



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Nom de naissance

- CHARLET

Nom d'usage

- CHARLET

Prénom

- Jessy

Adresse

- 152 Avenue Colonel Picot
Le Cercis A.
83100 TOULON

Titre professionnel visé

Développeur Web et Web Mobile

MODALITÉ D'ACCÈS :

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

DOSSIER PROFESSIONNEL (DP)

Sommaire

Exemples de pratique professionnelle

Développer la partie front-end d'une application web ou web mobile sécurisée	p.	4
- Installer et configurer son environnement de travail en fonction du projet web ou web mobile .	p.	4
- Maquetter des interfaces utilisateur web ou web mobile	p.	7
- Réaliser des interfaces utilisateur statiques web ou web mobile	p.	13
- Développer la partie dynamique des interfaces utilisateur web ou web mobile	p.	20
Développer la partie back-end d'une application web ou web mobile sécurisée	p.	
- Mettre en place une base de données relationnelle	p.	24
- Développer des composants d'accès aux données SQL et NoSQL	p.	27
- Développer des composants métier côté serveur	p.	
- Documenter le déploiement d'une application dynamique web ou web mobile	p.	
Titres, diplômes, CQP, attestations de formation (facultatif)	p.	-
Déclaration sur l'honneur	p.	-
Documents illustrant la pratique professionnelle (facultatif)	p.	-
Annexes (Si le RC le prévoit)	p.	-



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

EXEMPLES DE PRATIQUE PROFESSIONNELLE

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

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

- Exemple n°1* - Installer et configurer son environnement de travail en fonction du projet web ou web mobile

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

Pour ce projet personnel de Pokédex, créé dans le but de tester Angular, j'ai commencé par installer un IDE (*integrated development environment*), j'ai choisi Visual Studio Code pour sa popularité ainsi que pour ses nombreuses extensions. J'ai continué en installant Node.js et NPM (Node Package Manager) pour gérer les dépendances. J'ai ensuite installé Git pour gérer le versionning et pour finir j'ai installé le framework Angular.

2. Précisez les moyens utilisés :

J'ai commencé par installer VS Code en le téléchargeant ici : <https://code.visualstudio.com/>.

Pour Node.js j'ai téléchargé la dernière version ici : <https://nodejs.org/fr>

Grâce à NPM j'ai pu installer Angular avec la ligne de commande :

```
npm install -g @angular/cli
```

J'ai ensuite créé mon projet :

```
ng new pokdex
```

J'ai créé un repo Git

```
git init pokdex
```

et pour finir je peux lancer mon serveur pour voir mon application en local sur le port 4200 :

<http://localhost:4200>

```
ng serve
```



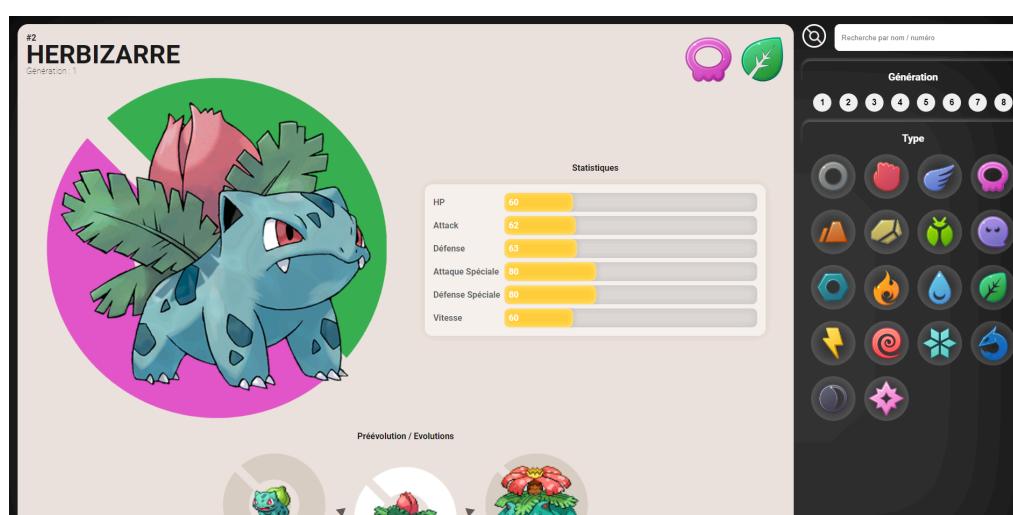
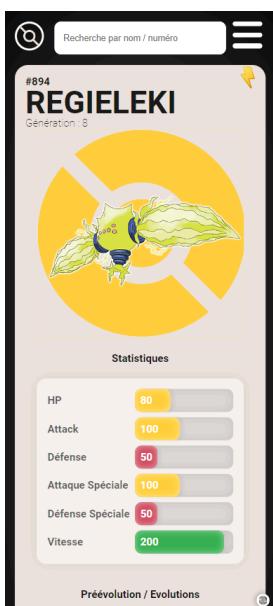
DOSSIER PROFESSIONNEL (DP)

rendu du projet Pokédex :

Version mobile



Version desktop



DOSSIER PROFESSIONNEL (DP)

3. Avec qui avez-vous travaillé ?

Seul

4. Contexte

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

Chantier, atelier, service ➤ *Projet personnel dans le but de découvrir un framework (Angular dans le cas présent)*

Période d'exercice ➤ Du : 03/06/2024 au : 10/06/2024

5. Informations complémentaires (facultatif)



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

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

Exemple n° 1 - Maquetter des interfaces utilisateur web ou web mobile

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

Au cours de mon stage chez "Innovation E-Santé Sud", j'ai été amené à développer une version mobile de leur site interne qui gère le temps de travail des agents ainsi que la réservation de bureaux. J'ai évalué auprès des agents, les besoins de chacun, les changements, améliorations et nouvelles fonctions pour cette application mobile. Ce qui m'a permis de constituer une user story (1). Je suis parti sur une PWA (progressive web app) afin que l'utilisation quotidienne soit plus simple et fluide. J'ai tout d'abord réalisé une maquette wireframe (2). Puis une fois celle-ci validée j'ai pu passer à une maquette plus aboutie (3). Pour ces deux dernières étapes j'ai utilisé Figma, qui m'a permis de présenter une maquette interactive.

2. Précisez les moyens utilisés :

(1) User story

En tant que	Je souhaite	Afin
user	Consulter un tableau de bord épuré avec mes informations importantes	d'avoir accès en un clin d'œil aux informations concernant ma journée et celle à venir.
user	Avoir accès au décompte complet de mes congés annuels, astreintes, heures supplémentaires	de pouvoir organiser mon emploi de temps au mieux
user	avoir un calendrier clair et intuitif et interactif	de rentrer simplement mes journée et avoir un aperçu général du mois en cours
user	pouvoir programmer une semaine type	de gagner du temps lors de l'enregistrement de mes journées

DOSSIER PROFESSIONNEL (DP)

user	avoir un plan du bâtiment lisible et interactif	afin d'identifier quels sont les bureaux libres et de pouvoir réserver celui que je souhaite simplement en un clic, et de voir ceux réservés par mes collègues
------	---	--

(2) Maquette wireframe

The wireframe illustrates two main views of the professional dossier:

- Ma semaine type** (Left View):
 - Titre**: At the top right.
 - Mise à jour**: A tooltip for the date navigation area.
 - Aperçu du jour**: A tooltip for the daily preview section.
 - Menu**: At the bottom right.
- Flex office** (Right View):
 - Titre**: At the top right.
 - Jours**: A tooltip for the date navigation area.
 - Plan**: A tooltip for the floor plan area.
 - étage**: A tooltip for the floor selection area.
 - Menu**: At the bottom right.

