



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Nom de naissance

- SUTHAHARAN

Nom d'usage

-

Prénom

- VISHNUGOPY

Adresse

- 16 RUE DEGAS ERMONT 95120

Titre professionnel visé

Concepteur développeur d'applications

MODALITÉ D'ACCÈS :

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

Présentation du dossier

DOSSIER PROFESSIONNEL (DP)

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

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

Pour rédiger le DP, le candidat peut être aidé par un formateur ou par un accompagnateur VAE.

Il est consulté par le jury au moment de la session d'examen.

Pour prendre sa décision, le jury dispose :

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

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

Ce dossier comporte :

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

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

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



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Sommaire

Exemples de pratique professionnelle

Intitulé de l'activité-type n° 1 : Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité	p. 6
» Intitulé de l'exemple n° 1 : Maquetter une application	p. 6-8
» Intitulé de l'exemple n° 2 : Développer une interface utilisateur de type desktop (appli)	p. 9-12
» Intitulé de l'exemple n° 3 : Développer des composants d'accès aux données (orm)	p. 13
» Intitulé de l'exemple n° 4 : Développer la partie front-end d'une interface utilisateur web	p. 15
» Intitulé de l'exemple n° 5 : Développer la partie back-end d'une interface utilisateur web express	p. 14
Intitulé de l'activité-type n° 2 : Concevoir et développer la persistance des données en intégrant les recommandations de sécurité	p. 18
» Intitulé de l'exemple n° 1 : Concevoir une base de données	p. 18
» Intitulé de l'exemple n° 2 : Mettre en place une base de données	p. 19
» Intitulé de l'exemple n° 3 : Développer des composants dans le langage d'une base de données	p. 20
Intitulé de l'activité-type n° 3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité	p. 22
» Intitulé de l'exemple n° 1 : Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement	p. 22
» Intitulé de l'exemple n° 2 : Concevoir une application	p. 24
» Intitulé de l'exemple n° 3 : Développer des composants métier	p. 26
» Intitulé de l'exemple n° 4 : Construire une application organisée en couches	p. 28
» Intitulé de l'exemple n° 5 : Développer une application mobile	p. 30
» Intitulé de l'exemple n° 6 : Préparer et exécuter les plans de tests d'une application	p. 32
» Intitulé de l'exemple n° 7 : Préparer et exécuter le déploiement d'une application	p. 34
Titres, diplômes, CQP, attestations de formation (facultatif)	p. 36
Déclaration sur l'honneur	p. 37
Documents illustrant la pratique professionnelle (facultatif)	p. 38

DOSSIER PROFESSIONNEL (DP)

Annexes (*Si le RC le prévoit*)

p. 39



DOSSIER PROFESSIONNEL (DP)

EXEMPLES DE PRATIQUE PROFESSIONNELLE

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Exemple n°1 -

Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Maquetter une application

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

Pendant la formation, j'ai réalisé un projet pour prendre des rendez-vous en médecine. Nous l'avons appelé "we care".

Tout d'abord, nous avons analysé les besoins du client :

- Le client peut créer un compte et se connecter.
- il peut effectuer une recherche avec les symptômes pour trouver un hôpital.
- ils peuvent prendre des rendez-vous.

Nous avons utilisé un outil appelé figma pour créer le zonage et la maquette. Nous avons commencé par le zonage de chaque page pour avoir une idée précise de la disposition des éléments. Le zonage permet de diviser visuellement la page en plusieurs zones en fonction des besoins. Cette étape intervient généralement après avoir réalisé l'arborescence du futur site web. Ce plan du site permet d'avoir une idée générale du parcours de navigation des utilisateurs.



L'exemple de wireframe de la page d'accueil.



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)



Search

Pearl River Medical Clinic

12 rue de france , Paris 75123
07 98 26 35 45

TAG	TAG	TAG
TAG	TAG	TAG
TAG	TAG	TAG

Pearl River Medical Clinic

● 07 98 26 35 45 ● 12 rue de france , Paris 75123

TAG

TAG

TAG

TAG

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed libero praesent ullamcorper enim euismod adipiscing pharetra turpis ac. Integer netus amet, arcu, amet turpis. Fames tortor commodo ornare nisl, nisl nisl, morbi vestibulum commodo. Nec eget eget in ante suspendisse ornare vestibulum neque velit.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed libero praesent ullamcorper enim euismod adipiscing pharetra turpis ac. Integer netus amet, arcu, amet turpis. Fames tortor commodo ornare nisl, nisl nisl, morbi vestibulum commodo. Nec eget eget in ante suspendisse ornare vestibulum neque velit.

Hopital le plus grand filmon

12 rue de france , Paris 75123
07 98 26 35 45

TAG	TAG	TAG
TAG	TAG	TAG
TAG	TAG	TAG

Prendre RDV

Les wireframes de la version mobile.

Une fois qu'on a bien précisé nos idées à l'aide des wireframes, on concrétise le visuel du site via des maquettes qui seront des représentations réalistes du site final et qui serviront de modèle aux développeurs. On intègre également la charte graphique de site qui nous avait été fournie en amont.

DOSSIER PROFESSIONNEL (DP)

2. Précisez les moyens utilisés :

3. Avec qui avez-vous travaillé ?

En équipe avec 2 autres apprenants.

4. Contexte

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

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

Période d'exercice ➤ Du : Cliquez ici au : Cliquez ici

5. Informations complémentaires (*facultatif*)



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Exemple n°2 - *Développer une interface utilisateur de type desktop*

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

Pour commencer ce projet, nous devons créer un site vitrine pour une entreprise fictive. Dans notre cas, nous avons imaginé un site pour un magasin qui vendrait des plantes. En utilisant HTML(Hypertext Markup Language) ,CSS (Feuilles de style en cascade) et JavaScript. Le site doit être adaptable.

1.On a commencé avec les WIREFRAMES du site en utilisant le site Figma



2.Pour suivre on a décidé les couleurs et les polices pour la mise en page du site.

