

# DOSSIER PROFESSIONNEL (DP)

<i>Nom de naissance</i>	▶ Cimino
<i>Nom d'usage</i>	▶ Cimino
<i>Prénom</i>	▶ Thomas
<i>Adresse</i>	▶ 11 lotissement le Piache Sud 26390 Hauterives

## Titre professionnel visé

*Graduate Developer Web Full Stack*

### MODALITE D'ACCES :

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

## Sommaire

### Exemples de pratique professionnelle

<b>Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité</b>	<b>p.</b>	<b>4</b>
▶ Mon portfolio :	p.	4
• Réaliser une interface utilisateur web statique est adaptable		
▶ Carrousel d'images (Mon portfolio) :	p.	7
• Développer une interface utilisateur web dynamique		
▶ Projet d'ECF (Quai Antique) :	p.	12
• Maquetter une application		
• Réaliser une interface utilisateur avec une solution de gestion de contenu ou e-commerce		
<b>Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité</b>	<b>p.</b>	<b>17</b>
▶ Conception d'une base de données (Quai antique) :	p.	17
• Créer une base de données		
▶ Gestion des données du projet (Quai antique) :	p.	22
• Développer les composants d'accès aux données		
• Développer la partie back-end d'une application web ou web mobile		
▶ Gestionnaire d'images (Quai antique) :	p.	29
• Elaborer et mettre en œuvre des composants dans une application de gestion de contenu ou e-commerce		
<b>Titres, diplômes, CQP, attestations de formation</b> <i>(facultatif)</i>	<b>p.</b>	<b>38</b>
<b>Déclaration sur l'honneur</b>	<b>p.</b>	<b>39</b>
<b>Documents illustrant la pratique professionnelle</b> <i>(facultatif)</i>	<b>p.</b>	<b>40</b>
<b>Annexes</b> <i>(Si le RC le prévoit)</i>	<b>p.</b>	<b>41</b>

# **EXEMPLES DE PRATIQUE PROFESSIONNELLE**

## Activité-type 1

### Développer la partie front-end d'une application web ou web mobile

Exemple n°1 ► Mon Portfolio

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

Afin de mettre en pratique les compétences front end apprises au cours de ma formation (langages HTML, CSS ainsi que les framework CSS) et pour donner de la visibilité à mes réalisations, qui me serviront pour mes recherches d'emplois, j'ai développé mon portfolio.

Dans un premier temps, j'ai cherché des exemples sur le web pour voir les réalisations que d'autres personnes avaient réalisé, pour me donner de l'inspiration et voir quelles seraient les informations utiles qui me permettraient de mettre en valeur les compétences acquises.

Après cette recherche, j'ai commencé à créer les pages de mon site en les commentant avec des éléments que je souhaitais montrer aux visiteurs. J'ai donc fait des recherches pour trouver le framework CSS qui me conviendrait le mieux, tout en gardant à l'esprit que mon site devait être responsive (s'adapter à la taille de l'écran sur lequel il sera vu).

J'ai choisi le framework Tailwind pour le découvrir, l'ayant survolé lors de la formation, car il a de nombreux composants, possibilités et classes de style responsive.

**Pour le rendu final : voir annexes 1 page 41**

#### 2. Précisez les moyens utilisés :

Concernant la partie développement, j'ai utilisé l'éditeur de code Visual Studio Code.

J'ai installé le framework Tailwind CSS, grâce à la documentation et des lignes de commandes node.js (environnement bas niveau permettant l'exécution de JavaScript coté serveur) et son outil « npm », dans le terminal de mon projet.

```
npm install -D tailwindcss  
npx tailwindcss init
```

Dans mon fichier `tailwind.config.js`, j'ajoute le chemin d'accès à tous mes fichiers de modèles et je défini le thème sombre par défaut.

```
/** @type {import('tailwindcss').Config} */
module.exports = {
  content: [
    "/*.{html,js,ts,php}", "./layout/*.{html,js,ts,php}"
  ],
  theme: {
    extend: {},
  },
  plugins: [],
  darkMode: 'class',
}
```

J'ajoute ensuite les directives '@tailwind' dans mon fichier CSS principal.

```
@tailwind base;
@tailwind components;
@tailwind utilities;
```

J'exécute ensuite l'outil CLI pour qu'il analyse les classes des fichiers de mon projet, et qu'il puisse créer le CSS dans un fichier.

```
npx tailwindcss -i ./src/input.css -o ./dist/output.css --watch
```

Je peux ensuite commencer à utiliser tailwind dans mon projet, en ajoutant dans les balises `<head>`, mon fichier CSS compilé, en utilisant les classes utilitaires pour le styliser.

```
<link href="./dist/output.css" rel="stylesheet">
```

# DOSSIER PROFESSIONNEL (DP)

L'import de ce framework me permet donc de rendre responsive rapidement mon site, ainsi que de gérer le thème de couleur, à l'aide de classes utilitaires et de quelques balises. Par exemple les classes préfixées par « **md:** » dans l'attribut de la balise <nav>.

```
<body class="bg-white dark:bg-black dark:text-white text font-['Audiowide'] min-h-screen">
<header class="border-b border-gray-500 dark:border-white rounded-b-3xl shadow-sm shadow-black dark:shadow-white">
  <nav class="container mx-auto md:px-16">
    <div class="flex flex-wrap justify-between items-center lg:py-2 p-4 mx-auto">
      <div class="flex items-center ">
        <div class="min-w-[64px]">
          
        </div>
        <div class="ml-2 text-base font-semibold uppercase mr-4">
          <h3 class=""><a class="hover:text-gray-400" href="."/>CIMINO
Thomas</a></h3>
          <h3 class="">Développeur Web Full Stack</h3>
        </div>
      </div>
    </div>
  </nav>
</header>
```

Extrait du fichier header.php

## 3. Avec qui avez-vous travaillé ?

J'ai travaillé seul sur ce projet et j'ai pu échanger sur des communautés de développeurs pour améliorer mon portfolio.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► *Studi*

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

Période d'exercice ► Du : **01/01/2023** au : **01/02/2023**

## 5. Informations complémentaires (facultatif)

## Activité-type 1

## Développer la partie front-end d'une application web ou web mobile

Exemple n° 2 ► Carrousel d'images (Mon portfolio)

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

Lors de la conception de mon portfolio, j'ai eu l'envie de faire un carrousel d'images, qui afficherait et mettrait en valeur les compétences acquises lors de ma formation, pour pouvoir présenter un joli effet lors de potentiels entretiens.

J'ai utilisé HTML, JavaScript ainsi que le framework Tailwind CSS pour la mise en forme et la partie responsive du site.

Pour ceci, le carrousel devait contenir :

- Des boutons pour changer les images
- Un bloc parent
- Des titres pour les blocs enfants

### 2. Précisez les moyens utilisés :

Concernant le développement, j'ai utilisé l'éditeur de code Visual Studio Code, le framework Tailwind CSS, ainsi que les langages HTML et JavaScript.

Pour développer ceci, j'ai mis en forme mon bloc parent grâce au framework Tailwind et je lui ai attribué un id pour pouvoir récupérer sa taille et le sélectionner. J'ai ajouté des boutons pour que l'utilisateur puisse changer d'images. Il m'a fallu également mettre en forme mes éléments enfants avec leurs images.

Le bloc <div> avec l'id « carrousel\_panorama » sera notre carrousel, c'est dans ce bloc que j'ai mis les éléments enfants qui contiennent les images de mes compétences acquises lors de la formation.