Both views include a header bar with buttons for **flex office**, **calendrier**, **semaine type** (highlighted in orange), and **profil**.



DOSSIER PROFESSIONNEL (DP)

The wireframe illustrates the DP application's interface, divided into two main sections: 'calendrier' (left) and 'Home' (right).

calendrier Section:

- Titre**: Title input field.
- Selection du moi**: Selection dropdown for 'moi'.
- Calendrier**: A large calendar view for June 2024. It includes a modal for modifying a specific day (Mercredi, 19 Juin 2024) which shows fields for Type (DH Présentiel), Arrival (09:00), Departure (17:00), and Breaks (0:30). It also displays a summary of 7H30 worked hours and validation checkboxes for agent and responsible.
- Jour à modifier**: Button to open the day modification modal.
- Formulaire de modification**: Description of the day modification form.

Home Section:

- Titre**: Title of the user (Jessy Charlet).
- Récapitulatif des compteurs**: Summary of counters (Delta -14h00, Astreintes 0, CA 0).
- Message d'alerte**: Alert message.
- Alerte**: Alert message indicating validation status.
- Aperçu du jour**: Preview of the day and tomorrow's info.
- Compteurs**: Quick overview of counters.
- Heures supp**: Extra hours table.
- Menu**: Main menu button.

(3) Maquette interactive

DOSSIER PROFESSIONNEL (DP)

Administrateur | 35 heures

Delta -70h00	Astreintes 0	CA 0
-----------------	-----------------	---------

Validation en retard !

Ma journée | 12/07

???	??	??	6-3
-----	----	----	-----

Demain | 13/07

???	??	??	6-5
-----	----	----	-----

Mes compteurs

CA	0
CA N-1	?

Astreinte 0

Jour	Valid.	Pris
?	?	?

Heures supp 0

Modifier mon mot de passe **Déconnexion**

The interface is a mobile application for managing professional time. It features a header 'DOSSIER PROFESSIONNEL (DP)'. On the left, a sidebar contains sections for 'Administrateur' (35 hours), a red warning box for 'Validation en retard !', 'Ma journée' (12/07), 'Demain' (13/07), and 'Mes compteurs' (CA 0, CA N-1 ?). It also includes an 'Astreinte' table and an 'Heures supp' counter. At the bottom are buttons for 'Modifier mon mot de passe' and 'Déconnexion', along with icons for a computer, calendar, heart, profile, and a logo.

Ma semaine type

Lundi				
→ 08:45	→ 17:45	⌚ 01:15	DH Prés.	07:45

Mardi				
→ 09:45	→ 16:15	⌚ 00:00	DH Prés.	06:30

Mercredi			
Type DH Télétravail	Arrivée 08 : 15	Départ 18 : 00	Pause 01 : 00

Jeudi							
⌚ 08:00	⌚ 17:00	⌚ 00:00	⌚ 06:30	📅	🕒	👤	sf

The right side of the interface shows a summary of the 'Ma semaine type' (My typical week) for Monday, Tuesday, and Wednesday. Each day has a list of activities with their start and end times. Wednesday's details are expanded, showing a dropdown for 'Type' (Telework), and specific times for 'Arrivée' (Arrival), 'Départ' (Departure), and 'Pause' (Break). Below this is a summary for Thursday with icons for arrival, departure, break, and a user profile.

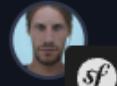
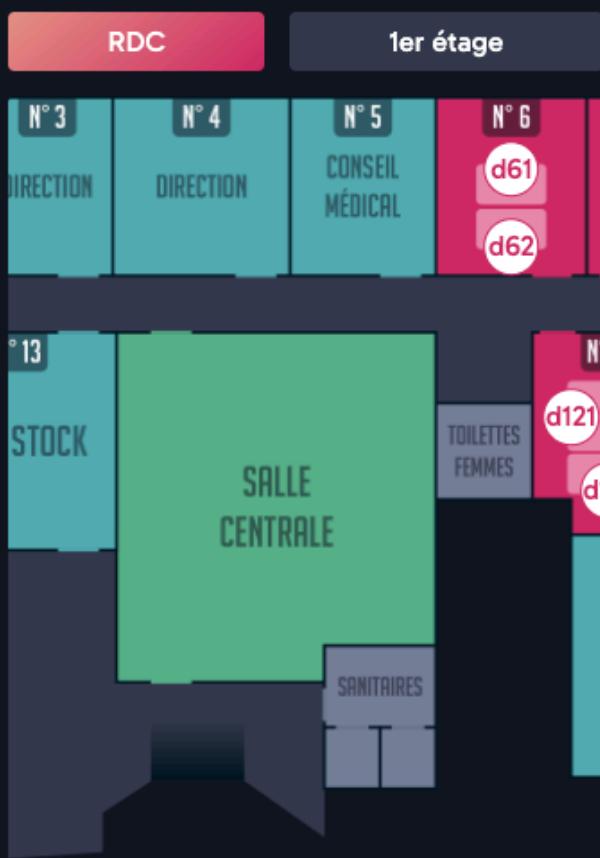


MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Flex Office

Ven. 12/06	Lun. 15/06	Mar. 16/06	Mer. 17/06	Jeu. 18/06
Ven. 19/06	Lun. 22/06	Mar. 23/06	Mer. 24/06	Jeu. 25/06



Mon calendrier

< JUIN 2024 >

Lun. Mar. Mer. Jeu. Ven. Total

Semaine 22

27	28	29	30	31	0h00
					-35h00

Semaine 22

3	4	5	6	7	0h00
					-35h00

Semaine 22

10	11	12	13	14	0h00
					-35h00

Semaine 22

17	18	19	20	21	37h30
DPH	DPH	DPH	DPH	DPH	2h30

Semaine 22

24	25	26	27	28	37h30
DPH	DPH	DPH	DPH	DPH	2h30



3. Avec qui avez-vous travaillé ?

DOSSIER PROFESSIONNEL (DP)

Seul

4. Contexte

Nom de l'entreprise, organisme ou association - *Innovation E-Santé Sud*

Chantier, atelier, service - *Développement d'une application mobile interne*

Période d'exercice - Du : 15/06/2024 au : 17/06/2024

5. Informations complémentaires (*facultatif*)

Activité-type 1



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

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

Exemple n° 1 ▶ Réaliser des interfaces utilisateur statiques web ou web mobile

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

Nouvelle compétence, nouveau projet. Cette fois-ci nous allons utiliser mon portfolio.

Il s'agit d'un site assez simple de présentation, il est disponible à cette adresse :

<https://jessy-charlet.students-laplateforme.io>

Vous pouvez d'ailleurs y consulter l'ensemble de mes projets (scolaires et personnels) ainsi que leurs repo Git.