3.Puis on a créé plusieurs fichiers HTML pour chaque page du site en respectant la structure de wireframe.Une fois la structure HTML réalisée, on met en forme notre site grâce aux feuilles de style CSS. Par exemple , le header de notre site va être fixé en faisant le scroll vers le bas de la page . Pour avoir cette fonctionnalité, on a codé une fonction en javascript qui permet d'ajouter ou d'enlever une classe à la barre de navigation, en fonction de sa position.

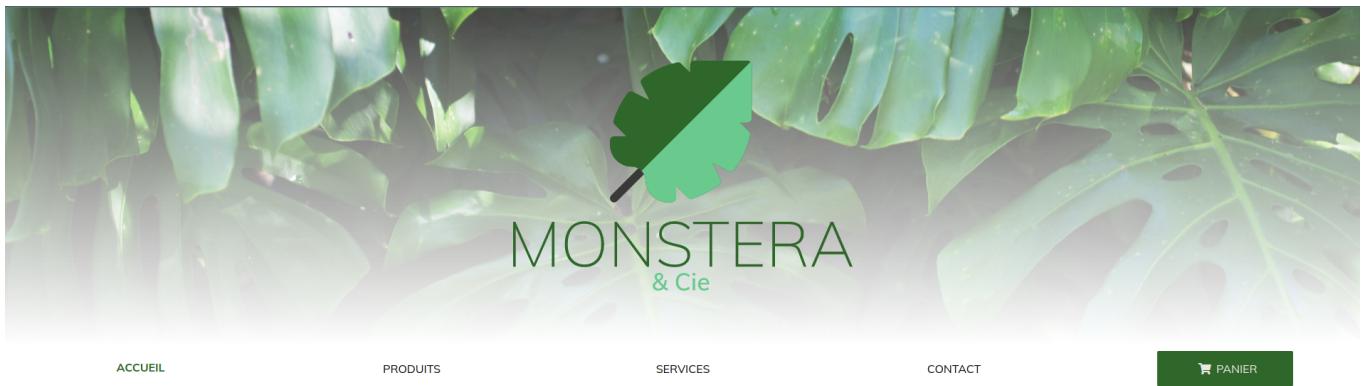
on a divisé le site en plusieurs parties , par exemple header , home section , footer , card etc . on commence avec une partie et on finalise le code et on avance pour le deuxième . En utilisant cette méthode, on arrive à avancer plus vite.

DOSSIER PROFESSIONNEL (DP)

Voici un exemple de code et de la sortie .

```
<header id="header">
  
</header>

<nav id="main">
  <ul>
    <li class="logo"><a href="#header"></a></li>
    <li><a class="activeted" href="index.html">Accueil</a></li>
    <li><a href="Product_page/produits.html">Produits</a></li>
    <li><a href="service_page/services.html">Services</a></li>
    <li><a href="contact_page/contact.html">Contact</a></li>
    <li><a class="button" href="#"><i class="fas fa-shopping-cart"></i>Panier</a></li>
  </ul>
</nav>
```



Voici le code qui permet de fixer la barre de navigation:

```
let logo = document.querySelector(".logo");
let main = document.querySelector("main");
let space = bar.offsetTop;

function fixMyNav() {
  if (window.scrollY >= space) {
    bar.classList.add("newnav");
    logo.style.maxWidth = "25%";
    main.style.paddingTop = "10%";
  }
  else {
    bar.classList.remove("newnav");
    logo.style.maxWidth = "0";
    main.style.paddingTop = "5%";
  }
}
window.addEventListener("scroll", fixMyNav);
```



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

4.A la fin on vérifie que le site représente la maquette qu'on avait fait en avant et je modifie le code pour que ça respecte la maquette .

MONSTERA & Cie

ACCUEIL PRODUITS SERVICES CONTACT PANIER

La boutique

Fournisseur de Curiosités Végétales, MONSTERA & Cie mise sur le rare, l'authentique et la pièce unique, au service du bien être et dans le respect du vivant.

Nous réalisons aussi, sur devis, la conception et l'installation de décor végétal pour les particuliers et les professionnels.

MONSTERA & Cie est un projet hybride, né

Plantes d'intérieures exotiques

Alocasia Cadieri

40€ -20% Ajouter au panier

Alocasia Red Secret

50€ Ajouter au panier

Alocasia Black Velvet

50€ Ajouter au panier

DOSSIER PROFESSIONNEL (DP)

[ACCUEIL](#)[PRODUITS](#)[SERVICES](#)[CONTACT](#) [PANIER](#)

Nos Services



LIVRAISON

Nous livrons et installons vos plantes spécifiques sur commande.

[Devis sur demande](#)

INSTALLATION

Nous aménageons et mettons en valeur vos espaces professionnels ou privés.

[Devis sur demande](#)

2. Précisez les moyens utilisés :

Je crée les wireframes avec Figma

J'utilise les langages HTML et CSS pour la partie statique de l'application

J'utilise javascript pour les fonctions.

J'utilise Git et Github pour sauvegarder mon travail.

3. Avec qui avez-vous travaillé ?

En équipe avec 3 autres apprenants.

4. Contexte

Nom de l'entreprise, organisme ou association ➤ [Cliquez ici pour taper du texte.](#)

Chantier, atelier, service ➤ [Cliquez ici pour taper du texte.](#)

Période d'exercice ➤ Du : [Cliquez ici](#) au : [Cliquez ici](#)

5. Informations complémentaires (facultatif)



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Exemple n°3 - Développer des composants d'accès aux données

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

Pendant mon alternance j'ai développé un composant d'accès aux données. Ce m'a permis de me défier et de mettre en pratique mes connaissances en matière de requête HTTP mais aussi de pouvoir mais aussi de pouvoir manipuler des données. Lors de la création d'une application web, le besoin d'afficher des données provenant d'une API est courant. Il y a plusieurs façons de le faire, mais une approche très populaire est d'utiliser fetch, un client HTTP basé sur les promesses.