Le bloc <div> avec l'id « nav-indicator » et l'id « nav-carrousel » contiennent eux les boutons qui nous permettront de faire bouger les blocs enfants de notre carrousel.

```
<div id="carrousel" class="relative mb-16 mx-1 sm:mx-8 md:mx-32 1
  <!-- Carousel wrapper -->
  <div id="carrousel_panorama" class="relative h-full rounded-l
    <!-- Item 1 -->
    <div id="carrousel-item-1" class="relative h-full grid ju
    </div>
    <!-- Item 2 -->
    <div id="carrousel-item-2" class="absolute h-full w-full
    </div>
    <!-- Item 3 -->
    <div id="carrousel-item-3" class="absolute h-full w-full
    </div>
    <!-- Item 4 -->
    <div id="carrousel-item-4" class="absolute h-full w-full
    </div>
  </div>
  <!-- Slider indicators -->
  <div id="nav-indicator" class="absolute z-30 flex space-x-3 -
  </div>

  <!-- Slider controls -->
  <div id="nav-carrousel">...
  </div>
</div>
```

Mon fichier JavaScript sera dans un fichier à part, donc on l'insérera comme ceci.

```
<script src="./carrousel.js"></script>
```

Grâce à JavaScript, je vais pouvoir écouter les événements lorsque l'utilisateur agira sur les boutons, et faire en sorte que mon carrousel réagisse selon ses choix.



```
// Panorama
const panorama = document.getElementById("carrousel_panorama"),
    panoramawidth = panorama.offsetWidth;

// Fleches de contrôle
const prevButton = document.getElementById("carrousel-prev"),
    nextButton = document.getElementById("carrousel-next");

// Container des ronds de contrôles
const navIndicatorsContainer = document.getElementById("nav-indicator");
```

Une fois tout ceci récupéré dans mon fichier JavaScript, il faudra attribuer une position par défaut à chacun de mes éléments enfants.

```
// Attribution des positions par défaut
function setDefaultPosition() {
    for (let i = 0; i < panorama.childElementCount; i++) {
        panorama.children[i].style = `left: ${panoramawidth * i}px`;
    }
    panorama.setAttribute("style", `left: 0px`);
}
```

Je n'ai plus qu'à ajouter un écouteur d'événements sur mes boutons, pour que le click de l'utilisateur fasse réagir mon carrousel et que mes images se déplacent.

```
// Flèche Next
nextButton.addEventListener("click", () => {
  if (
    Math.abs(leftShift) >=
    panoramaWidth * (panorama.childElementCount - 1)
  ) {
    leftShift = 0;
    panorama.setAttribute("style", `left: ${leftShift}px`);
  } else {
    leftShift -= panoramaWidth;
    panorama.setAttribute("style", `left: ${leftShift}px`);
  }
});

// Flèche Prev
prevButton.addEventListener("click", () => {
  if (leftShift >= 0) {
    leftShift = -Math.abs(panoramaWidth * (panorama.childElementCount - 1));
    panorama.setAttribute("style", `left: ${leftShift}px`);
  } else {
    leftShift += panoramaWidth;
    panorama.setAttribute("style", `left: ${leftShift}px`);
  }
});

// Indicateurs de navigation
for (let i = 0; i < navIndicatorsContainer.childElementCount; i++) {
  navIndicatorsContainer.children[i].addEventListener("click", () => {
    leftShift = -Math.abs(panoramaWidth * i);
    panorama.setAttribute("style", `left: ${leftShift}px`);
  });
}
```

### 3. Avec qui avez-vous travaillé ?

Pour ce projet, j'ai travaillé seul mais j'ai demandé de l'aide à un ami que j'ai rencontré sur une communauté de développeurs.

# DOSSIER PROFESSIONNEL (DP)

## 4. Contexte

Nom de l'entreprise, organisme ou association ► *Studi*

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

Période d'exercice ► Du : *01/01/2023* au : *01/02/2023*

## 5. Informations complémentaires (facultatif)

## Activité-type 1 Développer la partie front-end d'une application web ou web mobile

Exemple n° 3 ► *Projet d'ECF (Quai Antique)*

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

Pour le projet d'ECF, nous avons pour but de réaliser le site d'un restaurant. Ce site devait avoir un système de réservation, des images pour afficher des plats par le chef du restaurant, et devait également pouvoir présenter les différents menus et cartes aux futurs clients du restaurant.

Pour ce projet, j'ai recherché des sites de restaurants pour m'en inspirer, ainsi que des produits locaux qu'un chef de restaurant voudrait cibler, pour pouvoir lui proposer un panel de couleurs adapté à ce qu'il souhaite présenter aux clients (modifiable si souhaité)

Ce site devait également avoir un gestionnaire d'images, qui permettait de gérer les images de la page d'accueil.

Cette évaluation nous demandait :

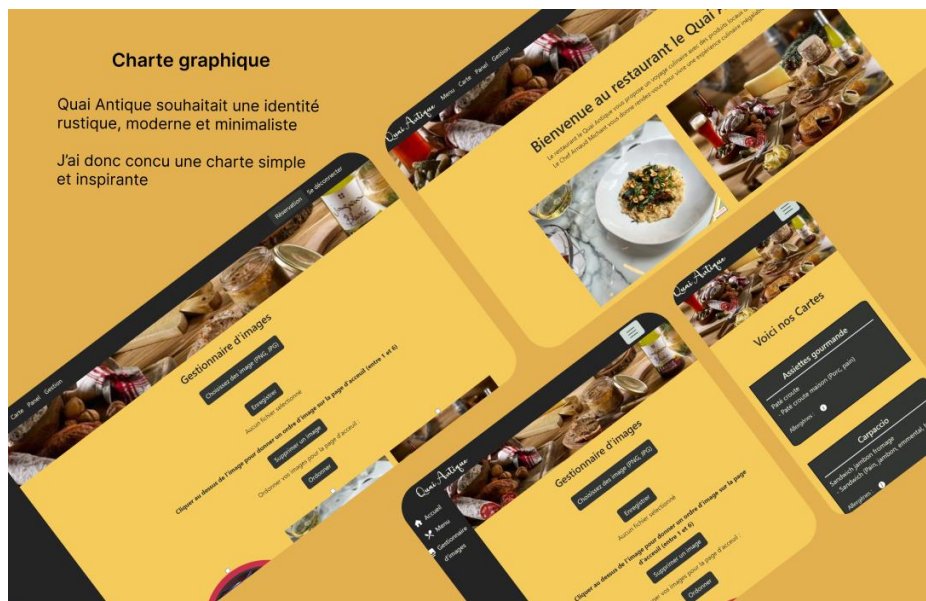
- De créer une maquette présentable au client
- D'utiliser un framework pour le CSS et le responsive du site (sélection de Bootstrap).
- D'utiliser JavaScript pour créer du contenu dynamique

**Pour le rendu final : voir annexe 2 page 42**

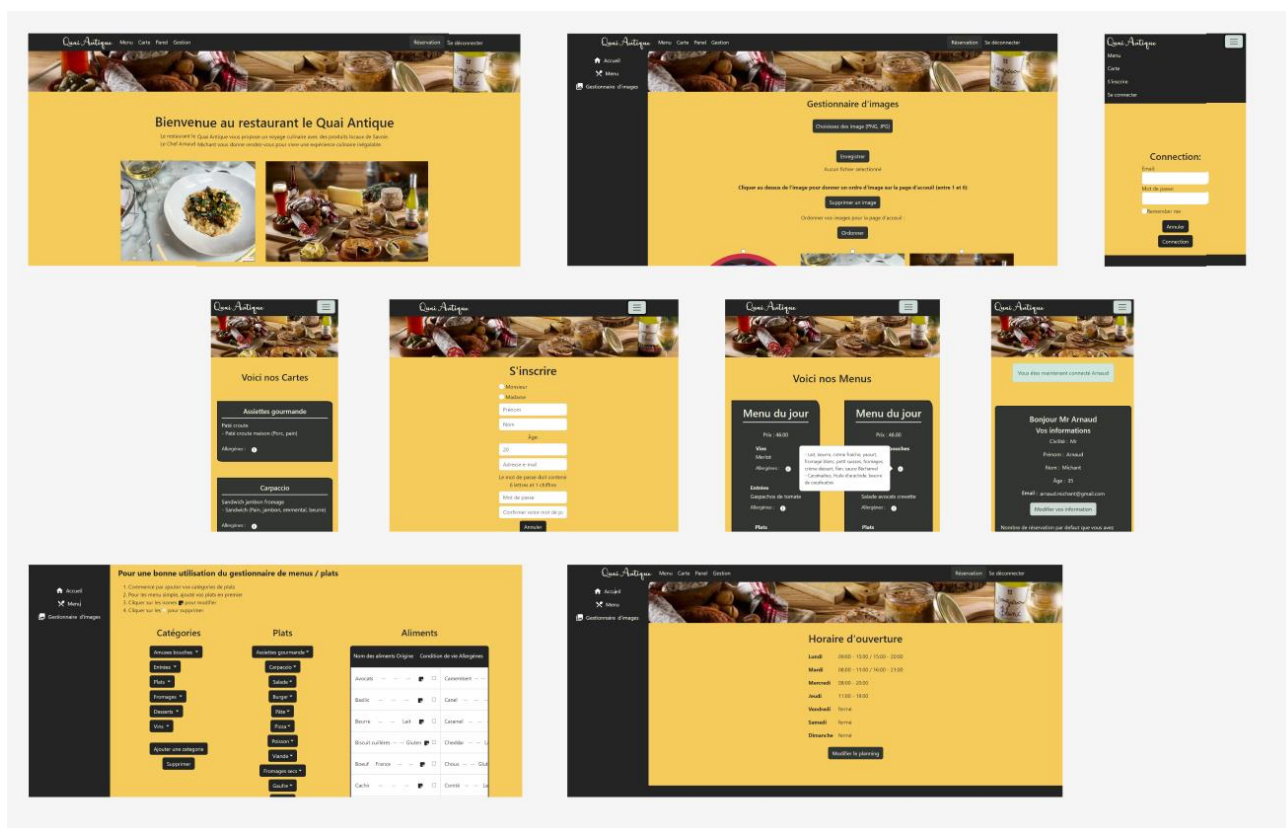
### 2. Précisez les moyens utilisés :

J'ai réalisé la maquette du site, la charte graphique, ainsi que la feuille de style sur l'outil en ligne Figma.