La structure de base est assez basique, j'ai utilisé la sémantique recommandé en utilisant les balises <nav><main><section><aside> etc..., au vu du design que j'ai voulu créer pour ce site, avec menu fixe sur le côté gauche de l'écran (en haut pour la version mobile) je n'ai pas trouvé pertinent d'utiliser de <header> et de <footer>.

On peut noter la présence de links font.googleapis qui me permettent d'utiliser des polices différentes de celles de base, ainsi que le link stylesheet qui me permet d'ajouter la feuille de style CSS.

Enfin le script jquery qui me permet d'utiliser cette bibliothèque dans mon javascript (bien que vu la taille du JS dans le cas présent... c'était un peu overkill :p)

Structure de la page projets perso :

DOSSIER PROFESSIONNEL (DP)

```
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      <meta name="description" content="Présentation des projets personnels en développement web réalisés par Jessy Charlet">
8      <title>Jessy Charlet développeur web - Projets personnels</title>
9      <link rel="preconnect" href="https://fonts.googleapis.com">
10     <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
11     <link href="https://fonts.googleapis.com/css2?family=Radio+Canada+Big:ital,wght@0,400..700;1,400..700&display=swap"
12         rel="stylesheet">
13     <link href="https://fonts.googleapis.com/css2?family=Oswald:wght@200..700&display=swap" rel="stylesheet">
14     <link rel="stylesheet" href="./index.css" <?>time(); ?>
15  </head>
16
17 <body>
18 >    <nav class="navLeft"> ...
19 >    </nav>
20 >    <nav id="navSlide" class="navSlide"> ...
21 >    </nav>
22 >    <main> ...
23 >        </main>
24        <script src="https://code.jquery.com/jquery-3.6.4.min.js"></script>
25        <script src="./index.js"></script>
26    </body>
27
28 </html>
```

Dans le but d'améliorer au maximum l'accessibilité et le référencement, j'ai rempli les attributs alt de toutes mes , ainsi que les attributs aria-label des mes <a>

structure d'un projet :

```
44      <aside class="fullScreen pokédex">
45          <div class="gradient"></div>
46      </aside>
47      <section class="black">
48          
49          <div class="infos">
50              <div class="titreH2"><h2>Présentation du projet</h2></div>
51              <p>
52                  
53                  
54                  Dans l'optique d'approfondire mes connaissances sur l'interaction avec les API et de découvrir un
55                  framework (en l'occurrence
56                  Angular), j'ai réalisé un pokédex basé sur l'API pokébuild. J'ai décidé d'en faire une PWA
57                  (Progressif Web App) adapté à
58                  l'utilisation mobile.
59              </p>
60              <h3>Technologies utilisées</h3>
61              <div class="technos">
62                  <span class="html">HTML</span>
63                  <span class="css">CSS</span>
64                  <span class="angular">Angular</span>
65                  <span class="typescript">TypeScript</span>
66              </div>
67              <div class="links">
68                  <a target="blank" href="https://jessy-charlet.students-laplateforme.io/pokedex/">
69                      
70                  <a target="blank" href="https://github.com/Jessy-Charlet/pokedex">
71                      
72                  </div>
73              </div>
74      </section>
```



DOSSIER PROFESSIONNEL (DP)

Le Javascript est utilisé ici simplement pour gérer l'animation d'ouverture et de fermeture du menu burger

javascript :

```
1  $(document).ready(function () {
2      $('#menuBurger').on('click', () => {
3          $('#navSlide').toggleClass('navActive');
4          $('#menuBurger').toggleClass('menuBurgerActive');
5      });
6  });
```

Le reste des animation est en CSS pur, j'ai utilisé la règle @keyframe ainsi que la propriété animation pour gérer toutes les petites autres animation

Animations css :

```
370 .gradient {
371     position: absolute;
372     width: 100%;
373     height: 10vh;
374     bottom: 0;
375     left: 0;
376     background-image: linear-gradient(180deg, □rgb(10, 10, 10, 0), □rgb(10, 10, 10, 1));
377
378     img {
379         height: 10vh;
380         position: absolute;
381         top: 0%;
382         left: 50%;
383         translate: -50% -50%;
384         backdrop-filter: blur(5px);
385         border-radius: 50%;
386         animation: 3s ease-in-out infinite alternate upDown;
387     }
388 }
389
390 /*--- animations ---*/
391 @keyframes upDown {
392     from {
393         top: 0%;
394     }
395
396     to {
397         top: 100%;
398     }
399 }
```

On voit ici que les images contenues dans les balises de la classe gradient vont se déplacer à l'infinie de haut en bas avec un ralentissement en début et en fin de course.

DOSSIER PROFESSIONNEL (DP)

Pour la version mobile, j'ai utilisé les media queries afin de modifier l'affichage selon l'orientation de l'écran dans ce cas.

exemple des media queries:

```
588  /*--- Version mobile ---*/
589  @media (orientation: portrait) {
590
591      .buttonRed {
592          display: none;
593      }
594
595      .navLeft {
596          grid-template-columns: 80px 1fr 80px;
597          width: 100vw;
598          height: 75px;
599          border-right: none;
600          border-bottom: 1px solid black;
601          position: fixed;
602
603          .rotate270 {
604              writing-mode: horizontal-tb;
605              transform: rotate(0deg);
606
607              h1:hover {
608                  letter-spacing: 0em;
609              }
610
611              h2:hover {
612                  letter-spacing: 0.2em;
613              }
614          }
615      }
```

Ici nous pouvons constater que si l'écran est en mode portrait, les balises de la classe buttonRed disparaissent et que le menu de gauche passe en haut de l'écran (en tournant donc le texte dans le bon sens, en modifiant les proportions et les bordures)



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

Version mobile

JESSY CHARLET
DÉVELOPPEUR WEB

Présentation du projet

Etant un grand fan de TCG (Trading Card Game), j'ai voulu créer une petite application qui permettrait d'améliorer la "quality of life" des joueurs d'Altered. En effet la lisibilité en jeu n'étant pas évidente, Altered Counter permet d'identifier instantanément sur quel biome les joueurs s'affrontent et calcule instantanément le vainqueur de chaque expéditions. J'ai créé un mode solo (qui permet de voir l'entièreté des expéditions) ainsi qu'un mode Duo, où chaque joueur met son smartphone contre l'expédition à sa droite. Chaque joueur peut également choisir un thème afin d'afficher les couleurs de sa faction de son côté du terrain.

Technologies utilisées

HTML CSS Javascript Jquery

Globe icon and GitHub icon

Version desktop

JESSY CHARLET
DÉVELOPPEUR WEB

Altered COUNTER

Présentation du projet

Etant un grand fan de TCG (Trading Card Game), j'ai voulu créer une petite application qui permettrait d'améliorer la "quality of life" des joueurs d'Altered. En effet la lisibilité en jeu n'étant pas évidente, Altered Counter permet d'identifier instantanément sur quel biome les joueurs s'affrontent et calcule instantanément le vainqueur de chaque expéditions. J'ai créé un mode solo (qui permet de voir l'entièreté des expéditions) ainsi qu'un mode Duo, où chaque joueur met son smartphone contre l'expédition à sa droite. Chaque joueur peut également choisir un thème afin d'afficher les couleurs de sa faction de son côté du terrain.