Le HTTP est un protocole de la couche application. Cela veut dire qu'il doit être implémenté dans les logiciels responsables pour cette communication comme dans les navigateurs et les serveurs web. Il travaille toujours avec deux autres protocoles : Le CTP (responsable de la transférence des informations) et le IP (responsable de l'acheminement des informations).

Cette communication réalisée par le HTTP, suis un modèle de Request et Response : Une demande est effectuée et une réponse est envoyée. Lorsqu'on accède à un site on effectue une demande au serveur, le serveur en recevant cette demande fait tous les processus internet et envoie une réponse. Généralement cette réponse est un contenu en HTML qui serait interprété par le navigateur, donnant ainsi le contenu de la page.

Après ce processus de demande du navigateur et de réponse du serveur, la connexion se termine. Ainsi il est nécessaire de répéter tout le cycle lorsqu'une nouvelle demande est effectuée. Cela veut dire que la connexion n'est pas persistante. Pour chaque accès ou actualisation du contenu une nouvelle connexion HTTP est créée et tout le cycle est répété. Il est important de préciser que le protocole HTTP ne se limite qu'aux navigateurs web. Il est aussi utilisé dans la plupart des applications mobiles pour communiquer avec la partie backend.

Il existe plusieurs manières d'interroger une API, mais il est préférable de d'abord connaître la structure des données qu'elle renvoie afin de savoir ce qu'elle va afficher. Tout commence par une demande au serveur. Ici trois informations très importantes sont nécessaires.

- ❖ L'identification de la méthode (GET , POST , PATCH , PUT , DELETE). Pour mon exemple c'est un GET
- ❖ L'URL, l'adresse où sera envoyé la demande
- ❖ La version du protocole : HTTP 0.9, HTTP 1.0, HTTP 1.1 et HTTP 2. HTTP 1.1 est la version adoptée par la plupart des systèmes.

DOSSIER PROFESSIONNEL (DP)

```
axios({
  method: 'get',
  url: `${process.env.REACT_APP_API_URL}/search`,
  withCredentials: true,
})
```

Après avoir envoyé une demande, le serveur m'a envoyé une réponse avec des informations très importantes à regarder comme :

- ❖ La ligne de statuts avec trois informations très importantes : La version du protocole utilisée par le serveur, le code numérique du statut qui indique comment notre demande a été conditionnée dans le serveur et le texte associé au statut. Le status code est une des informations la plus importante d'une réponse. Elle nous permet de savoir si la demande a été effectuée avec ou sans succès.
- ❖ L'entête contient des informations supplémentaires sur la demande et le type de réponse. Des Informations comme : Content-Type le contenu envoyé dans la réponse, Access-Control-Allow indique si la réponse peut être accessible par l'origine de la demande et le date , le contenu de la réponse générée.

Voici le résultat de l'api call ce dessus.

```
▼ {data: {...}, status: 200, statusText: 'OK', headers: {...}, config: {...}, ...} ⓘ index.js:23
  ► config: {transitional: {...}, transformRequest: Array(1), transformResponse: Array(1), timeout: 0, adapter: f, ...}
  ► data: {data: Array(10)}
  ► headers: {content-length: '17075', content-type: 'application/json; charset=utf-8'}
  ► request: XMLHttpRequest {onreadystatechange: null, readyState: 4, timeout: 0, withCredentials: true, upload: XMLHttpRequestUpload, ...}
    status: 200
    statusText: "OK"
  ► [[Prototype]]: Object

  ▼ data:
    ▼ data: Array(10)
      ► 0: {AnnouncementId: 199, UserId: 28, Title: 'text', Description: 'decrivez', AddressCity: 'Gassin', ...}
      ► 1: {AnnouncementId: 198, UserId: 28, Title: 'Cannes', Description: "qui l'entoure", AddressCity: 'Toulouse', ...}
      ► 2: {AnnouncementId: 194, UserId: 29, Title: 'Canne', Description: 'entoure', AddressCity: 'Lestelle-de-Saint-Martory', ...}
      ► 3: {AnnouncementId: 192, UserId: 8, Title: 'RamZ', Description: 'si tout fonctionne', AddressCity: 'FRance', ...}
      ► 4: {AnnouncementId: 191, UserId: 8, Title: 'RamZ', Description: 'si tout fonctionne', AddressCity: 'FRance', ...}
      ► 5: {AnnouncementId: 190, UserId: 27, Title: 'Title', Description: 'Un rond-point', AddressCity: 'Toulouse', ...}
      ► 6: {AnnouncementId: 187, UserId: 21, Title: 'RamZ', Description: 'si tout fonctionne', AddressCity: 'FRance', ...}
      ► 7: {Announc[ Object ] 186, UserId: 21, Title: 'RamZ', Description: 'si tout fonctionne', AddressCity: 'FRance', ...}
      ► 8: {AnnouncementId: 185, UserId: 21, Title: 'RamZ', Description: 'si tout fonctionne', AddressCity: 'FRance', ...}
      ► 9: {AnnouncementId: 184, UserId: 21, Title: 'RamZ', Description: 'si tout fonctionne', AddressCity: 'FRance', ...}
      length: 10
```

une fois que j'ai la réponse de l'api je le stock dans un state react pour afficher .

```
const [searchResult, setSearchResult] = useState([]);
```



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

En utilisant le state j'affiche les données en front end en faisant un map .

```
<section className="resultSection">
  {searchResult.map((result, index) =>
    <Card
      key={index}
      CoverImage={result.AnnouncementPhotos.length > 0 ? result.AnnouncementPhotos[0].Content : Noimage}
      Title={result.Title}
      Avatar={result.User.Avatar}
      City={result.AddressCity}
      ZipCode={result.ZipCode}
      HourPrice={result.HourPrice}
      Id={result.AnnouncementId}
    />
  ))}
</section>
```

à la fin ça affiche comme ça pour les clients.



2. Précisez les moyens utilisés :

3. Avec qui avez-vous travaillé ?

j'ai travaillé seul.