# DOSSIER PROFESSIONNEL (DP)



Charte graphique



Maquette



Feuille de style

Concernant la partie Développement, j'ai utilisé l'éditeur de code Visual Studio Code.

J'ai installé le framework Bootstrap sur le projet, pour qu'il n'y ait pas de problème de version. Pour son installation, j'ai utilisé l'outil « npm » de node.js.

```
npm install bootstrap@5.3.0
```

Pour que mon projet puisse avoir accès aux classes CSS disponibles dans ce framework, ainsi qu'au JavaScript compilé, il m'a fallu ajouter ces lignes :

```
<link rel="stylesheet" href="./node_modules/bootstrap/dist/css/bootstrap.css">
```

Extrait de la balise <head> de mes fichiers

```
<script src="./node_modules/bootstrap/dist/js/bootstrap.bundle.js"></script>
```

Script de mes fichiers

L'import de ce code me permet donc d'utiliser les classes du framework Bootstrap, pour styliser une fenêtre modale responsive contenant un formulaire, afin que les clients du restaurant puissent réserver.

```
<div class="modal fade" id="modal_calendar" tabindex="-1" aria-hidden="true">
  <div class="modal-dialog">
    <div class="modal-content" style="background-color: #f5cb5c;">
      <div class="modal-body">
        <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
        <form method="get" class="d-flex flex-column ">
          <label class="mb-3 mx-auto">
            Choisissez une date de réservation
          </label>
          <div class="d-flex gap-2 justify-content-center">
            <input class="form-control" type="date" id="calendar" name="calendar" max="<?= $dateEnd ?>" min="<?= $dateCurrent ?>">
            <input class="btn" id="validate_calendar" type="button" value="Valider" style="background-color: #333533 ;color: #e8eddf;">
          </div>
        </form>

        <div id="display_button" class="d-none flex-column flex-wrap justify-content-center gap-3 align-items-center">
          <div id="first_part" class="d-flex flex-column align-items-center">
            <h3 class="mt-2">Repas du midi</h3>

            <div class="my-2" id="part_1">...
          </div>
        </div>

          <div id="second_part" class="d-flex flex-column align-items-center">
            <h3 class="mt-2">Repas du soir</h3>

            <div class="my-2" id="part_2">...
```

Extrait de la balise <header>

Ce script JavaScript, ajoute du contenu à la fenêtre modale une fois que l'utilisateur a sélectionné une date de réservation. Ensuite, il pourra entrer un nombre de personne pour lequel il souhaite réserver, et également sélectionner l'horaire souhaité.

# DOSSIER PROFESSIONNEL (DP)

```
const displayCalendar = document.getElementById('display_button')
const validateCalendar = document.getElementById('validate_calendar')
const inputDate = document.getElementById('calendar')

// function returnNumber
const firstPart = document.getElementById('first_part')
const secondPart = document.getElementById('second_part')
const buttonCalendarPart1 = document.querySelectorAll('#button_calendar_part1')
const buttonCalendarPart2 = document.querySelectorAll('#button_calendar_part2')

let numberPart1 = document.createElement('p')
numberPart1.id = "number_reservation1"
let numberPart2 = document.createElement('p')
numberPart2.id = "number_reservation2"

validateCalendar.addEventListener('click', (event) => {
  const inputCalendar = inputDate.value
  displayCalendar.classList.replace('d-none', 'd-flex')

  returnNumberReservationPart1(inputCalendar)
  returnNumberReservationPart2(inputCalendar)
})
```

Extrait du script modal.js

### 3. Avec qui avez-vous travaillé ?

J'ai réalisé ce projet seul, tout en m'aidant des cours de la formation.

### 4. Contexte

Nom de l'entreprise, organisme ou association ► *Studi*

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

Période d'exercice ► Du : *16/01/2023* au : *23/05/2023*

### 5. Informations complémentaires (facultatif)



## Activité-type 2

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

Exemple n° 1 ► Conception d'une base de données (Quai antique)

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

Au cours de la formation, nous avons abordé la conception et le développement d'une base de données, à l'aide de la méthode Merise, du langage de données SQL et du langage back-end PHP (code inaccessible à l'utilisateur). Pour mettre en application ce que j'ai appris, et pour valider ces compétences, j'ai conceptualisé une base de données d'un restaurant, destinée aux utilisateurs et à l'administrateur.

Dans cette application :

- L'administrateur devait avoir un back office (service de gestion) qui permettait de gérer les menus, les cartes visibles pour les clients et les allergènes sur les aliments. Ce back office devait également contenir un questionnaire d'images pour modifier la page d'accueil, les plannings et le nombre de réservation disponible.
- Il fallait créer un système d'inscription et d'authentification pour le client, afin qu'il puisse se connecter, entrer un nombre de convive dans ses réglages de réservation, et réserver.

### 2. Précisez les moyens utilisés :

J'ai commencé par élaborer le MCD, le modèle conceptuel de données, qui consiste à créer les différentes entités qui composeront le système d'information, et à établir des relations entre elles.

Ainsi, j'ai obtenu mes entités souhaitées pour réaliser le site d'un restaurant, composées de leurs propriétés, comme exemple d'entités : « Menu », « Dishes », « Sub\_categorie », « Food » et « Users », et comme exemple de propriétés de la première entité : « id » qui sera de type **INTEGER** (nombre entier), « name » qui sera de type **VARCHAR** (chaîne de caractères).

Ensuite j'ai réfléchi aux cardinalités :

- Un aliment (food) appartient à aucun plat (dishes) ou plusieurs, un plat possède 1 ou plusieurs aliments.
- Un plat (dishes) appartient à 0 ou plusieurs menus, un menu a forcément un plat ou plusieurs.
- Un plat (dishes) appartient à une seule sous-catégorie (sub\_categorie), une sous-catégorie possède un ou plusieurs plats.

Une fois les cardinalités établies, je peux passer à l'élaboration du MLD, le modèle logique de données. Chaque entité devient une table, et leur identifiant devient la clé primaire de la table.

Le (\*) vaut 0 ou 1 dans l'exemple qui suit.