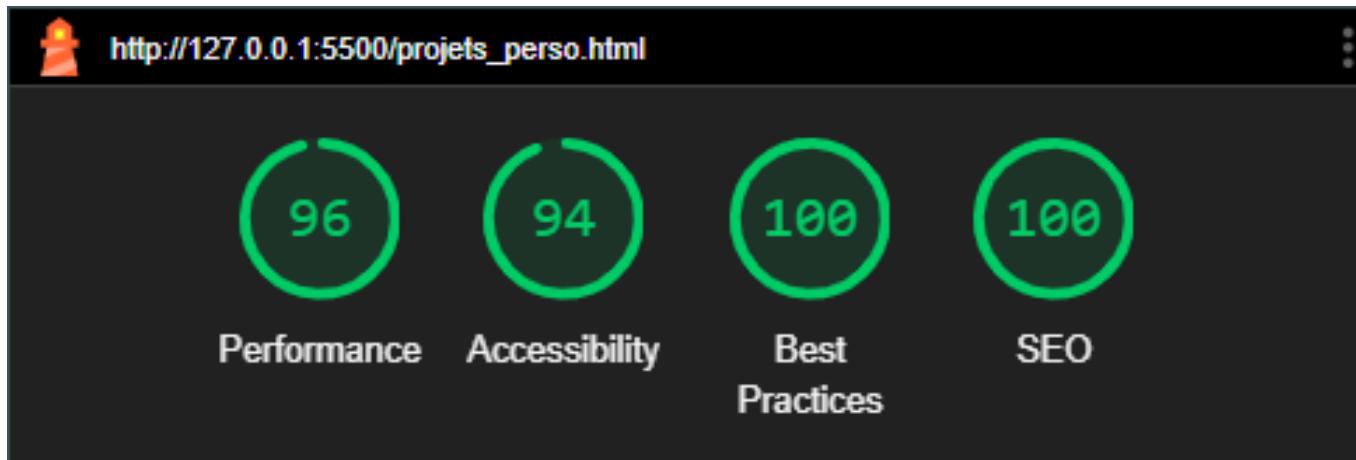
Technologies utilisées

HTML CSS Javascript Jquery

Globe icon and GitHub icon

2. Précisez les moyens utilisés :

Pour ce projet j'ai simplement utilisé VSCode avec le module Go Live pour accéder instantanément au rendu sur le port 5500. Pour vérifier l'accessibilité, les bonnes pratiques SEO et les performances j'ai utilisé l'outil Lighthouse de la console de développement de mon navigateur.



J'ai également utilisé la console pour simuler la version mobile et tester les media queries.

DOSSIER PROFESSIONNEL^(DP)

Pour le déployer j'ai utilisé l'hébergeur Plesk.

3. Avec qui avez-vous travaillé ?

Seul

4. Contexte

Nom de l'entreprise, organisme ou association ➤ *Projet perso*

Chantier, atelier, service ➤ *Portfolio*

Période d'exercice ➤ Du : *13/07/2024* au : *15/07/2024*

5. Informations complémentaires (*facultatif*)



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

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

Exemple n° 1 - Développer la partie dynamique des interfaces utilisateur web ou web mobile

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

Au cours de mon apprentissage, j'ai dû créer un memory uniquement en php vanilla. Le but du jeu étant de retrouver les paires de cartes faces cachées. Le cahier des charge était donc :

- A chaque partie, les cartes doivent être répartis aléatoirement.
- Lorsque l'on retourne une carte face visible :
 - Si aucune autre carte n'est retournée face visible (hors paires validées), elle reste face visible.
 - Si une autre carte est retournée face visible et qu'elle est identique, la paire est validée et les deux cartes restent face visible pour toute la partie.
 - Si une autre carte est retournée face visible et qu'elles sont différentes, alors les deux sont remises face cachée.
- Une fois toutes les paires retrouvées, un score est donné au joueur en fonction du temps qu'il a mis et de son nombre de tentatives.

Ce n'était pas demandé, mais j'en ai fait une PWA, j'ai trouvé que ça se prêtait bien pour une application mobile.

La classe qui gère les cartes :

```
7  class Card
8  {
9      public $id;
10     public $visible;
11     public $value;
12     public $even;
13     public $art;
14     public function __construct($card = null)
15     {
16         $this->id = $card->id ?? 0;
17         $this->visible = $card->visible ?? false;
18         $this->even = $card->even ?? false;
19         $this->art = $card->art ?? 0;
20         $this->value = $card->value ?? 0;
21     }
22 }
23 }
```

DOSSIER PROFESSIONNEL (DP)

On peut voir ici ses différentes variables publiques tels que
\$id -> int : son id unique
\$visible -> bool : indiquant si elle est face visible ou cachée
\$value -> int : la valeur de la carte (pour pouvoir comparer les paires)
\$even -> bool : indiquant si la paire est validée ou non
\$art -> string : le lien de l'image du recto de la carte
et enfin son constructeur.

Le lancement de partie :

```
28     /** Lancement de la partie */
29     if (isset($_POST["play"])) {
30         session_unset();
31         $cards = setcards(6);
32         shuffle($cards);
33         $_SESSION["try"] = 0;
34         $_SESSION["even"] = 0;
35         $_SESSION["click"] = 0;
36         $_SESSION["play"] = true;
37         $_SESSION["return"] = 0;
38         $_SESSION["cards"] = $cards;
39         $_SESSION["returnF"] = array(0,0);
40         $_SESSION["returnS"] = array(0,0);
41         $_SESSION["start"] = date("YmdHis");
42     }
```

Celui-ci se déclenche lorsque le joueur clique sur play, ce qui envoie un post qui est lu au chargement de la page index, celui-ci supprime la session existante (si il y a eu une partie avant), utilise la fonction setCards (ci-dessous) avec un attribut de 6 (pour une partie à 6 paires). Elle va nous retourner un tableau avec toutes les cartes dedans. La fonction interne de PHP shuffle va mélanger ce tableau afin de rendre aléatoire le placement des cartes. Et enfin on enregistre en session toutes les infos dont on aura besoin à chaque recharge de la page.



DOSSIER PROFESSIONNEL (DP)

```
5  function setCards($number)
6  /** Cr ation des cartes */
7  {
8      $a = 1;
9      $cards = array();
10 {
11     for ($i = 1; $i <= $number; $i++) {
12         $cards[$a] = new Card();
13         $cards[$a]->value = $i;
14         $cards[$a]->id = $a;
15         $cards[$a]->art = "<img src='./assets/card-" . $i . ".png' />";
16         $a++;
17         $cards[$a] = new Card();
18         $cards[$a]->value = $i;
19         $cards[$a]->id = $a;
20         $cards[$a]->art = "<img src='./assets/card-" . $i . ".png' />";
21         $a++;
22     }
23     return $cards;
}
```

La fonction setCards dont nous avons parl  au dessus qui va cr er un tableau d'objets dont on va initialiser les valeurs voulu (le \$a qui va s'incr emerter de 1   chaque carte pour les infos propres   celle ci, et le \$i qui quand   lui sera commun   la paire (valeur et image).

```
87 function evenCounter()
88 /** D etecte Les paire et les enregistres */
89 {
90     if ($_SESSION["returnF"][0] != 0 and $_SESSION["returnF"][1] == $_SESSION["returnS"][1]) {
91         $_SESSION["even"]++;
92         $f = $_SESSION["returnF"][1];
93         foreach ($_SESSION["cards"] as $card) {
94             if ($card -> value == $f) {
95                 $card -> even = true;
96             }
97         }
98     }
99 }
```

La fonction evenCounter calcule comme son nom l'indique le nombre de paires. Si la valeur session 'visible' de returnF (first) (premi re carte retourn e) n'est pas 0 (donc visible) et que returnF et de returnS (second) (deuxi me carte retourn e) sont  gales, alors pour chaque carte dont la valeur est  gale   celle de returnF on modifie sa valeur even pour la passer   true (et ainsi valider la paire en session)

DOSSIER PROFESSIONNEL (DP)

Version mobile



Version desktop





DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI



2. Précisez les moyens utilisés :

Ce projet étant en PHP vanilla, j'ai simplement utilisé VSCode et Laragoon.