4. Contexte

DOSSIER PROFESSIONNEL (DP)

Nom de l'entreprise, organisme ou association ➤ SWIMANGO

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

Période d'exercice ➤ Du : Cliquez ici au : Cliquez ici

5. Informations complémentaires (*facultatif*)



DOSSIER PROFESSIONNEL (DP)

Activité-type 1

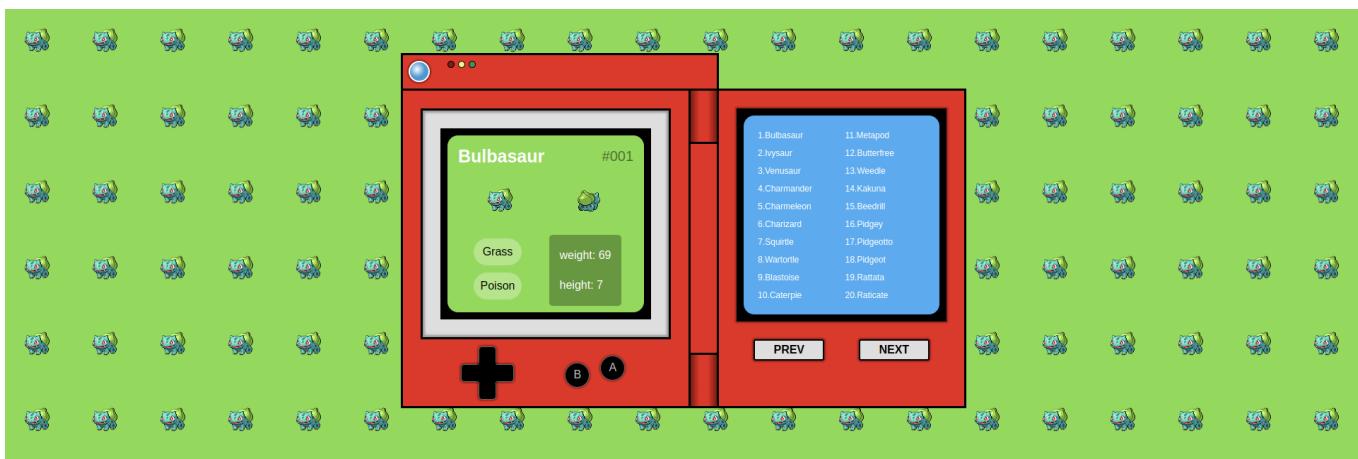
Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Exemple n°4 - Développer la partie front-end d'une interface utilisateur web

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

L'objectif de ce projet était la mise en pratique d'AJAX en récupérant les informations d'une API pour les injecter dynamiquement dans mon application, ici un Pokédex. Pour ce faire, nous devions récupérer le contenu en faisant des requêtes à l'API pokeAPI (<https://pokeapi.co/>). Une fois les données récupérées, nous devions les intégrer dynamiquement sur notre page HTML. Le projet intégrait les fonctionnalités suivantes :

- Quand j'appuie sur next, la liste se met à jour avec les 20 Pokémons suivants (s'il y en a).
- Quand j'appuie sur prev, la liste se met à jour avec les 20 Pokémons précédents (s'il y en a).
- Quand je clique sur le nom d'un Pokémon, le Pokédex affiche les informations du Pokémon concerné .
- La couleur de fond s'adapte au type du Pokémon et quand le Pokémon a un seul type, afficher un .
- seul type, quand il en a deux, afficher les deux .
- Ajouter une fonction avec konami code .



Le visuel de la page au début

j'ai utilisé la méthode fetch en javascript pour récupérer le nom , l'image , les types et les couleurs de fond .La méthode fetch renvoie une promesse (un objet de type Promise) qui va se résoudre avec un objet Response qui contient tous les données que je demande.

Pour chaque clic dans les noms de pokémons il y a une demande de ressources à faire avec la méthode fetch en Javascript pour avoir le nom , l'image , les types et les couleurs de fond .Voici un exemple qui

DOSSIER PROFESSIONNEL (DP)

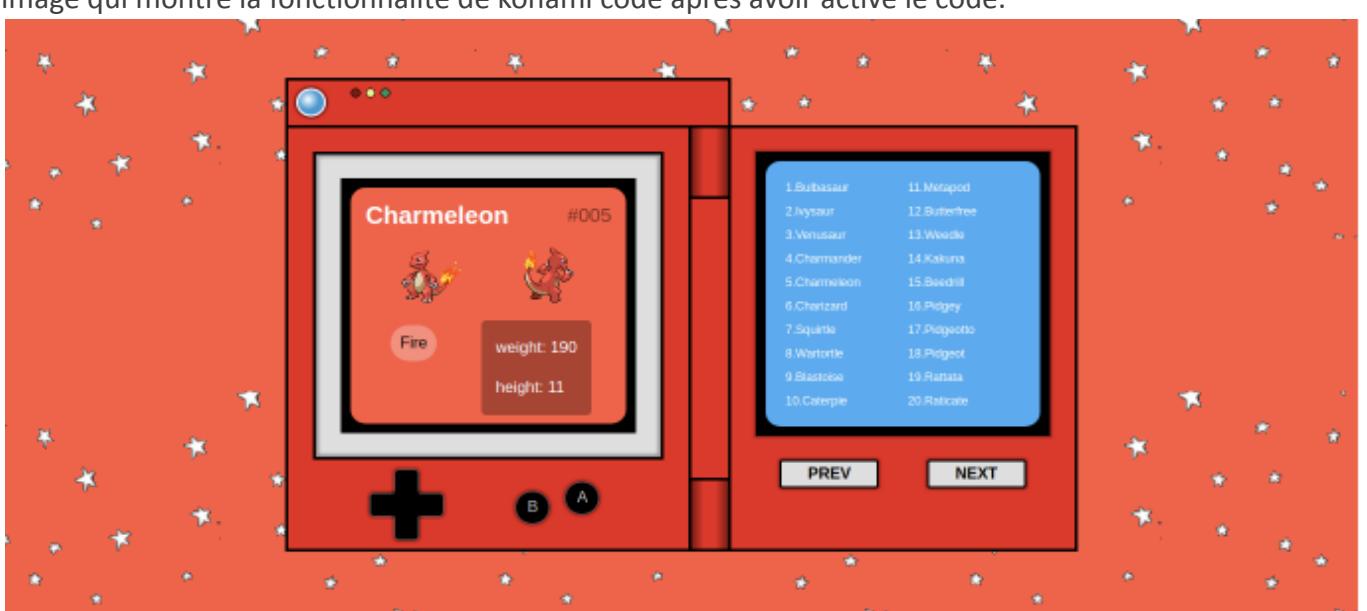
montre la méthode fetch.

```
function getpokies() {
    fetch(apilink)
        .then(function (response) {
            response.json()
                .then(function (data) {
                    if (response.status == 400) {
                        console.log(data);
                        // gestion erreur données envoyées à la requête
                    } else if (response.status == 403) {
                        console.log(data);
                        // gestion erreur authentification
                    } else {
                        //console.log(data);
                        let arr = data["results"];
                        let str = "";
                        for (let i = 0; i < arr.length; i++) {
                            fetch(data["results"][i]["url"])
                                .then(function (response) {
                                    response.json()
                                        .then(function (data) {
```