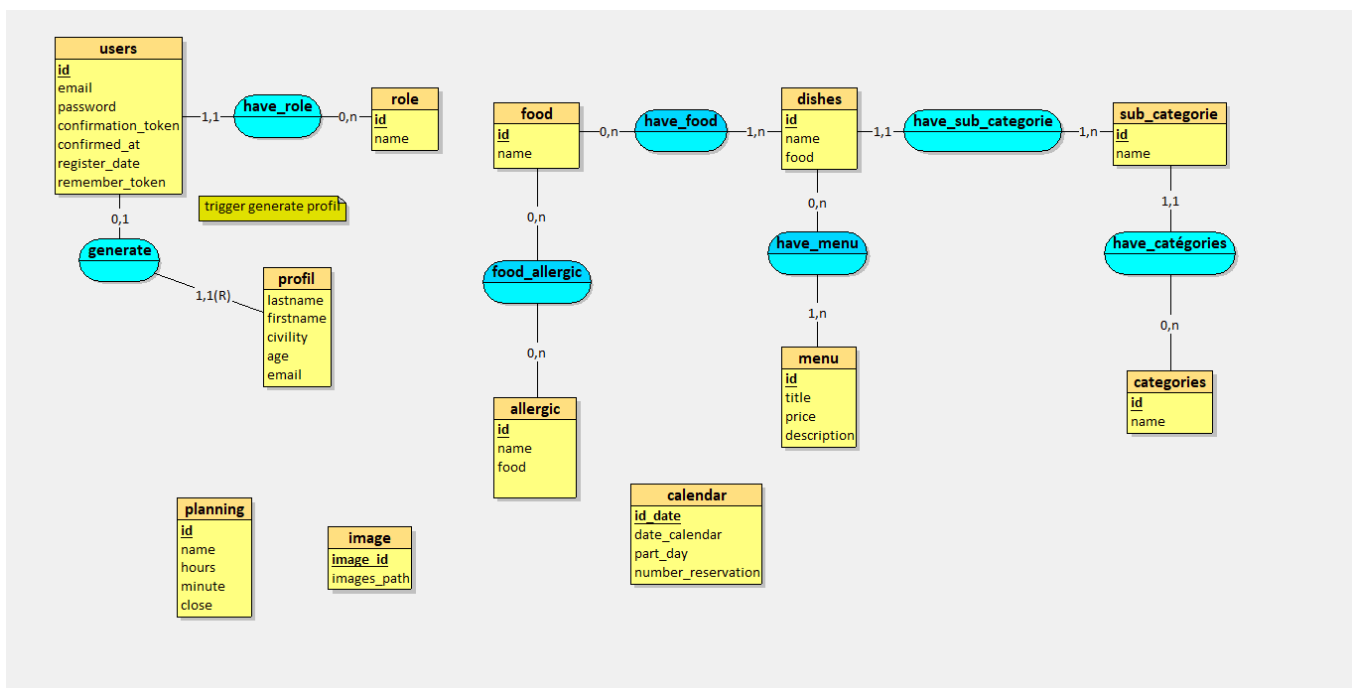
Pour les relations  $*,1 - *,n$  la clé primaire de la table côté  $*,n$  devient une clé étrangère de la table côté  $*,1$  créant ainsi la relation entre les tables.

Pour les relations  $*,n - *,n$  une table de liaison sera créée contenant les clés primaire des tables qui seront également les clé primaire de cette table créée, créant ainsi les relation entre les tables.

La table « have\_food » récupère le champ « food\_id » clé primaire qui fait référence à la clé primaire de la table « Food », elle récupère également le champ « dishes\_id » clé primaire qui fait référence à la clé primaire de la table « dishes ».

La table « dishes » récupère le champ « sub\_categorie\_id » clé étrangère qui fait référence à la clé primaire de la table « Sub\_categorie ».

Une fois toute ces étapes réalisées j'ai pu élaborer mon plan UML (Langage de Modélisation Unifié) qui consiste à réaliser un diagramme d'objets qui contiendra mes tables, dans lequel sera renseigner le type des propriétés de l'objet ainsi que les relations de l'objet.



Plan UML

Une fois mon plan UML réalisé, J'ai utilisé Docker, pour l'utiliser j'ai dû l'installer sur ma machine (ordinateur) car je souhaité expérimenter cette technologie, conseillée par un ami. Elle permet de déployer des bases de données sur notre serveur local et de développée avec des multi-conteneur (plusieurs applications). Une fois Docker installé j'ai accès à la commande « docker ».

Pour ceci il m'a fallu créer une image dans docker avec la commande :

```
docker run unless-stopped mysql:8.0
```

Ensuite, dans mon fichier docker-compose.yml j'ai défini le service de base de données MySQL avec ces variables d'environnement, le port du service tout en spécifiant l'image utilisé.

```
version: '3.7'
services:
  mysql:
    image: mysql:8.0
    hostname: mysql
    container_name: quai_antique_bd
    restart: unless-stopped
    ports:
      - 3306:3306
    volumes:
      - ./mysql:/var/lib/mysql
    env_file:
      - database.env

volumes:
  mysql:
    name: quai_antique
```

Et pour finir la partie Docker j'ai démarré l'application avec la commande ci-dessous :

```
docker compose up
```

J'ai pu enfin écrire mon code SQL que j'ai implémenté via le projet et grâce au langage de programmation PHP, afin qu'il soit généré dans ma base de données.

```
$createRoleTable =  
'CREATE TABLE IF NOT EXISTS role(  
  role_id INT AUTO_INCREMENT,  
  role_name VARCHAR(50) NOT NULL,  
  PRIMARY KEY(role_id)  
)';  
  
$createUsersTable =  
'CREATE TABLE IF NOT EXISTS users(  
  user_id INT AUTO_INCREMENT,  
  user_email VARCHAR(50) NOT NULL,  
  user_password TEXT NOT NULL,  
  user_confirmation_token TEXT,  
  user_confirmed_at DATE,  
  user_register_date DATETIME,  
  user_remember_token TEXT,  
  user_role_id INT NOT NULL,  
  PRIMARY KEY(user_id),  
  FOREIGN KEY(user_role_id) REFERENCES role(role_id)  
)';  
  
$createProfilTable =  
'CREATE TABLE IF NOT EXISTS profil(  
  profil_id INT,  
  profil_lastname VARCHAR(30),  
  profil_firstname VARCHAR(30),  
  profil_civility VARCHAR(20),  
  profil_age INT(10),  
  profil_email VARCHAR(50),  
  profil_default_reservation INT,  
  PRIMARY KEY(profil_id),  
  FOREIGN KEY(profil_id) REFERENCES users(user_id)  
)';
```

Extrait du fichier db.php

J'ai d'abord créé les tables et vérifié grâce à l'opérateur « IF NOT EXISTS » que mes tables n'existent pas pour éviter une duplication des tables lors du lancement du script, en précisant le type de leurs propriétés et leurs restrictions ainsi que leur clés primaire « PRIMARY KEY », leur clés étrangères et leurs références (mise en relation).

### 3. Avec qui avez-vous travaillé ?

J'ai réalisé ce projet seul dans le cadre d'une évaluation.

# DOSSIER PROFESSIONNEL (DP)

## 4. Contexte

Nom de l'entreprise, organisme ou association ► *Studi*

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

Période d'exercice ► Du : *16/01/2023* au : *23/05/2023*

## 5. Informations complémentaires (facultatif)

## Activité-type 2

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

Exemple n° 2 ► Gestion des données du projet (Quai antique)

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

Pour le projet d'ECF du restaurant « Quai Antique », je devais relier mon projet à ma base de données, et concevoir tout la gestion et les opérations que le chef du restaurant souhaité pouvoir réaliser sur son site web.

Cette application devait donc :

- Avoir un fichier avec les informations pour connecter mon projet a la base de données
- Alimenter mes tables de la base de données pour pouvoir tester les opérations demandées
- Gérer la partie administrative du site

**Pour le rendu final : voir annexe 3 et 4 page 43**

### 2. Précisez les moyens utilisés :

Pour commencer j'ai relié mon projet a la base de données en local que j'ai crée ainsi que la partie production (une fois le site déployer).

```
<?php
if ($_SERVER['SERVER_NAME'] == "localhost") {
    // Information a modifier pour une utilisation local
    $host_name = '127.0.0.1';
    $database = 'quaiantique';
    $user_name = 'admin';
    $password = 'quaiantique';
} else {
    $host_name = 'db5013125203.hosting-data.io';
    $database = 'dbs11016128';
    $user_name = 'dbu1243084';
    $password = 'quai.AntiqueArnaud3';
}
```

Extrait de fichier info\_db.php

En reliant ma base de données au site web, grâce à l'extension PDO (PHP Data Objects) de PHP j'ajoute les tables nécessaires pour mon projet, en vérifiant qu'elle ne se duplique pas.

```
try {
    $pdo = new PDO("mysql:host=$host_name; dbname=$database;", $user_name, $password);
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $pdo->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_OBJ);

    $pdo->exec($createAllergicTable);
    $pdo->exec($createFoodTable);
    $pdo->exec($createFoodAllergicTable);
    $pdo->exec($createCategorieFoodTable);
    $pdo->exec($createSubcategorieFoodTable);
    $pdo->exec($createDischesTable);
    $pdo->exec($createHaveFoodTable);
    $pdo->exec($createMenuTable);
    $pdo->exec($createHaveMenuTable);
    $pdo->exec($createTableImage);
    $pdo->exec($createTableCalendar);

    $pdo->exec($createRoleTable);
    $pdo->exec($createUsersTable);
    $pdo->exec($createProfilTable);
    $pdo->exec($createTriggerUserToProfil);

    $pdo->exec($createPlanning);
} catch (PDOException $e) {
    echo "Connection échoué: " . $e->getMessage();
}
```

```
$createHaveMenuTable = 'CREATE TABLE IF NOT EXISTS have_menu(
    menu_id INT,
    dishes_id INT,
    menu_categorie INT,
    PRIMARY KEY(menu_id, dishes_id),
    FOREIGN KEY(menu_id) REFERENCES menu(menu_id),
    FOREIGN KEY(dishes_id) REFERENCES dishes(dishes_id)
);
';
```