3. Avec qui avez-vous travaillé ?

Seul

4. Contexte

Nom de l'entreprise, organisme ou association ▶ *Centre de formation : La_plateforme*

Chantier, atelier, service ▶ *exercise : memory*

Période d'exercice ▶ Du : 17/01/2024 au : 22/01/2024

DOSSIER PROFESSIONNEL (DP)

5. Informations complémentaires (facultatif)

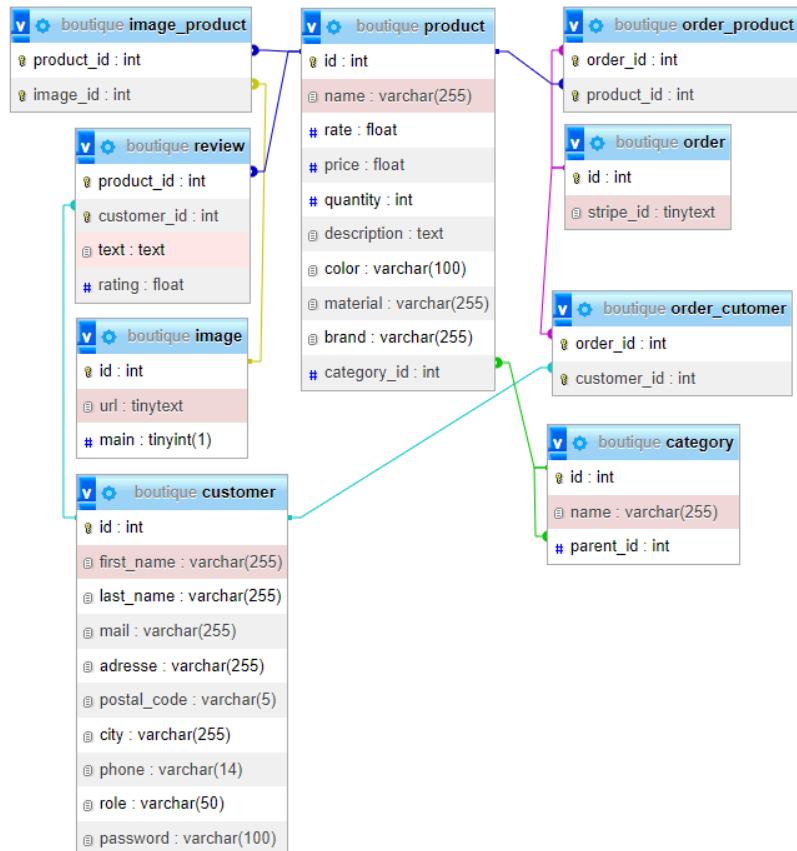
Développer la partie back-end d'une application web ou web mobile sécurisée

Activité-type 1

Exemple n° 1 - Mettre en place une base de données relationnelle

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

Pour cet exemple nous allons prendre le projet boutique que j'ai eu à réaliser durant la formation. Il a tout d'abord fallu analyser les besoins et créer un MCD (Modèle Conceptuel des Données) afin de définir les entités et les relations de manière claire et compréhensible. Puis de le transformer en MLD (Modèle Logique des Données) plus proche de la structure de la base de données relationnelle avec des clés primaires et étrangères





MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

On a dans cet exemple réalisé sous phpMyAdmin les relations entre les tables qui sont matérialisées par les traits colorés avec suivant le type de relation des accroches différentes. Par exemple un customer peut être l'auteur de plusieurs review de produit, mais une review ne peut avoir qu'un seul customer comme auteur. Il s'agit donc d'une relation oneToMany. En revanche, un produit peut avoir plusieurs catégories et une catégorie peut avoir plusieurs produits. Dans ce cas, il s'agit d'une relation manyToMany. On peut également voir une liaison particulière au niveau des catégories. En effet, celle-ci possède une relation récursive, c'est-à-dire qu'elle entretient une relation avec elle-même. Une catégorie peut être liée à une autre catégorie ce qui en ferait une sous-catégorie. C'est assez rare mais utile dans ce cas de figure car on pourra utiliser l'instruction SQL "WITH RECURSIVE" afin de récupérer toutes les sous-catégories d'une catégorie par exemple. On peut également noter la présence de tables de jointures. Elles sont reconnaissables par leur nom qui comporte celui des deux tables jointes. Par exemple, produit_order qui fait la jointure entre product et order. Sa clé primaire est la combinaison des clés étrangères liées aux tables qu'elle joint. Ces tables permettent d'éviter la redondance des données, de modéliser facilement les relations manyToMany et elles maintiennent l'intégrité référentielle entre les tables associées.

2. Précisez les moyens utilisés :

Pour le MCD je l'ai fait à la main sur une feuille en cour, j'aurais pu le faire sur MySQL Workbench par exemple, mais cette base de donnée n'étant pas extrêmement complexe, c'était suffisant.
J'ai utilisé phpMyAdmin pour la création de base de données.
J'ai ensuite exporté celle-ci pour la copier sur plesk afin de mettre le projet en ligne.

3. Avec qui avez-vous travaillé ?

Alina Yefimova et Ismael Lebrun

4. Contexte

Nom de l'entreprise, organisme ou association ▶

Centre de formation : La_plateforme

Chantier, atelier, service ▶

projet boutique

Période d'exercice

▶ Du : 10/04/2024 au : 21/04/2024

DOSSIER PROFESSIONNEL (DP)

5. Informations complémentaires (*facultatif*)

Dictionnaire de données

category

Column	Type	Null	Default	Comments				
id (<i>Primary</i>)	int	No						
name	varchar(255)	No						
parent_id	int	No						
Indexes								
Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	
parent_id	BTREE	No	No	parent_id	0	A	No	

customer

Column	Type	Null	Default	Comments				
id (<i>Primary</i>)	int	No						
first_name	varchar(255)	No						
last_name	varchar(255)	No						
mail	varchar(255)	No						
adresse	varchar(255)	No						
postal_code	varchar(5)	No						
city	varchar(255)	No						
phone	varchar(14)	No						
role	varchar(50)	No						
password	varchar(100)	No						
Indexes								
Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	1	A	No	

image

Column	Type	Null	Default	Comments				
id (<i>Primary</i>)	int	No						
url	tinytext	No						
main	tinyint(1)	No						
Indexes								
Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	43	A	No	

image_product

Column	Type	Null	Default	Comments				
product_id (<i>Primary</i>)	int	No						
image_id (<i>Primary</i>)	int	No						
Indexes								
Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	product_id	7	A	No	
image_id	BTREE	No	No	image_id	15	A	No	
product_id	BTREE	No	No	product_id	7	A	No	
product_id_2	BTREE	No	No	product_id	7	A	No	
				image_id	15	A	No	