J'ai codé une fonctionnalité pour ajouter le konami code.konami code est un code secret qui permet de tricher sur les jeux notamment le code secret est ↑↑↓↓←→←→. J'ai ajouté pour changer le style de site pour m'amuser. Pour cette functionality j'ai écrit un fonction qui est à l'écoute de tous les button qu'on appuie .

```
▶ (11) ['F12', 'ArrowUp', 'ArrowUp', 'ArrowUp', 'ArrowUp', 'ArrowUp', 'ArrowUp',
'ArrowRight', 'ArrowRight', 'Control', 'Shift']
```

une fois que la combinaison de boutons sont même à la konami code la fonction va appeler .voici une image qui montre la fonctionnalité de konami code après avoir activé le code.





MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

2. Précisez les moyens utilisés :

- J'ai utilisé la méthode fetch en langage de programmation Javascript,HTML,CSS pour effectuer la requête API ainsi que l'intégration dynamique de mon contenu
- J'ai utilisé Git et Github pour sauvegarder mon travail.

3. Avec qui avez-vous travaillé ?

J'ai réalisé cette application avec un binôme

4. Contexte

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

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

Période d'exercice ➤ Du : Cliquez ici au : Cliquez ici

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 1

Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Exemple n°5 - Développer la partie back-end d'une interface utilisateur web

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

Pendant la formation, nous avons appris à créer une application backend avec node js.

Dans l'exemple qui suis, je présente une application de librairie en ligne.

J'ai développé l'application en adoptant une architecture multicouche de type modulaire. L'idée est de tout diviser en plusieurs modules. Dans mon projet, j'ai un module User et un module Book. Chaque module est indépendant.

Par exemple, chacun de mes modules contient un controller, un service, des dao, un fichier de routes etc. Les routes sont ensuite injectées dans un fichier d'index permettant de centraliser celles-ci.

Un fichier javascript nommé app est créé à la racine, dans celui-ci sont importés tous les fichiers de configuration ainsi que l'index de chaque dossier. C'est dans ce fichier que j'initialise le serveur HTTP.

Dans mon dossier config, se trouvent toutes les configurations principales de l'application, nous pouvons retrouver le fichier de configuration de base de données, le fichier de variable d'environnement ainsi qu'un fichier server.js.

C'est dans ce fichier de serveur que se trouve toute l'initialisation de l'application.

J'ai créé une classe Serveur qui initialise mon routeur et les middlewares. Chaque module comporte un fichier index. Dans celui-ci nous créons les nouvelles instances de nos différents services ou controller.

J'exporte également ces instances pour pouvoir les utiliser, on appelle ça l'injection de dépendances.

La principale cause intéressante de l'injection de dépendance est de pouvoir diviser le code par responsabilité. Cela nous évite de devoir tout coder au même endroit dans une même classe.

Voici ci-dessous un exemple de module :



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

```
1 import { Model, DataTypes } from 'sequelize';
2 import db from '../config/database';
3
4 class Book extends Model {
5     static init(sequelize) {
6         return super.init(
7             {
8                 title: DataTypes.STRING,
9                 code: DataTypes.INTEGER,
10                quantity:DataTypes.INTEGER,
11                userId:{
12                    type:DataTypes.INTEGER,
13                    primaryKey:true,
14                    autoIncrement:true
15                }
16            },
17            { sequelize, modelName: 'Book' }
18        );
19    }
20
21    static associate(models) {
22        // define association here
23        //this.belongsTo(models.User);
24        return this;
25    }
26}
27
28 Book.init(db.sequelize);
29
30 export default Book;
```

Le controller a sa propre responsabilité d'appeler les services et de renvoyer un statut après l'exécution d'un service. Le service quant à lui s'occupe de la logique métier et d'appeler un repository qui lui-même s'occupe de faire des traitements en base de données et de retourner une valeur. C'est ce que nous allons démontrer dans la partie consacrée à la base de données.

DOSSIER PROFESSIONNEL (DP)

Ci-dessous un exemple de mon fichier de contrôleur:

```
import Book from './book.js';
import env from '../../config/env'

class BookController {

    #models;
    constructor(models) {
        this.#models = models;
    }

    getAll = async ({ rq ,res, next }) => {
        try {
            const books = await Book.findAll();
            res.status(200).json(books);
        } catch (err) {
            next(err);
        }
    }
    add = async (req, res, next) => {
        try {
            const book = await Book.create({ ...req.body });
            res.status(201).json(book);
        } catch (err) {
            next(err);
        }
    }
};

export default BookController;
```

2. Précisez les moyens utilisés :

J'ai utilisé Node JS.

3. Avec qui avez-vous travaillé ?

J'étais dans un groupe de trois .

