



# DOSSIER PROFESSIONNEL (DP)

*Nom de naissance* > ZAK  
*Nom d'usage* > ZAK  
*Prénom* > Lucien  
*Adresse* > Le Hameau de Pignelle  
330 chemin de camp redon  
83149 BRAS

## Titre professionnel visé

Concepteur Développeur Informatique

### MODALITÉ D'ACCÈS :

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

## Présentation du dossier

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

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

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

### Pour prendre sa décision, le jury dispose :

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

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

### Ce dossier comporte :

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

# DOSSIER PROFESSIONNEL <sup>(DP)</sup>

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

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

## Sommaire

### Exemples de pratique professionnelle

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

p. 5

➤ Quizz Connect

p. p. 6

➤ Sokoban (Projet Python scolaire)

p. p. 9

➤ Quiz Connect

p. p. 13

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

p.

➤ Quizz Connect

p. p. 17

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

p.

➤ Quizz Connect

p. p. 23

➤ Portfolio

p. p. 26

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 <sup>(DP)</sup>

---

---

# **EXEMPLES DE PRATIQUE PROFESSIONNELLE**

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

### Activité-type 1 sécurité

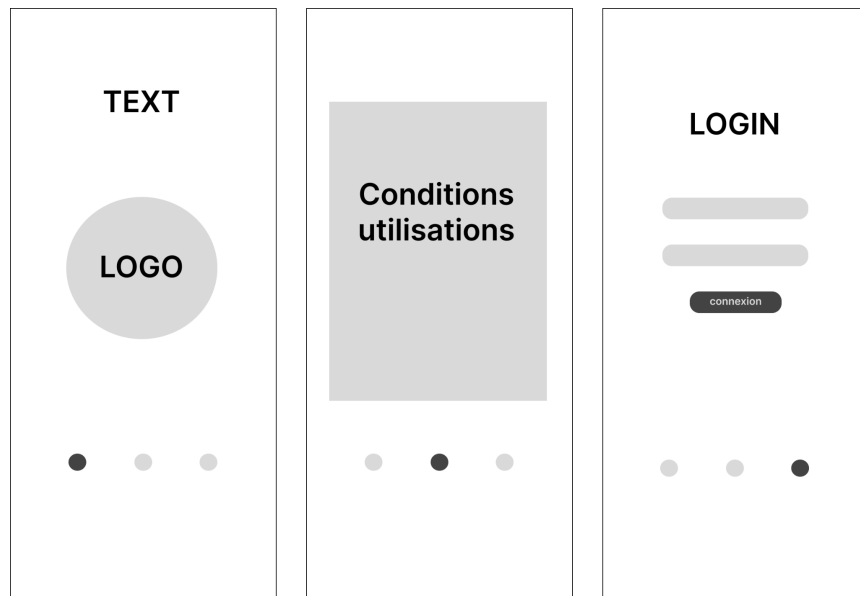
Exemple n°1 - Quizz Connect

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

Dans le cadre de ma formation au sein de la Plateforme, lors d'un projet de groupe, nous avons utilisé des maquettes pour concevoir la partie front de l'application.

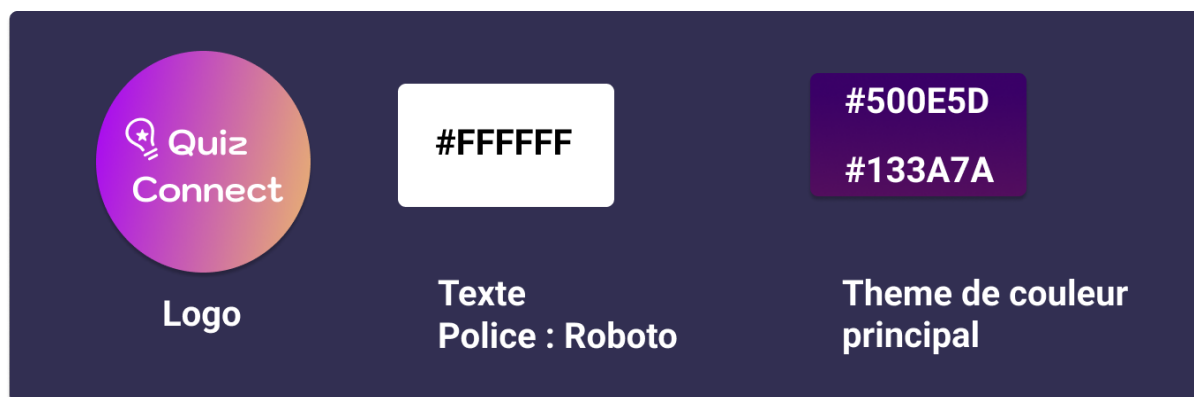
Cela nous a permis de prévoir l'emplacement des composants constituant nos écrans. La maquette a été établie à partir d'un cahier des charges précis.