Extrait du fichier db.php

J'y ajoute également les données que je souhaite en vérifiant qu'elle n'existe pas pour pouvoir gérer les rôles.

```
foreach ($createRole as $role) {
    $reqInsertRole = $pdo->prepare('INSERT INTO role SET role_name = ?');

    $reqSelectRole = $pdo->prepare('SELECT * FROM role WHERE role_name = ?');
    $reqSelectRole->execute([$role]);
    $selectRole = $reqSelectRole->fetch();

    if (!$selectRole) {
        $reqInsertRole->execute([$role]);
    }
}
```

Pour ce qui est du register (l'inscription de l'utilisateur), j'ai créé un fichier contenant le formulaire avec la méthode « post » (envoi de données dans le corps de la requête HTTP) pour que le client remplisse les coordonnées nécessaires pour la connexion.

```
<div class="d-flex flex-column align-items-center gap-2">
  <h1>S'inscrire</h1>
  <form action="" method="post" class="d-flex flex-column gap-2">
    <div>
      <input class="form-check-input" type="radio" id="Mr" name="civilite" value="Mr">
      <label for="Mr" class="inline-label">Monsieur</label>
    </div>
    <div>
      <input class="form-check-input" type="radio" id="Mme" name="civilite" value="Mme">
      <label for="Mme" class="inline-label">Madame</label>
    </div>
    <input class="form-control" type="text" name="firstname" id="surname" placeholder="Prénom">
    <input class="form-control" type="text" name="lastname" id="name" placeholder="Nom">
    <label class="text-center" for="age">Âge:</label>
    <input class="form-control" type="select" name="age" id="age" placeholder="20">
    <input class="form-control" type="email" name="email" id="email" placeholder="Adresse e-mail">
    <label class="text-center">Le mot de passe doit contenir<br>6 lettres et 1 chiffres</label>
    <input class="form-control" type="password" name="password" placeholder="Mot de passe">
    <input class="form-control" type="password" name="password_confirm" placeholder="Confirmer votre mot de passe">
    <div class="d-flex flex-column align-items-center justify-content-center gap-2">
      <a class="text-decoration-none text-dark" href="/">
        <button class="btn" style="color: #e8eddf; background-color: #242423 ;">Annuler</button>
      </a>
      <button class="btn" style="background-color: #242423 ;color: #e8eddf;" type="submit">S'inscrire</button>
    </div>
  </form>
</div>
```

Avec les vérifications nécessaires pour vérifier les données rentrées dans le formulaire par l'utilisateur. Par exemple l'utilisation de la fonction « preg\_match() » pour vérifier les caractères du champ du formulaire suivant une expression régulière (regex), et la fonction empty() pour vérifier si ce champs n'est pas vide.



```
if (!empty($_POST)) {  
    $errors = array();  
    require './sys/db.php';  
  
    if (empty($_POST['civilite'])) { ...  
    }  
  
    if (empty($_POST['firstname']) || !preg_match('/^[a-zA-Zéèâàô]+$/ ', $_POST['firstname'])) { ...  
    }  
  
    if (empty($_POST['lastname']) || !preg_match('/^[a-zA-Zéèâàô]+$/ ', $_POST['lastname'])) { ...  
    }  
  
    if (empty($_POST['age']) || !preg_match('/^[0-9]{2}/ ', $_POST['age'])) { ...  
    }  
  
    if (empty($_POST['email']) || !filter_var($_POST['email'], FILTER_VALIDATE_EMAIL)) {  
        $errors['email'] = "Votre email n'est pas valide";  
    } else {  
        $req = $pdo->prepare('SELECT user_id FROM users WHERE user_email = ?');  
        $req->execute([$_POST['email']]);  
        $user = $req->fetch();  
        if ($user) {  
            $errors['email'] = "Cette adresse email est déjà utilisé";  
        }  
    }  
}  
  
if (!preg_match('/^[a-zA-Z]{6,}[0-9]{1,}/ ', $_POST['password'])) {  
    $errors['password'] = "Le mot de passe ne comporte pas au minimum 1 caractères numérique";  
  
    if (empty($_POST['password']) || $_POST['password'] != $_POST['password_confirm']) {  
        $errors['password'] = "Les mots de passe ne correspondent pas";  
    }  
}
```

Extrait du fichier register.php

Avec un envoi d'email de confirmation d'enregistrement, via la classe PHPMailer, si toutes les données attendues sont soumises par le formulaire. Le mot de passe sera haché (transformation du mot de passe grâce à une méthode de chiffrement) pour éviter que le mot de passe soit stocké dans la base de données en clair (tel que l'utilisateur l'a rentré). Les données attendues sont soumises à la base de données, pour enregistrer l'utilisateur avec des requêtes « INSERT » et « UPDATE » grâce à l'extension PDO.

```
use PHPMailer\PHPMailer\PHPMailer;  
use PHPMailer\PHPMailer\Exception;
```

# DOSSIER PROFESSIONNEL (DP)

```
if (empty($errors)) {
    $password = password_hash($_POST['password'], PASSWORD_BCRYPT);
    $token = rand();
    $req = $pdo->prepare("INSERT INTO users SET user_email = ?, user_password = ?, user_confirmation_token = ?, user_register_date = NOW(), user_role_id = 1");
    $req->execute([$_POST['email'], $password, $token]);
    $userId = $pdo->lastInsertId();
    $reqProfil = $pdo->prepare("UPDATE profil SET profil_lastname = ?, profil_firstname = ?, profil_age = ?, profil_civility = ? WHERE profil_id = ?");
    $reqProfil->execute([$_POST['lastname'], $_POST['firstname'], $_POST['age'], $_POST['civility'], $userId]);
    $mail = new PHPMailer(true);

    $protocol = explode('/', $_SERVER['SERVER_PROTOCOL']);
    if (isset($protocol) && $protocol[0] == 'HTTP') { ...
    } else { ...
    }

    $url .= "://";
    $url .= $_SERVER['HTTP_HOST'];
    // echo $url;

    try {
        $secret_m = 'byzloakvucfvloe';
        $username_m = 'arnaud.michant@gmail.com';
        $username = 'Arnaud Michant';
        $subject = "Confirmation de votre compte " . $_POST['firstname'] . " " . $_POST['lastname'];
        $mailBody = "Merci de cliquer sur le lien ci-dessous pour confirmer votre compte <br> $url/sys/confirm.php?id=$userId&token=$token";

        //Server settings...
        $mail->SMTPDebug = 2; // debugage: 1 = Erreurs et messages, 2 = messages seulement
        $mail->isSMTP(); //Send using SMTP
        $mail->Host = 'smtp.gmail.com'; //Set the SMTP server to send through
        $mail->SMTPAuth = true; //Enable SMTP authentication
        $mail->Username = $username_m; //SMTP username
        $mail->Password = $secret_m; //SMTP password
        $mail->SMTPSecure = 'TLS'; // STARTTLS pour outlook //Enable implicit TLS encryption
        $mail->Port = 587; //TCP port to connect to; use 587 if you have set 'SMTPSecure = PHPMailer::ENCRYPTION_STARTTLS'

        $mail->smtpConnect();
        //Recipients
    }
```