4. Contexte



DOSSIER PROFESSIONNEL (DP)

Nom de l'entreprise, organisme ou association ▶ [Cliquez ici pour taper du texte.](#)

Chantier, atelier, service ▶ [Cliquez ici pour taper du texte.](#)

Période d'exercice ▶ Du : [Cliquez ici](#) au : [Cliquez ici](#)

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL (DP)

Activité-type 2

Exemple n°1 -

Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

Concevoir une base de données

Mettre en place une base de données

Développer des composants dans le langage d'une base de données

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

Pendant la formation, j'ai travaillé à créer une application de gestion de bibliothèque en ligne.

Pour valider les compétences :

- Concevoir une base de données
- Mettre en place une base de donnée
- Développer des composants dans le langage d'une base de données

Je vais expliquer comment j'ai dû créer un modèle physique de données, puis comment j'ai configuré ma base de données en fonction de cette schématisation, et enfin ce que j'ai mis en place pour accéder aux données stockées dans cette base.

L'objectif principal est de montrer comment j'ai assuré le stockage et la gestion des données, en ayant préalablement pensé à la modélisation et à la création de cette base de données, afin d'optimiser et d'automatiser les processus d'accès dont les résultats sont destinés à être exploités.

1. Concevoir une base de données

Dans un premier temps je suis la réalisation du MPD qui fait partie de la méthode Merise, celle de la construction concrète et utilisable, qui doit permettre de se baser exactement sur le schéma produit pour créer une base de données. En effet, lorsque ce schéma est correctement réalisé sur une plate-forme adéquate, il peut être directement exploité pour créer automatiquement une base de données.

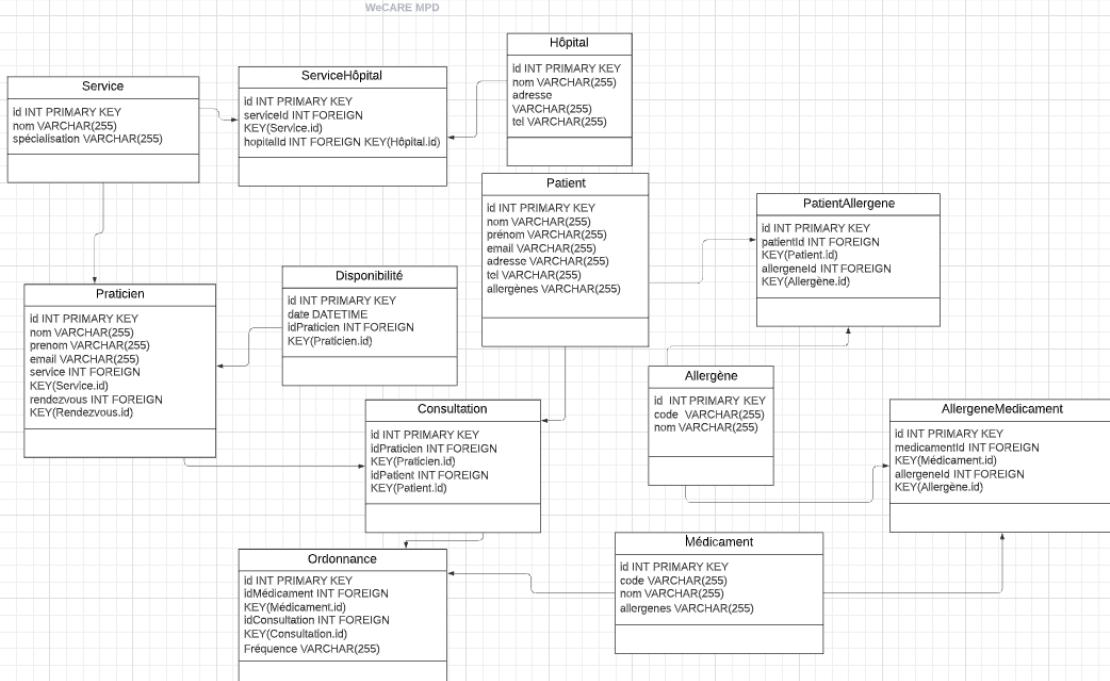
2. Mettre en place une base de donnée

Ici je vais présenter la réalisation du MPD qui est une partie de la méthode Merise, celle de la construction concrète et utilisable, qui doit permettre de se baser exactement sur le schéma produit pour créer une base de données. En effet, lorsque ce schéma est correctement réalisé sur une plateforme adéquate il peut être directement exploité pour créer automatiquement une base de données. Cette étape me permet donc d'avoir une vision solide et mise en pratique de la manière dont je vais stocker et utiliser mes données, le plus logiquement et efficacement possible selon l'utilisation que je vais en faire.



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI



Sur cette image, les rectangles représentent les tables de ma base de données, on y voit les attributs propres à chaque champ. La notion de clés est aussi présente, c'est ce qui va me permettre de connecter mes tables entre elles, au moyen d'identifiants uniques : les primary key sont uniques (ici ce sont des ID auto incrémentées). Les foreign key sont les attributs par lesquels les tables peuvent être connectées, jointes. Les relations entre les différentes tables se font donc au moyen de clés étrangères. La base de données que je vais développer et utiliser sur ce projet est en MySQL, c'est un système de gestion de bases de données relationnelles, il utilise pour cela le langage SQL.

3.Développer des composants dans le langage d'une base de données

Pour ce projet je fais le choix de travailler avec un "object-relational mapping" - ORM , Sequelize, afin de sécuriser l'accès aux stockage en base de données. Un ORM est une sorte de programme qui fait l'interface entre mon code et ma base de données. Je vais ainsi déléguer la gestion de ma base de données à cet ORM pour y ranger, trier et accéder aux informations stockées. Mon but ici est multiple, autant de prévenir au maximum les risques d'injections SQL non désirées ou malveillantes, que pour faciliter l'accès aux données avec le canal précis que représente les méthodes interne à l'ORM. Ces dernières permettent à l'ORM de traduire mes demandes en commandes SQL, des plus simples comme la création d'une table .

