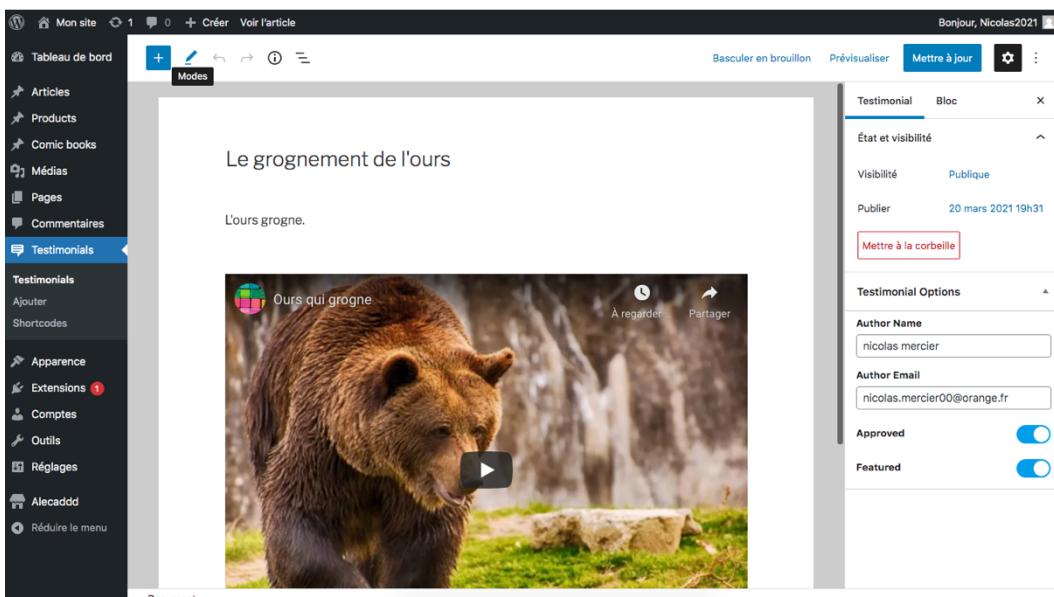


Figure 94: Ajout d'un article média

Ainsi qu'un article pour poster un témoignage à partir de mon **livre d'or**.

DOSSIER PROFESSIONNEL (DP)

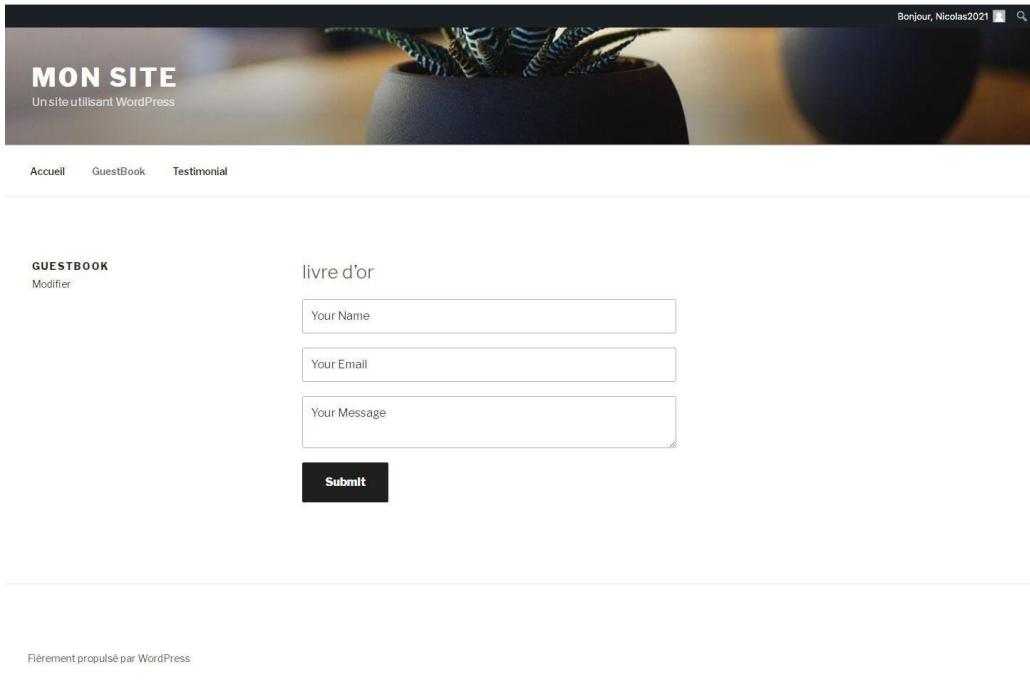


Figure 95 : Livre d'or

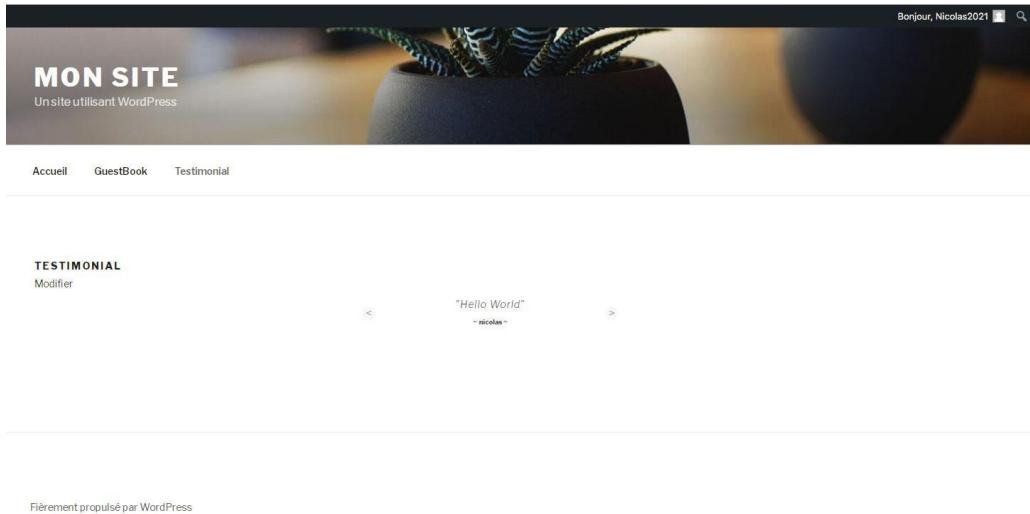


Figure 96: Poste d'un témoignage

Le résultat de ma page d'accueil de mon projet **Plugin WordPress**.