Extrait du fichier register.php

Grâce à l'extension PDO, un formulaire que l'utilisateur/administrateur devra remplir et le back-end que j'ai réalisé, celui-ci pourra traiter avec les données enregistrées, par exemple pour se connecter au site pour accéder à du contenu disponible pour des clients ou l'administration du site.

```
<div class="mx-4">
    <div class="d-flex flex-column align-items-center mt-5">
        <h1 class="mt-2">Connection:</h1>
        <form method="POST" class="d-flex flex-column">
            <label>Email:</label>
            <input class="form-control" type="text" name="email">
            <label for="">Mot de passe:</label>
            <input class="form-control" type="password" name="password">
            <label class="py-2">
                <input class="form-check-input" type="checkbox" name="remember" value="1" />Remember me
            </label>
            <div class="d-flex flex-column align-items-center justify-content-center gap-2 mt-2">
                <a href="/">
                    <button class="btn" style="color: #e8eddf; background-color: #242423 ;">Annuler</button>
                </a>
                <input class="btn" style="color: #e8eddf; background-color: #242423 ;" type="submit" value="Connection">
            </div>
        </form>
    </div>
</div>
```

```
if (!empty($_POST['email']) && !empty($_POST['password'])) {
    require_once './sys/db.php';

    try {
        $req = $pdo->prepare('SELECT * FROM users WHERE user_email = ? ');
        $req->execute([$_POST['email']]);
        $user = $req->fetch();

        if ($user->user_confirmed_at == 'NULL') {
            $errors['confirm'] = "L'utilisateur n'a pas confirmé l'email";
        } else {
            if (password_verify($_POST['password'], $user->user_password)) {
                $reqProfil = $pdo->prepare('SELECT * FROM profil WHERE profil_id = ? ');
                $reqProfil->execute([$user->user_id]);
                $userProfil = $reqProfil->fetch();
                $_SESSION['auth'] = $user;
                $_SESSION['flash']['success'] = "Vous êtes maintenant connecté " . $userProfil->profil_firstname;

                if ($_POST['remember']) {
                    $remember_token = rand();
                    $req = $pdo->prepare('UPDATE users SET user_remember_token = ? WHERE user_id = ?');
                    $req->execute([$remember_token, $user->user_id]);
                    setcookie('remember', $user->user_id . '/' . $remember_token . sha1($user->user_id . 'ratonlaveurs'), time() + 60 * 60 * 24 * 7, '/');
                }

                // header('Location: panel_user.php;Refresh: 0;');
                $urlLogin = 'panel_user.php';
                echo '<script type="text/javascript">window.location.href="' . $urlLogin . '";</script>';
                exit();
            } else {
                $errors['Email'] = "Email ou mot de passe incorrect !";
            }
        }
    } catch (PDOException $e) {
        $errors['user'] = "L'utilisateur n'existe pas !";
    }
}
```

Extrait du fichier login.php

Suivant le rôle de l'utilisateur il aura accès à certaines pages dans la navbar et s'il est connecté le bouton « s'inscrire » et le bouton se « connecter » seront remplacés par « se déconnecter » du fait que la variable `$_SESSION['auth']` ait enregistré l'utilisateur lors de la connexion.

# DOSSIER PROFESSIONNEL (DP)

```
<div class="navbar-nav" style="z-index: 2;">
  <a id="button_header" href="./menu_show.php" class="nav-link" style="color: #e8eddf;">Menu</a>
  <a id="button_header" href="./card_show.php" class="nav-link" style="color: #e8eddf;">Carte</a>
  <?php
  if (isset($_SESSION['auth'])->user_email) : ?>
    <a id="button_header" href="./panel_user.php" class="nav-link" style="color: #e8eddf;">Panel</a>
  <?php endif;
  $reqSelectRole = $pdo->prepare('SELECT * FROM role WHERE role_id = ?');
  $reqSelectRole->execute([$_SESSION['auth']->user_role_id]);
  $selectRole = $reqSelectRole->fetch();

  if ($selectRole->role_name == 'admin') : ?>
    <a id="button_header" href="./panel.php" class="nav-link" style="color: #e8eddf;">Gestion</a>
  <?php endif; ?>
</div>

<div class="d-flex navbar-nav">
  <?php
  if (isset($_SESSION['auth'])) : ?>
    <button id="button_show_calendar" class="btn" data-bs-target="#modal_calendar" data-bs-toggle="modal">
      Réserve
    </button>
    <a id="button_header" class="nav-link" href="/logout.php" style="color: #e8eddf;">Se déconnecter</a>
  <?php else : ?>
    <a id="button_header" class="nav-link" href="/register.php" style="color: #e8eddf;">S'inscrire</a>
    <a id="button_header" class="nav-link" href="/login.php" style="color: #e8eddf;">Se connecter</a>
  <?php endif; ?>
</div>
```

## 3. Avec qui avez-vous travaillé ?

J'ai réalisé ce projet seul en me référant au cours de la formation et la documentation.

## 4. Contexte

Nom de l'entreprise, organisme ou association ► *Studi*

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

Période d'exercice ► Du : *16/01/2023* au : *23/05/2023*

## 5. Informations complémentaires (facultatif)

## Activité-type 2

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

Exemple n° 3 ► Gestionnaire d'images (Quai antique)

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

Pour le projet d'ECF nous avons pour objectif d'avoir un gestionnaire d'images, pour que l'administrateur du site puisse afficher des images, gérer leurs ordres d'affichage dans une page précise ou les supprimer.

Afin de réaliser ce gestionnaire d'images il m'a fallu :

- Créer la page où les images seront affichées
- Avoir un fichier qui me permettra d'ajouter/supprimer les images sur le projet ou de les sélectionner pour qu'elles s'affichent
- Avoir un dossier pour enregistrer des images
- Ainsi qu'une table de données afin d'enregistrer un ordre pour l'affichage de mes images

**Pour le rendu final : voir annexe 4 page 44**

### 2. Précisez les moyens utilisés :

Pour débiter j'ai créé un fichier avec un formulaire grâce à la balise `<form>`, en précisant son attribut « `enctype` », afin que le formulaire envoie au serveur le bon encodage du type MIME (Multipurpose Internet Mail Extensions). Dans ce formulaire j'y ai ajouté une balise `<label>` afin de donner une explication sur les extensions d'images attendues (.png, .jpg) et une balise `<input>` de type « `file` » avec l'attribut « `multiple` » afin que l'utilisateur puisse ajouter plusieurs images de son appareil.

```
<form method="post" class="d-flex flex-column justify-content-center align-items-center" enctype="multipart/form-data">
  <div class="d-flex flex-column align-items-center flex-wrap">
    <label class="btn p-2" style="background-color: #333533 ;color: #e8eddf;" for="image_uploads">Choisissez des image (PNG, JPG)</label>
    <input class="" type="file" id="image_uploads" name="image_uploads[]" accept=".jpg, .jpeg, .png" multiple />
  </div>
  <br>
  <input class="btn mb-2" style="background-color: #333533 ;color: #e8eddf;" type="submit" value="Enregistrer">
  <div id="preview" class="d-inline-flex flex-wrap justify-content-center align-items-center gap-3">
    <p>Aucun fichier sélectionné</p>
  </div>
</form>
```

Pour la partie enregistrement, il fallait un dossier où enregistrer les images que l'utilisateur ajoutait de son appareil, donc j'ai créé ce dossier en ajoutant une vérification, afin de vérifier si le dossier existe bien au moment où l'utilisateur enregistrera une/des images.

Je n'ai plus qu'à récupérer mes images grâce à la superglobal (variable prédéfinies de PHP) « \$\_FILES » et les enregistré dans le dossier souhaité avec la fonction « move\_uploaded\_file ».

```
<?php
// verification d'ajout d'image
if (isset($_FILES['image_uploads'])) {
    foreach ($_FILES['image_uploads']['tmp_name'] as $f => $file) {
        if (isset($_FILES['image_uploads']['tmp_name'][$f])) {

            $tmp_name = $_FILES['image_uploads']['tmp_name'][$f];
            $name = basename($_FILES["image_uploads"]["name"][$f]);

            $moveUpload = move_uploaded_file($tmp_name, "$uploadDir/$name");
            if ($moveUpload) {
                echo '<p>La photo ' . $name . ' a bien été envoyée.</p>';
            }
        }
    }
}
```

Extrait du fichier image\_manager.php

J'ai développé une partie en JavaScript qui me permettait de vérifier si le fichier sélectionné était bon pour qu'il soit enregistré sur le projet, s'il l'était, l'utilisateur avait un rendu des images qu'il avait sélectionnée, sur l'appareil avant la soumission de l'enregistrement, dans le dossier du site web.