DOSSIER PROFESSIONNEL (DP)

```
1 import { Model, DataTypes } from 'sequelize';
2 import db from '../config/database';
3
4 class Book extends Model {
5     static init(sequelize) {
6         return super.init(
7             {
8                 title: DataTypes.STRING,
9                 code: DataTypes.INTEGER,
10                quantity:DataTypes.INTEGER,
11                userId:{
12                    type:DataTypes.INTEGER,
13                    primaryKey:true,
14                    autoIncrement:true
15                }
16            },
17            { sequelize, modelName: 'Book' }
18        );
19    }
20
21    static associate(models) {
22        // define association here
23        //this.belongsTo(models.User);
24        return this;
25    }
26 }
27
28 Book.init(db.sequelize);
29
30 export default Book;
```

Voici l'exemple.

2. Précisez les moyens utilisés :

- J'ai utilisé Sequelize.

3. Avec qui avez-vous travaillé ?

-J'étais dans un groupe de trois .

4. Contexte

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

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

Période d'exercice ➤ Du : Cliquez ici au : Cliquez ici

5. Informations complémentaires (*facultatif*)



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

Activité-type 2

Exemple n°2 - Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

Mettre en place une base de données

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

Pour un projet personnel, j'ai mis en place une base de données en utilisant un ORM(Prisma) . De nos jours, il est très rare de configurer une base de données "à la main" avec des requêtes SQL car en le faisant à la main on risque d'avoir des erreurs et c'est plus difficile quand il y a une modification de la structure. Les développeurs utilisent couramment les ORM, qui sont maintenant des technologies stables et matures.

Un ORM est un ensemble de classes permettant de manipuler les tables d'une base de données relationnelle comme s'il s'agissait d'objets. C'est une couche d'abstraction d'accès à la base de données qui donne l'illusion de ne plus travailler avec des requêtes mais de manipuler des objets. Pour ce faire, l'ORM va s'appuyer sur des modèles qui vont chacun représenter une table de la base de données.

Un des avantages de l'ORM c'est qu'on ne dépend plus d'un SGBD (Système de gestion de base de données) en particulier. Si l'on souhaite changer de SGBD, l'ORM va s'occuper de transformer les requêtes pour les rendre compatibles avec la nouvelle base de données.

L'utilisation d'un ORM découle du principe de « séparation des responsabilités », il permet de découpler une application de son système de persistance des données. Au cours de ma formation, on m'avait déjà demandé d'utiliser un ORM appelé Sequelize. Cependant, Sequelize est un peu compliqué pour moi et sa documentation aussi.

Malheureusement, il n'y a pas beaucoup d'ORMs matures dans l'écosystème JavaScript puisque ce langage est relativement récent dans son utilisation pour le back-end (Node.js). Après quelques recherches, j'ai choisi Prisma pour son excellente documentation et son langage de déclaration de modèle lisible. Étant donné que Prisma est un ORM principalement destiné au langage TypeScript (une version améliorée de JavaScript qui est ensuite transposée en JavaScript), je préfère Prisma à Sequelize.

DOSSIER PROFESSIONNEL (DP)

Le fichier « schema.prisma » contient la définition des modèles et la configuration de la connexion à la base de données. J'ai par ailleurs choisi PostgreSQL en tant que SGBD, pour sa très bonne compatibilité avec Prisma. Voici à quoi ressemble le début début du fichier :

```
generator client {
    provider = "prisma-client-js"
}

datasource db {
    provider = "mysql"
    url      = env("DATABASE_URL")
}
```

Pour faciliter la configuration de l'application, l'URL de connexion à la base de données est stocké dans un fichier de variables d'environnement. Pour commencer j'ai utilisé un data base local et J'ai créé une base de données nommée dbalertngo.

```
DATABASE_URL="mysql://vishnugopy0010:10131013@localhost:3306/dbalertngo"
```

Une fois la configuration terminée, on peut commencer à définir nos modèles. Il me suffit globalement de traduire mon MDP en suivant la syntaxe de déclaration de modèle de Prisma. Voici le modèle User et Post:

```
model User {
    id      Int      @id @default(autoincrement())
    email   String   @unique
    name    String?
    posts   Post[]
    profile Profile?
}
```



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

```
model Post {
    id      Int      @id @default(autoincrement())
    createdAt DateTime @default(now())
    updatedAt DateTime @updatedAt
    title   String   @db.VarChar(255)
    content String?
    published Boolean @default(false)
    author   User     @relation(fields: [authorId], references: [id])
    authorId Int
}
```

Une fois les modèles bien défini, j'utilise l'outil en ligne de commande « migrate » fourni par Prisma, qui permet de :

Générer les fichiers SQL de migration.

- Exécute les fichiers de migration pour créer les tables de la base de données.
- Génère le « Prisma Client » qui est un ensemble de classes (une par table) avec des méthodes (find Unique, find Many, create, update, delete) pour manipuler la base de données.

Voici un exemple de fichier de migration pour la table User et la table Post généré par prima :

```
-- CreateTable
CREATE TABLE `Post` (
    `id` INTEGER NOT NULL AUTO_INCREMENT,
    `createdAt` DATETIME(3) NOT NULL DEFAULT CURRENT_TIMESTAMP(3),
    `updatedAt` DATETIME(3) NOT NULL,
    `title` VARCHAR(255) NOT NULL,
    `content` VARCHAR(191) NULL,
    `published` BOOLEAN NOT NULL DEFAULT false,
    `authorId` INTEGER NOT NULL,

    PRIMARY KEY (`id`)
) DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;

-- CreateTable
CREATE TABLE `User` (
    `id` INTEGER NOT NULL AUTO_INCREMENT,
    `email` VARCHAR(191) NOT NULL,
    `name` VARCHAR(191) NULL,

    UNIQUE INDEX `User_email_key`(`email`),
    PRIMARY KEY (`id`)
) DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
```