DOSSIER PROFESSIONNEL (DP)

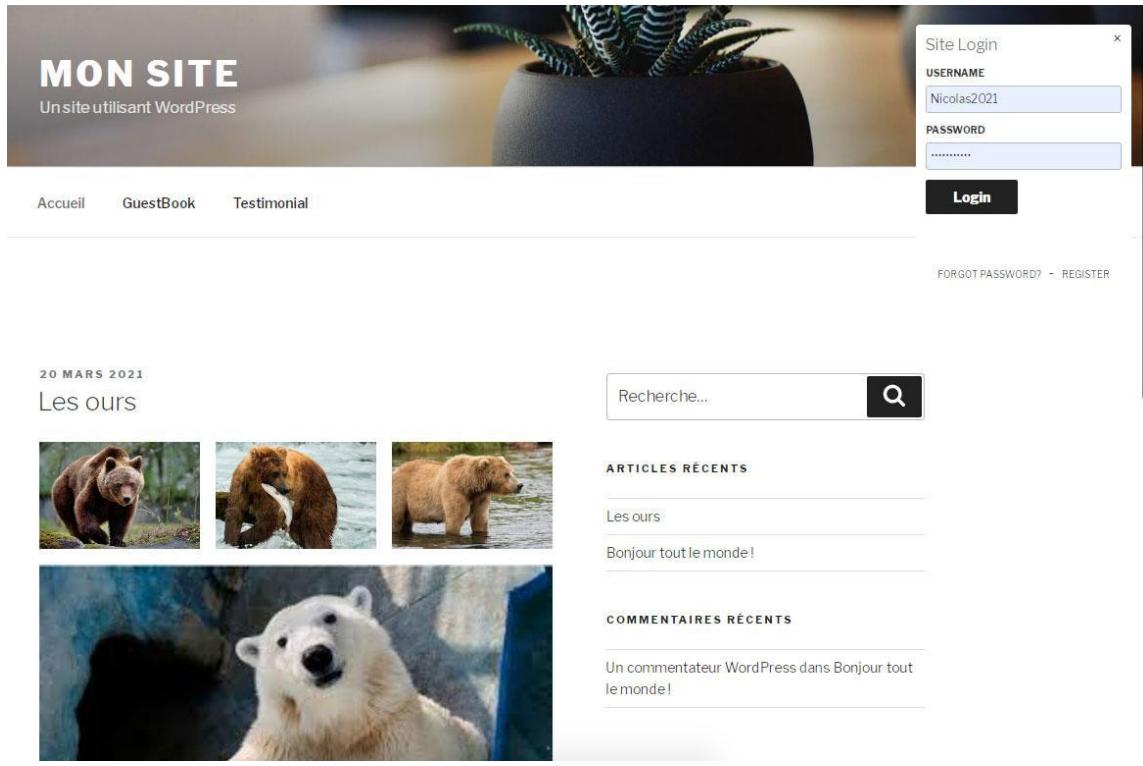


Figure 97 : Page d'accueil WordPress

2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisé pour cette huitième compétence :

Documentation: Site WordPress.

Logiciel: Visual studio code, Mamp, Mysql.

Site web: Github desktop.

Cms: WordPress.

Application: PhpMyAdmin.

Langage: Html5, Css3, Php.

3. Avec qui avez-vous travaillé ?

Pour les Compétences de 1 à 8 j'ai travaillé seul en me documentant sur internet

4. Contexte

DOSSIER PROFESSIONNEL (DP)

Nom de l'entreprise, organisme ou association ▶ *Formation AFPA*

Chantier, atelier, service ▶ *Cliquez ici pour taper du texte.*

Période d'exercice ▶ Du : *05/10/2020* au : *11/06/2021*

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Activité-type 3 Cliquez ici pour entrer l'intitulé de l'activité

Exemple n° 1 ▶ Cliquez ici pour entrer l'intitulé de l'exemple

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

2. Précisez les moyens utilisés :

3. Avec qui avez-vous travaillé ?

4. Contexte

Nom de l'entreprise, organisme ou association ▶ *Cliquez ici pour taper du texte.*

Chantier, atelier, service ▶ *Cliquez ici pour taper du texte.*

Période d'exercice ▶ Du : *Cliquez ici* au : *Cliquez ici*

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Titres, diplômes, CQP, attestations de formation

(facultatif)

Intitulé	Autorité ou organisme	Date
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.

DOSSIER PROFESSIONNEL (DP)

Déclaration sur l'honneur

Je soussigné(e) [prénom et nom] **Nicolas Mercier**,

déclare sur l'honneur que les renseignements fournis dans ce dossier sont exacts et que je suis l'auteur(e) des réalisations jointes.

Fait à Limeil-Brévannes

le 11/06/2021

Pour faire valoir ce que de droit.

Signature :

DOSSIER PROFESSIONNEL^(DP)

Documents illustrant la pratique professionnelle

(facultatif)

Intitulé

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

ANNEXES

(Si le RC le prévoit)