# DOSSIER PROFESSIONNEL (DP)

```
const input = document.getElementById('image_uploads')
const preview = document.getElementById('preview')

input.style.opacity = 0

const fileTypes = [
  'image/apng',
  'image/bmp',
  'image/gif',
  'image/jpeg',
  'image/pjpeg',
  'image/png',
  'image/svg+xml',
  'image/tiff',
  'image/webp',
  'image/x-icon',
]

function validFileType(file) {
  return fileTypes.includes(file.type)
}

function returnFileSize(number) {
  if (number < 1024) {
    return `${number} bytes`
  } else if (number >= 1024 && number < 1048576) {
    return `${(number / 1024).toFixed(1)} KB`
  } else if (number >= 1048576) {
    return `${(number / 1048576).toFixed(1)} MB`
  }
}
```

```
function updateImageDisplay() {
  while (preview.firstChild) {
    preview.removeChild(preview.firstChild)
  }

  const curFiles = input.files
  if (curFiles.length === 0) {
    const para = document.createElement('p')
    para.textContent = 'Aucun fichier sélectionné'
    preview.appendChild(para)
  } else {
    const list = document.createElement('div')
    list.className =
      'd-inline-flex gap-4 flex-wrap flex-column flex-md-row justify-content-center align-items-center px-4'

    for (const file of curFiles) {
      const listItem = document.createElement('li')
      listItem.className = 'd-flex align-content-center'
      listItem.style = 'list-style-type: none;'
      const para = document.createElement('p')
      if (validFileType(file)) {
        const image = document.createElement('img')
        image.className = 'd-block'
        image.src = URL.createObjectURL(file)
        image.style.width = '15rem'

        listItem.appendChild(image)
      } else {
        para.textContent = `Le nom du fichier ${file.name}: N'a pas un type valide. Update your selection.`
        listItem.appendChild(para)
      }
      preview.appendChild(listItem)
    }
  }
}
input.addEventListener('change', updateImageDisplay)
```

Extrait du fichier image\_manager.php

Une fois que mes images sont enregistrées sur le site, je veux que l'administrateur ai accès à toutes les images qu'il a enregistré sur le projet, pour qu'il puisse leur donner un ordre d'affichage sur la page voulu ou les supprimer s'il souhaite les changer. Pour ceci je vais englober le tout dans un formulaire <form> et y ajouter un input de type « checkbox » à chaque image, pour pouvoir cocher celle souhaité.

Les images seront afficher avec la fonction « readdir() », cette fonction me permet de lire tous les chemin des images de mon dossier donc je n'ai plus qu'à boucler dessus grâce à la fonction « while() », il me faut ensuite ajouter ma balise <img> qui aura dans son attribut « src » le chemin récupérer de l'image.

J'ai également ajouté les boutons pour supprimer et ordonner mes images.



```
$openFiles = opendir('./images/image_base');
$uploadDir = './images/image_base';
?>
<form method="post" id="formImage" class="d-flex flex-column-reverse align-items-center my-3 gap-3">
  <?php
  // formulaire d'image
  if (readdir($openFiles)) {
    echo '<div class="d-flex flex-wrap justify-content-center gap-3">';
    while ($files_current = readdir($openFiles)) {
      $extension = strtolower(strrchr($files_current, '.'));
      if ($extension == '.jpg' || $extension == '.jpeg' || $extension == '.png') {
        $pathImageRead = $uploadDir . '/' . $files_current;

        if (file_exists($pathImageRead)) {
          $haveImage = true;
          echo '<div class="d-flex flex-column align-items-center">
            <input type="checkbox" name="path_image[]" id="path_image" value="" . $pathImageRead . ">
            <img style="width:20rem;" src="" . $pathImageRead . "" title="" . $files_current . ">
          </div>';
        }
      }
    }
    echo '</div>';
  }
  // gestion d'images
  ?>
  <div id="add_input"></div>

  <?php
  if (isset($haveImage)) {
    echo "<div class='d-flex flex-column align-items-center gap-3'>
    <label class='text-center'><b>Cliquer au dessus de l'image pour donner un ordre d'image sur la page d'accueil (entre 1 et 6) </b></label>;
    echo '<input type='submit' class='btn' style='background-color: #333533 ;color: #e8eddf;' name='delete_img' value='Supprimer un image'>;
    echo '<label>Ordonner vos images pour la page d'accueil :</label>
    <input type='submit' class='btn overflow-auto' style='background-color: #333533 ;color: #e8eddf;' id='select_position' name='select_position' value='Ordonner'>
    </div>';
  }
  ?>
</form>
```

Extrait du fichier image\_manager.php

Pour ce qui est de l'ordre des images sur la page souhaitée, j'ai créé un script JavaScript qui me permet d'ajouter un champ de formulaire invisible avec sa valeur, sur la page lors de la sélection d'une image. Et lorsque je clique sur le bouton « Ordonner » j'enregistre le chemin de mes images avec leurs ordres dans une base de données, avec une limitation pour le rendu sur la page souhaité, soit 2, 4 ou 6 images.

```
<script>
  // script Order
  const form = document.getElementById('formImage')
  const buttonCheckbox = document.querySelectorAll('#path_image')
  let countSelect = 1
  const addInput = document.getElementById('add_input')
  const errorsSelect = document.getElementById('errors_select')
  const buttonOrder = document.getElementById('select_position')

  for (let checkbox of buttonCheckbox) {
    let input = document.createElement('input')
    checkbox.onchange = () => {
      if (checkbox.checked) {
        // si l'element est supprimé on récupère son ordre
        pathImage = checkbox.value
        input.type = "hidden"
        input.name = "image_path_order[]"
        input.value = pathImage
        addInput.append(input)

        if (checkbox.checked && countSelect < addInput.childElementCount) {
          // si le count select et plus petit que le nombre d'element du tableau
          countSelect = addInput.childElementCount
        } else {
          if (countSelect == addInput.childElementCount) {
            countSelect = addInput.childElementCount
            countSelect += 1
          } else {
            countSelect++
          }
        }
      } else {
        // voir le test sinon inseré 12/05
        regex = /[0-9*]/gm
        intInput = input.name.match(regex)
        countSelect--
        input.remove()
      }

      if (countSelect == 7) {
        countSelect = 1
      }

      buttonOrder.onclick = (event) => {
        if (addInput.childElementCount == 6 || addInput.childElementCount == 4 || addInput.childElementCount == 2) {
          // pErrors.remove()
        } else {
          event.preventDefault()
          alert("Choisissez un nombre paire (2-4-6) d'images")
        }
      }
    }
  }
}
</script>
```

```
if (isset($_POST['select_position'])) {  
    if (count($_POST['image_path_order']) == 2 || count($_POST['image_path_order']) == 4 || count($_POST['image_path_order']) == 6) {  
        // si le nombre d'images de la bd est different du nombre d'images ajouter a insérer alors on supprime les autres.  
        // et on modifie celle existante  
        $reqInsertImage = $pdo->prepare('INSERT INTO images SET images_id = ?, images_path = ?');  
        $reqUpdateImage = $pdo->prepare('UPDATE images SET images_path = ? WHERE images_id = ?');  
  
        if ($countTableImage->number_images > count($_POST['image_path_order'])) {  
            $reqDeleteImageSuperiorAtCountTable = $pdo->prepare('DELETE FROM images WHERE images_id > ?');  
            $reqDeleteImageSuperiorAtCountTable->execute([count($_POST['image_path_order'])]);  
            // modify image_path  
            foreach ($_POST['image_path_order'] as $o => $orderImage) {  
                $reqUpdateImage->execute([$orderImage, $o + 1]);  
            }  
        } else {  
            // insertion en base de donnée  
            foreach ($_POST['image_path_order'] as $o => $orderImage) {  
                if ($o + 1 > $countTableImage->number_images) {  
                    $reqInsertImage->execute([$o + 1, $orderImage]);  
                } else {  
                    $reqUpdateImage->execute([$orderImage, $o + 1]);  
                }  
            }  
        }  
    } elseif (count($_POST['image_path_order']) < 6) {  
        $errors['bad_number'] = "Vous ne pouvez selection que 6 images";  
    }  
}
```

Pour le rendu des images sur la page souhaitée, afin de réussir l'objectif de l'ECF, je récupère le chemin de mes images, en vérifiant qu'il y est bien quelque chose d'enregistré dans la table de la base de données, si la table de la base de données est vide, des images par défaut seront affichées.