DOSSIER PROFESSIONNEL (DP)

2. Précisez les moyens utilisés :

3. Avec qui avez-vous travaillé ?

j'ai travaillé seul

4. Contexte

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

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

Période d'exercice ➤ Du : Cliquez ici au : Cliquez ici

5. Informations complémentaires (*facultatif*)



DOSSIER PROFESSIONNEL (DP)

Activité-type 2

Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

Exemple n°3 - Développer des composants dans le langage d'une base de données

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

-
-
-
-
-

2. Précisez les moyens utilisés :

-
-
-
-
-

3. Avec qui avez-vous travaillé ?

-
-
-
-
-

4. Contexte

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

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

Période d'exercice ➤ Du : Cliquez ici au : Cliquez ici

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Activité-type 3

Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Exemple n°2 ▶ *Concevoir une application*

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

-
-
-
-
-

2. Précisez les moyens utilisés :

-
-
-
-
-

3. Avec qui avez-vous travaillé ?

-
-
-
-
-

4. Contexte

Nom de l'entreprise, organisme ou association ▶ [Cliquez ici pour taper du texte.](#)

Chantier, atelier, service ▶ [Cliquez ici pour taper du texte.](#)

Période d'exercice ▶ Du : [Cliquez ici](#) au : [Cliquez ici](#)

5. Informations complémentaires (*facultatif*)



DOSSIER PROFESSIONNEL (DP)

Activité-type 3

Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Exemple n°3 ▶ Développer des composants métier

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

-
-
-
-
-

2. Précisez les moyens utilisés :

-
-
-
-
-

3. Avec qui avez-vous travaillé ?

-
-
-
-
-

4. Contexte

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

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

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

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Activité-type 3

Exemple n°4 ▶

Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité
Construire une application organisée en couches

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

-
-
-
-
-

2. Précisez les moyens utilisés :

-
-
-
-
-

3. Avec qui avez-vous travaillé ?

-
-
-
-
-

4. Contexte

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

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

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

5. Informations complémentaires (facultatif)



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Activité-type 3

Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Exemple n°5 ▶ Développer une application mobile

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

-
-
-
-
-

2. Précisez les moyens utilisés :

-
-
-
-
-

3. Avec qui avez-vous travaillé ?

-
-
-
-
-

4. Contexte

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

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

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

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Activité-type 3

Exemple n°6 ▶ *Préparer et exécuter les plans de tests d'une application*

Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

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

-
-
-
-
-

2. Précisez les moyens utilisés :

-
-
-
-
-

3. Avec qui avez-vous travaillé ?

-
-
-
-
-

4. Contexte

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

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

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

5. Informations complémentaires (*facultatif*)



DOSSIER PROFESSIONNEL (DP)

Activité-type 3

Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Exemple n°7 - Préparer et exécuter le déploiement d'une application

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

Après avoir développé mon application, il est intéressant de pouvoir la déployer en production. Pour se faire, il est important de corriger et d'apporter certaines modifications à nos projets back-end aussi bien que le front-end. Certaines configurations ne sont pas recommandées pour une mise en production mais seulement utile en phase développement.

Pour mettre à disposition mon application web de chat en ligne sur le web, j'ai décidé d'utiliser un VPS pour héberger mon site. Plusieurs hébergeurs proposent ce type de service. J'ai choisi d'acheter un VPS (Serveur dédié virtuel) chez OVH.

J'ai installé dessus une machine Ubuntu et un serveur HTTP. Le logiciel de serveur que j'ai utilisé est Nginx. Il s'agit d'un reverse proxy, son rôle étant d'agir en tant qu'intermédiaire sur la communication réseau qui reçoit les requêtes pour les transmettre ensuite à l'ordinateur cible. C'est donc sur celui-ci que j'ai déployé mes applications front-end et back-end.

Après avoir acheté mon VPS sur Ovh, j'ai reçu des identifiants permettant de me connecter à celui via un protocole sécurisé, SSH. J'ai ensuite installé Nginx sur cette machine via une ligne de commande. J'ai également installé les dépendances nécessaires comme Node, Mysql et NPM le gestionnaire de paquet Node. J'ai créé une base de données sur ce serveur après avoir créé un nouvel utilisateur. Pour pouvoir envoyer sur mon serveur mon projet front, j'ai donc cloné le dossier Dist de mon projet et copié dans un répertoire de mon vps au chemin suivant : /var/www/html.

Cette manipulation rend tout de suite le projet front disponible sur l'adresse ip de mon VPS. J'ai répété la même manipulation pour installer sur mon serveur mon API. Pour pouvoir faire communiquer mon front et mon API, j'ai dû modifier les CORS permettant d'autoriser le front à pointer sur l'adresse IP de mon serveur (qui auparavant pointait en local). Pour pouvoir faire en sorte que mon API soit toujours disponible et active, j'ai installé un process manager pour node, PM2 sur mon VPS.

PM2 peut générer des scripts de démarrage et les configurer afin de conserver une liste de processus intacts lors des redémarrages de machines attendus ou inattendus. Pour lancer l'exécution de mon API en continu j'utilise la commande : pm2 start.

2. Précisez les moyens utilisés :

DOSSIER PROFESSIONNEL (DP)

3. Avec qui avez-vous travaillé ?

j'ai travaillé avec deux collègues

4. Contexte

Nom de l'entreprise, organisme ou association ➤ Simplon

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

Période d'exercice ➤ Du : Cliquez ici au : Cliquez ici

5. Informations complémentaires (*facultatif*)



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

Titres, diplômes, CQP, attestations de formation

(facultatif)

DOSSIER PROFESSIONNEL (DP)

Déclaration sur l'honneur

Je soussigné(e) Vishnugopy SUTHAHARAN

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 ERMONT

le 07 / 06/2022

pour faire valoir ce que de droit.

Signature :



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

Documents illustrant la pratique professionnelle

(facultatif)

DOSSIER PROFESSIONNEL (DP)

ANNEXES

(Si le RC le prévoit)