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Développer la partie back-end d'une application web ou web mobile sécurisée

Activité-type 1

Exemple n° 1 - Développer des composants d'accès aux données SQL et NoSQL

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

Dans le cadre du projet Game Throne, la boutique en ligne spécialisée dans les chaises gaming, le développement des composants d'accès aux données SQL constitue une étape cruciale. Ces composants jouent un rôle fondamental en garantissant une interaction fluide et efficace entre l'application web et la base de données relationnelle qui stocke toutes les informations essentielles, telles que les détails des produits, les commandes des clients, les avis des utilisateurs et les configurations personnalisées.

Pour concevoir ces composants, nous utiliserons des requêtes SQL écrites manuellement afin d'interagir directement avec le Système de Gestion de Base de Données Relationnel MySQL. Cette approche offre un contrôle granulaire sur les opérations de base de données, permettant une optimisation spécifique et une compréhension approfondie des transactions SQL.

Les principaux modules de ces composants incluront :

Gestion des Produits :

Insertion : Insertion de nouvelles chaises gaming dans la base de données via des requêtes INSERT INTO.

Mise à jour : Mise à jour des descriptions, des prix et des options de personnalisation avec des requêtes UPDATE.

Recherche : Recherche et récupération de produits en fonction de divers critères (marque, prix, ergonomie) en utilisant des requêtes SELECT.

Gestion des Utilisateurs :

Enregistrement : Enregistrement des nouveaux clients avec des requêtes INSERT INTO.

Authentification : Vérification des informations de connexion avec des requêtes SELECT.

Gestion des avis : Insertion des avis des utilisateurs avec des requêtes INSERT INTO et récupération

DOSSIER PROFESSIONNEL (DP)

d'avis avec des requêtes SELECT.

Gestion des Commandes :

Traitement des commandes : Insertion de nouvelles commandes avec des requêtes INSERT INTO et mise à jour de leur statut avec des requêtes UPDATE.

Suivi des transactions : Récupération des détails des commandes et des transactions avec des requêtes SELECT.

Gestion des Stocks :

Suivi en temps réel : Mise à jour des niveaux de stock avec des requêtes UPDATE et récupération du stock actuel avec des requêtes SELECT.

Alertes de stock : Détection des produits à faible stock avec des requêtes SELECT et génération d'alertes.

Gestion des Requêtes :

Requêtes complexes : Exécution de requêtes complexes pour les rapports de ventes, les tendances d'achat, et les analyses de marché.

Chaque composant sera conçu en respectant les principes de l'ACID (Atomicité, Cohérence, Isolation, Durabilité) pour assurer la fiabilité des transactions et la cohérence des données. Des procédures de sauvegarde et de restauration seront également mises en place pour protéger les données contre les pertes ou les corruptions. En outre, l'implémentation de mesures de sécurité, telles que le chiffrement des données sensibles (hachage des mots de passe) et l'authentification robuste, garantira la protection des informations des clients et la conformité aux réglementations en vigueur.

PDO (PHP Data Objects) sera utilisé pour gérer les interactions avec la base de données SQL. PDO offre une interface unifiée et sécurisée pour accéder à différentes bases de données, garantissant une gestion efficace et sécurisée des données critiques du site web.



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

```
2 class Database
3 {
4     private static $servername = 'localhost';
5     private static $username = 'root';
6     private static $password = '';
7     private static $BDD = 'boutique';
8     private static $conn = null;
9
10    public function __construct()
11    {
12        die('Init function is not allowed');
13    }
14    public static function connect()
15    { //fonction de connexion à La BDD
16        if (null == self::$conn) { //si la connexion est nulle
17            try { //on essaie de se connecter
18                self::$conn = new PDO("mysql:host=" . self::$servername . ";" . "dbname=" . self::$BDD, self::$username, self::$password);
19                self::$conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
20            } catch (PDOException $e) {
21                die($e->getMessage());
22            }
23        }
24        return self::$conn;
25    }
26    public static function disconnect()
27    {
28        self::$conn = null;
29    }
```

La classe Database gère tout ce qui touche à la base de données, elle contient par exemple la fonction connect, qui permet de se connecter à la base de données avec PDO afin de faire des requêtes par la suite.

Exemple avec la fonction ci-dessous : getProductById() qui prend en paramètre l'id du produit que l'on souhaite récupérer.

On utilise ici le try / catch afin de pouvoir récupérer l'erreur s'il y a un problème avec la connexion à la base de données.

Puis on prépare la requête SQL avec des attributs que l'on va hydrater grâce à bindParam() et enfin exécuter la requête.

On récupère les données qu'on place dans un tableau à l'intérieur de la variable \$product, on ferme la connexion et on retourne \$product.

DOSSIER PROFESSIONNEL (DP)

```
112 // Récupération des produits par ID
113 public static function getProductById($id)
114 {
115     try {
116         // Connection à La BDD
117         $conn = Database::connect();
118
119         // Préparation de La requête SQL pour récupérer l'ensemble produits
120         $stmt = $conn->prepare("SELECT * FROM product WHERE id=:id");
121         $stmt->bindParam(':id', $id);
122         $stmt->execute();
123
124         // Execution de la requête SQL
125         $product = $stmt->fetch(PDO::FETCH_ASSOC);
126
127         // Fermeture de la connection
128         $conn = null;
129
130         // return["id"] pour l'ID du produit
131         // return["name"] pour le nom du produit
132         // return["rate"] pour la note du produit
133         // return["price"] pour le prix du produit
134         // return["quantity"] pour la quantité du produit
135         // return["description"] pour la description complète du produit
136         // return["color"] pour la couleur du produit
137         // return["material"] pour la matière du produit
138         // return["brand"] pour la marque du produit
139         // return["category_id"] pour la catégorie du produit
140         return $product;
141     } catch (PDOException $e) {
142         return $e;
143     }
144 }
```

2. Précisez les moyens utilisés :

Avec du PHP en POO (Programmation Orientée Objet), nous avons utilisé PDO afin de créer une connexion à la base de données pour y préparer les requêtes SQL paramétrées. pour nous protéger contre les injections SQL.