Ensuite je boucle sur toutes les informations enregistrées dans ma base de données en récupérant l'ordre et le chemin d'enregistrement des images pour les ajouter dans une balise <img>, j'ai également ajouté un style suivant l'ordre dans lequel elle apparaisse.

```
$reqSelectImage = $pdo->prepare('SELECT * FROM images');
$reqSelectImage->execute();
$selectImage = $reqSelectImage->fetchAll();

if ($selectImage) :
    // s'il y a des images dans la base de donnée on les affiche
    foreach ($selectImage as $i => $image) {
        // si je récupère bien les images alors je vais boucler dessus pour avoir les mettre dans la page d'accueil
        $nameImage = strtolower(strrchr($image->images_path, '/'));

        if ($i % 2 == 1) {
            if ($i == 3 && $i % 3 == 0) {
                echo '<div class="col-lg-5">
                
                </div>';
                echo '</div>';
            } else {
                echo '<div class="col-lg-6">
                
                </div>';
                echo '</div>';
            }
        } else {
            if ($i == 2 && $i % 2 == 0) {
                echo '<div id="div_image" style="width: 70%;" class="row gap-2 justify-content-center">';
                echo '<div class="col-lg-6">
                
                </div>';
            } else {
                echo '<div id="div_image" style="width: 70%;" class="row gap-2 justify-content-center">';
                echo '<div class="col-lg-5">
                
                </div>';
            }
        }
    }
}

?>
<?php else : ?>
    <div class="row gap-4 justify-content-center">
        <div class="col-lg-5">
            
        </div>
        <div class="col-lg-6">
            
        </div>
    </div>
    <div class="row gap-4 justify-content-center">
        <div class="col-lg-6">
            
        </div>
        <div class="col-lg-5">
            
        </div>
    </div>

```

### 3. Avec qui avez-vous travaillé ?

J'ai réalisé ce projet seul.

# DOSSIER PROFESSIONNEL (DP)

## 4. Contexte

Nom de l'entreprise, organisme ou association ► *Studi*

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

Période d'exercice ► Du : *16/01/2023* au : *23/05/2023*

## 5. Informations complémentaires (facultatif)

## DOSSIER PROFESSIONNEL (DP)

### Titres, diplômes, CQP, attestations de formation

*(facultatif)*

Intitulé	Autorité ou organisme	Date
CAP Menuisier Installateur	AFPIA	02/07/2013

## DOSSIER PROFESSIONNEL (DP)

### Déclaration sur l'honneur

Je soussigné(e) Cimino Thomas ,  
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 Hauterives le 26/06/2023  
pour faire valoir ce que de droit.

Signature : Cimino Thomas

## Documents illustrant la pratique professionnelle

*(facultatif)*

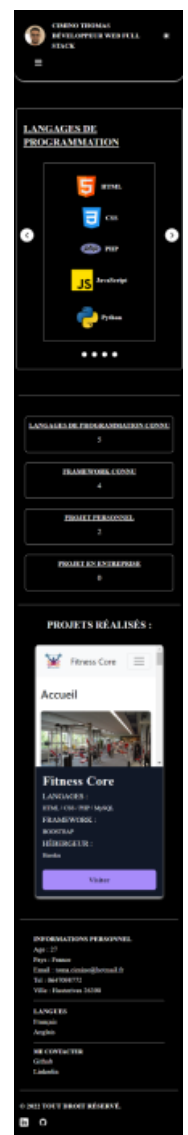
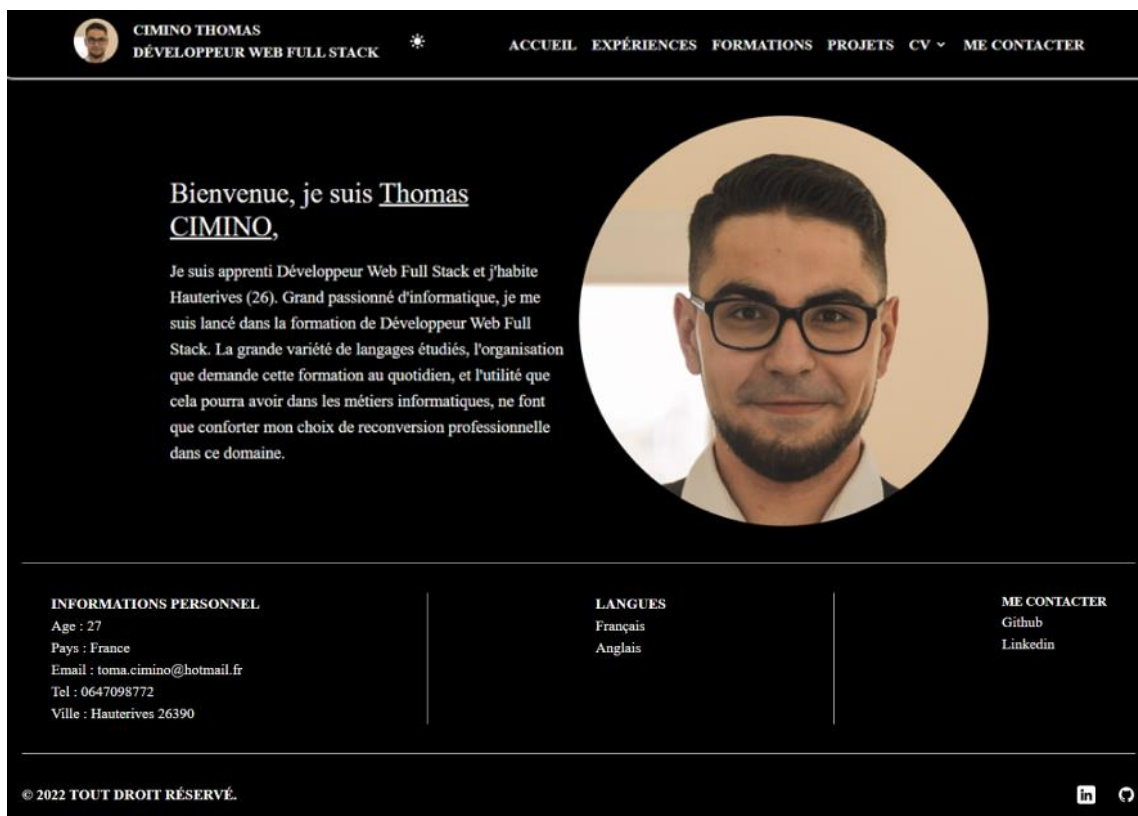
Intitulé
Annexe 1 : Mon portfolio
Annexe 2 : Quai antique
Annexe 3 : Formulaire d'inscription et de connexion
Annexe 4 : Navbar avec les droits administrateur
Annexe 5 : Gestionnaire d'images



# DOSSIER PROFESSIONNEL (DP)

## ANNEXES

### Annexe 1 – Mon portfolio



Vue ordinateur et mobile

## Annexe 2 – Quai Antique



Vue ordinateur

# DOSSIER PROFESSIONNEL (DP)

## Annexe 3 – Formulaire d'inscription et de connexion

**S'inscrire**

☐ Monsieur  
☐ Madame

Prénom

Nom

Âge:

20

Adresse e-mail

Le mot de passe doit contenir 6 lettres et 1 chiffres

Mot de passe

Confirmer votre mot de p

Annuler

Mot de passe

Confirmer votre mot de p

Annuler

S'inscrire

**Le Quai Antique**  
Restaurant gastronomique  
Chambery 73000  
Lorem Ipsum  
Tel: 06.00.00.00.00  
Mail du Chef: arnaud.michant@gmail.com  
Lundi Mardi Mercredi Jeudi Vendredi Samedi Dimanche

**Quai Antique**  
Menu  
Carte  
S'inscrire  
Se connecter

**Connection:**

Email:

35

Mot de passe:

\*\*\*\*\*

☐ Remember me

Annuler

Connection

**Le Quai Antique**  
Restaurant gastronomique  
Chambery 73000  
Lorem Ipsum  
Tel: 06.00.00.00.00  
Mail du Chef: arnaud.michant@gmail.com  
Lundi Mardi Mercredi Jeudi Vendredi Samedi Dimanche

Vue mobile

## Annexe 4 – Navbar avec les droits administrateur

**Quai Antique**

Menu

Carte

Panel

Gestion

Réservation

Se déconnecter

Vue tablette

## Annexe 5 – Gestionnaire d'images

Quai Antique

Menu Carte Panel Gestion

Réservation Se déconnecter

Accueil

Menu

Gestionnaire d'images

### Gestionnaire d'images

Choisissez des image (PNG, JPG)

Enregistrer





Aucun fichier sélectionné

Cliquer au dessus de l'image pour donner un ordre d'image sur la page d'accueil (entre 1 et 6)

Supprimer un image

Ordonner vos images pour la page d'accueil :

Ordonner



Le Quai Antique

Restaurant gastronomique

Chambery 73000  
Lorem Ipsum  
Tel: 06.00.00.00.00  
Mail du Chef: arnaud.michant@gmail.com

Lundi Mardi Mercredi Jeudi Vendredi Samedi Dimanche