3. Avec qui avez-vous travaillé ?

Alina Yefimova et Ismael Lebrun

4. Contexte



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Nom de l'entreprise, organisme ou association ➔ *Centre de formation : La_plateforme*

Chantier, atelier, service ➔ *projet boutique*

Période d'exercice ➔ Du : *10/04/2024* au : *21/04/2024*

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type

Développer la partie back-end d'une application web ou web mobile sécurisée

Exemple n° 1 - Développer des composants métier côté serveur

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

Toujours pour le projet Game Throne, j'ai codé l'affichage des produits en fonction de différents filtres (par exemple la couleur et un prix compris dans une fourchette définie). Ces filtres sont accessibles séparément, mais peuvent également se combiner.

```
1 $(document).ready(function () {
2   $("#filtersSup").hide();
3   sessionStorage.clear('color');
4   function afficherProduits(products) {
5     const liste = document.createElement('div');
6     liste.classList.add("productsGrid");
7     if ((0 < products.length)) {
8       for (var i = 0; i < products.length; i++) {
9         // Container
10        const listItem = document.createElement('div');
11        listItem.classList.add("product");
12        // Nom du produit
13        const titleElement = document.createElement('div');
14        titleElement.classList.add("productName");
15        titleElement.textContent = products[i]["name"];
16        // Lien du produit
17        const urlElement = document.createElement('a');
18        urlElement.href = `/produit?id=${products[i]["id"]}`;
19        // Image du produit
20        const imageElement = document.createElement('img');
21        const imageUrl = "./assets/img/products/" + products[i]["img"];
22        imageElement.src = imageUrl;
23        imageElement.alt = products[i]["name"];
24        // Prix du produit
25        const divElement = document.createElement('div');
26        divElement.classList.add("productDiv");
27        const priceElement = document.createElement('div');
28        priceElement.classList.add("productPrice");
29        priceElement.textContent = products[i]["price"] + " €";
30        // Note du produit
31        const ratingElement = document.createElement('div');
32        ratingElement.classList.add("productRate");
33        ratingElement.innerHTML = parseFloat(products[i]["rate"]).toFixed(1) + "<img src='./assets/img/star.png' />";
34        // Ajouter les éléments à La Liste
35        listItem.appendChild(urlElement);
36        urlElement.appendChild(imageElement);
37        urlElement.appendChild(titleElement);
38        urlElement.appendChild(divElement);
39        divElement.appendChild(priceElement);
40        divElement.appendChild(ratingElement);
41        liste.appendChild(listItem);
42      };
43      $("#filtersProducts").html(liste);
44    } else {
45      $("#filtersProducts").html("<div class='oups'>Oups... Aucun produit ne correspond à cette recherche :(</div>\"");
46    }
47  }
```

la fonction asynchrone recherche() va fetch() le controller recherche.php qui va effectuer une requête SQL avec les paramètres passés en argument puis va récupérer la réponse sous format json et ainsi



DOSSIER PROFESSIONNEL (DP)

appeler la fonction afficherProduits() que l'on à vu juste au dessus avec le résultat en argument. Nous avons ensuite plusieurs fonctions anonymes.

Une qui se déclenche lors du choix de couleur et qui va la passer dans le local.storage afin de pouvoir l'utiliser plus tard, et lancer une la fonction recherche() avec le resultat en argument.

Une autre qui se déclenche lors de la modification du slider de prix et qui va gérer celui ci ainsi que ses valeurs puis lancer une recherche() avec ,si une couleur est stockée en local storage, la couleur + la fourchette de prix, dans le cas contraire la recherche() avec uniquement la fourchette de prix.

```
48 	async function recherche(mini = 1, maxi = 1000, color = "all") {
49  	const reponse = await fetch("../controller/php/recherche.php?color=" + color + "&mini=" + mini + "&maxi=" + maxi);
50  	const products = await reponse.json();
51  	afficherProduits(products);
52 }
53 $("#noir, #blanc, #rouge, #jaune, #vert, #bleu, #violet, #gris").on("click", function () {
54  	sessionStorage.setItem('color', this.id);
55  	recherche($("#slider-range").slider("values", 0), $("#slider-range").slider("values", 1), sessionStorage.getItem('color'));
56  	$("#filtersSup").html("✖ Supprimer le filtre <span class='bold'>" + this.id + "</span>");
57  	$("#filtersSup").slideDown();
58 });
59 $(function () {
60  	$("#slider-range").slider({
61    	range: true,
62    	min: 1,
63    	max: 1000,
64    	values: [1, 1000],
65    	slide: function (event, ui) {
66      	$("#amount").val(ui.values[0] + " € - " + ui.values[1] + " €");
67      	if (sessionStorage.getItem('color')) {
68        	recherche($("#slider-range").slider("values", 0), $("#slider-range").slider("values", 1), sessionStorage.getItem('color'));
69       } else {
70        	recherche($("#slider-range").slider("values", 0), $("#slider-range").slider("values", 1));
71       }
72     }
73   });
74  	$("#amount").val($("#slider-range").slider("values", 0) +
75  	" € - " + $("#slider-range").slider("values", 1) + " €");
76 });
77 $(function () {
78  	if (sessionStorage.getItem('color')) {
79    	$("#filtersSup").slideDown();
80   }
81  	$("#filtersSup").on("click", function () {
82    	sessionStorage.clear('color');
83    	recherche($("#slider-range").slider("values", 0), $("#slider-range").slider("values", 1));
84    	$(this).slideUp();
85   });
86 }
87 });
88 recherche();
89 })
```

voici le contrôleur dont on parlait un peu plus haut, celui qui est appelé par le fetch du JS pour afficher les produits selon les différents filtres.

On peut voir au require qu'il à besoin de la class database pour fonctionner

On déclare \$selec qui est un tableau et qui sera la valeur retournée à la résolution du code, puis si il n'y a pas color dans l'URL on effectue une requête SQL via PDO pour récupérer tous les produits. Par contre

DOSSIER PROFESSIONNEL (DP)

si il y a color dans l'URL:

Si sa valeur est all, alors la requête ne prend en compte que la fourchette de prix.

Dans tous les autres cas, la requête prend en compte sa valeur en plus de la fourchette de prix.

Une fois cette requête préparée, on l'exécute et on place tout dans la \$product. Pour finir on appelle la fonction getImg() qui est tout en haut du code afin d'effectuer une requête SQL pour chaque produit dans le but d'y récupérer toutes ses images dans la table image qui est liée par une table de jointure image_product à la table product comme nous l'avons vu plus haut.

On termine par push \$product dans le tableau \$selec, on ferme la connexion et on echo le \$selec en format Json. Ce qui sera récupéré par le JavaScript et affiché dans le DOM.

```
3 require('./classes/Database.class.php');
4 $conn = Database::connect();
5 $selec = [];
6 function getImg($products)
7 {
8     global $selec;
9     global $conn;
10    foreach ($products as $product) {
11        try {
12            $sql = $conn->prepare("SELECT url FROM image
13                INNER JOIN image_product
14                WHERE image.id = image_product.image_id AND image_product.product_id = " . $product['id'] . " and image.main = 1");
15            $sql->execute();
16            $img = $sql->fetch(PDO::FETCH_ASSOC);
17            $product["img"] = $img["url"];
18            array_push($selec, $product);
19        } catch (PDOException $e) {
20            http_response_code(500);
21            echo json_encode(["msg" => "Erreur de connexion à la base de données: " . $e->getMessage()]);
22            exit();
23        }
24    }
25 }
26 if (isset($_GET["color"])) {
27     try {
28         switch ($_GET["color"]) {
29             case "all":
30                 $stmt = $conn->prepare("SELECT * FROM product WHERE price >= :mini AND price <= :maxi");
31                 break;
32             default:
33                 $stmt = $conn->prepare("SELECT * FROM product WHERE price >= :mini AND price <= :maxi AND color= :color");
34                 $stmt->bindParam(':color', $_GET['color']);
35                 break;
36         }
37         $stmt->bindParam(':mini', $_GET['mini']);
38         $stmt->bindParam(':maxi', $_GET['maxi']);
39         $stmt->execute();
40         $products = $stmt->fetchAll(PDO::FETCH_ASSOC);
41         getImg($products);
42     } catch (PDOException $e) {
43         http_response_code(500);
44         echo json_encode(["msg" => "Erreur de connexion à la base de données: " . $e->getMessage()]);
45         exit();
46     }
47 } else {
48     try {
```



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

```
49     $stmt = $conn->prepare("SELECT * FROM product");
50     $stmt->execute();
51     $products = $stmt->fetchAll(PDO::FETCH_ASSOC);
52     getImg($products);
53 } catch (PDOException $e) {
54     http_response_code(500);
55     echo json_encode(["msg" => "Erreur de connexion à la base de données: " . $e->getMessage()]);
56     exit();
57 }
58 $conn = null;
59 header('Content-Type: application/json');
60 echo json_encode($selec);
```

2. Précisez les moyens utilisés :

J'ai utilisé PHP et Javascript/Jquery pour les langages de programmation, Alt router pour la gestion des routes, la classe Database (vu dans le sujet précédent) pour la connexion à la base de donnée avec PDO pour les requêtes SQL.

3. Avec qui avez-vous travaillé ?

Alina Yefimova et Ismael Lebrun pour le projet, mais cette partie a été réalisée uniquement par mes soins.

4. Contexte

Nom de l'entreprise, organisme ou association ▶

Centre de formation : La_plateforme

Chantier, atelier, service ▶

projet : boutique

Période d'exercice

▶ Du : 10/04/2024 au : 21/04/2024

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL ^(DP)





DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

Développer la partie back-end d'une application web Activité-type 2 ou web mobile sécurisée

Exemple n° 1 ▶ Documenter le déploiement d'une application dynamique web ou web mobile

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

Pour ce dernier exemple, nous allons utiliser le livre d'or que j'ai créé au cours de la formation. Pour le déploiement du site j'ai utilisé l'hébergeur Plesk dont un compte nous a été fourni par l'école. J'ai utilisé le File Manager de Plesk dans lequel j'ai créé un dossier "livre-or pour ce site qui sera accessible à l'adresse <https://jessy-charlet.students-laplateforme.io/livre-or/>

The screenshot shows the Plesk File Manager interface. The left sidebar shows a tree view of the file structure under 'Home directory / httpdocs'. The main area displays a list of files and sub-directories. A search bar at the top right is set to 'Search in filename'. The list includes:

Date	Size	Permissions	Last Modified By	Action
Jan 17, 2024 11:07 AM	9.8 KB	r-- r-- r--	jessy-charlet psacln	More
Jan 17, 2024 09:57 AM	7.5 KB	r-- r-- r--	jessy-charlet psacln	More
Dec 20, 2023 02:22 PM		rwx r-x r-x	jessy-charlet psacln	More
July 1, 2024 05:36 PM		rwx r-x r-x	jessy-charlet psacln	More
Jan 15, 2024 11:53 AM		rwx r-x r-x	jessy-charlet psacln	More
Feb 7, 2024 01:07 PM		rwx r-x r-x	jessy-charlet psacln	More
July 15, 2024 08:52 PM		rwx r-x r-x	jessy-charlet psacln	More
June 12, 2024 02:33 PM		rwx r-x r-x	jessy-charlet psacln	More
Mar 11, 2024 04:54 PM		rwx r-x r-x	jessy-charlet psacln	More
Dec 20, 2023 02:42 PM		rwx r-x r-x	jessy-charlet psacln	More
Mar 22, 2024 03:31 PM		rwx r-x r-x	jessy-charlet psacln	More
Dec 20, 2023 02:51 PM		rwx r-x r-x	jessy-charlet psacln	More
Jan 17, 2024 09:58 AM		rwx r-x r-x	jessy-charlet psacln	More
July 15, 2024 10:10 PM		rwx r-x r-x	jessy-charlet psacln	More
July 10, 2024 11:48 AM		rwx r-x r-x	jessy-charlet psacln	More
April 24, 2024 10:18 AM		rwx r-x r-x	jessy-charlet psacln	More

J'y ai upload la page index.php ainsi que le reste du site (les autres pages, le css, les images...) En utilisant leur système de Databases j'ai créer une nouvelle base de donnée

DOSSIER PROFESSIONNEL (DP)

Add a Database

General

Database name * jessy-charlet_livreor
Database server localhost:3306 (default for MariaDB, v5.5.68)
Related site jessy-charlet.students-laplateforme.io

Dans phpMyAdmin j'ai exporté ma base de données local

The screenshot shows the phpMyAdmin interface with the 'livreor' database selected. A modal window titled 'Exporting tables from "livreor" database' is open. In the 'Export method:' section, 'Quick - display only the minimal options' is selected. In the 'Format:' section, 'SQL' is chosen. At the bottom right of the modal is a blue 'Export' button.

Je l'ai importé dans celle que je venais de créer sous Plesk

The screenshot shows the Plesk interface with several databases listed on the left: 'jessy-charlet_livreor', 'jessy-charlet_module-connexion', and 'jessy-charlet_reservation_salles'. A central modal window titled 'Import Database Dump' is open. It has two tabs: 'Upload' (selected) and 'Import'. Under 'Upload', there is a file input field labeled 'Choisir un fichier' with the message 'Aucun fichier choisi'. Below it is a dropdown menu 'Root directory' showing a list of files and folders: '.composer', '.pki', '.revitum_antivirus_cache', '.ssh', and '.trash'. A checkbox 'Recreate the database' is checked. At the bottom of the modal are 'OK' and 'Cancel' buttons.

Il ne me restait plus qu'à modifier mes variables :
\$sql_serveur , \$sql_utilisateur , \$sql_motDePasse et \$sql_baseDeDonnees
dans mon fichier PHP afin qu'elles collent avec la nouvelle base de données Plesk.

Le livre d'or est maintenant déployé.



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

2. Précisez les moyens utilisés :

J'ai utilisé Plesk pour l'hébergement du site et de la base de donnée, phpMyAdmin pour exporter ma base de données locale.

3. Avec qui avez-vous travaillé ?

seul

4. Contexte

Nom de l'entreprise, organisme ou association ▶ *centre de formation : La_plateforme*

Chantier, atelier, service ▶ *exercice : livre d'or*

Période d'exercice ▶ Du : 18.12.2023 au : 21/12/2023

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Titres, diplômes, CQP, attestations de formation

(facultatif)



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

DOSSIER PROFESSIONNEL (DP)

Déclaration sur l'honneur

Je soussignée CHARLET Jessy ,

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

Fait à TOULON

le 18/07/2024

pour faire valoir ce que de droit.

Signature :

A handwritten signature in black ink, appearing to read "Jessy Charlet".



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

Documents illustrant la pratique professionnelle

(facultatif)

DOSSIER PROFESSIONNEL^(DP)

ANNEXES

(Si le RC le prévoit)