Nous avons deux itérations, la première est la maquette Low-Fidelity ou WireFrame :

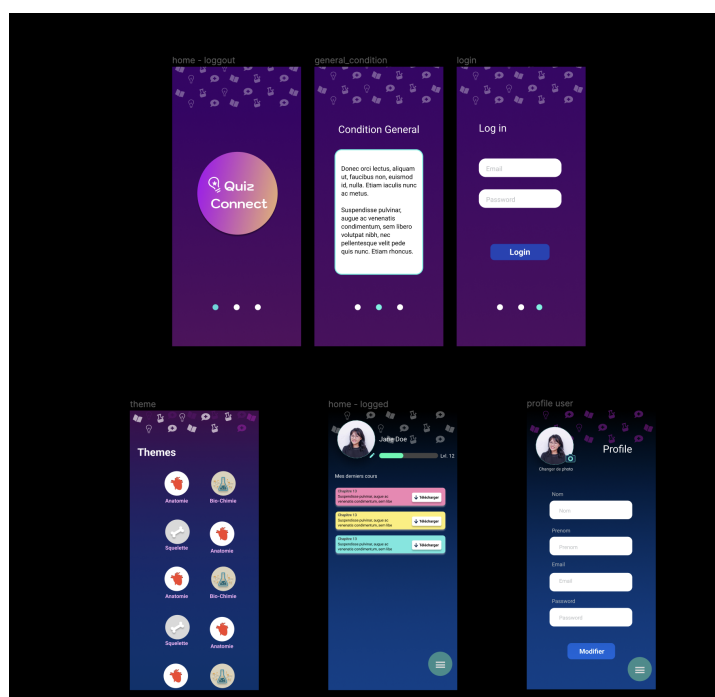


## DOSSIER PROFESSIONNEL <sup>(DP)</sup>

Nous avons ensuite décider d'une charte graphique, toujours en accord avec le cahier des charges :



A partir de cette conception, voici la maquette High-Fidelity, elle représente le rendu final de l'application.



L'outil **Figma** m'a permis de travailler de manière collaborative avec mes coéquipiers, ce qui a facilité l'échange de nos idées.

# DOSSIER PROFESSIONNEL <sup>(DP)</sup>

Compétences REAC couvertes par ce projet :

- **Maquetter une application**
- **Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement**
- **Concevoir une application**

## 2. Précisez les moyens utilisés :

OS : MacOS

Outils de maquettage : Figma

## 3. Avec qui avez-vous travaillé ?

Fatima Elmouhine et Ifanl Ibrahim

## 4. Contexte

Nom de l'entreprise, organisme ou association ▶ La Plateforme

Chantier, atelier, service ▶ Quizz Connect

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

## 5. Informations complémentaires (facultatif)



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

### Activité-type 1 sécurité

Exemple n°2 - Sokoban (Projet Python scolaire)

---

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

Dans le cadre scolaire nous avons dû développer une application desktop en Python. C'est un jeu Sokoban qui se lance en mode fenêtre, où l'utilisateur peut interagir avec l'interface via des événements claviers.

Pour pouvoir gagner une partie, l'utilisateur dispose d'un personnage, qu'il peut déplacer grâce aux touches directionnelles du clavier, et doit déplacer toutes les caisses sur les cibles.

Pour réaliser cette application et notamment gérer les collisions entre le personnage et les caisses j'ai utilisé la librairie PyGame qui vient avec beaucoup de fonctions utiles à la réalisation d'un jeu.

J'ai développé cette application en Programmation Orientée Objet (POO). La POO permet par la création de classe pour chaque composants du jeu (Personnage, Mur, Cibles, Caisse) une implémentation simple et facilement modulable grâce aux attributs et méthodes de classe, ainsi qu'à l'héritage qui permet de faire hériter des propriétés depuis une classe mère vers une classe fille.

Pour finir, j'ai créé un exécutable en un seul fichier et ne plus passer par la ligne de commande pour lancer mon script, j'ai utilisé la librairie **PyInstaller** et utilisé la commande (le dernier paramètre me sert à lui assigner une icon) : **`pyinstaller snake.py --onefile --windowed --icon snake-icon.icns`**

## Exemple d'une classe :

```
import pygame
from pygame.locals import *
import os

class Player(pygame.sprite.Sprite):

    def __init__(self, x, y, width, height, color, screen):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.color = color
        self.screen = screen
        self.image = pygame.transform.scale(pygame.image.load(os.path.join("assets", "personnage.png")),
        (self.width, self.height))

        self.rect = pygame.draw.rect(self.screen, self.color, (self.x, self.y, self.width, self.height))
        self.lastMove = (self.x, self.y)

    def draw(self):
        self.rect = pygame.draw.rect(self.screen, self.color, (self.x, self.y, self.width, self.height))
        self.screen.blit(self.image, self.rect)

    def move(self, event):
        if event.type == KEYDOWN:
            self.lastMove = (self.x, self.y)
            if event.key == K_LEFT:
                self.x -= 10
            if event.key == K_RIGHT:
                self.x += 10
            if event.key == K_UP:
                self.y -= 10
            if event.key == K_DOWN:
                self.y += 10

    def moveBack(self):
        self.x = self.lastMove[0]
        self.y = self.lastMove[1]
```

Initialisation du jeu dans la fenêtre Desktop :

```
pygame.init()
WIDTH = 1024
HEIGHT = 720
pygame.display.set_caption('Sokoban')
MAPS = [
    ['.', '.', '.', '.', 'W', 'W', 'W', 'W', 'W',
     '.', '.', '.', '.', '.', '.', '.', '.'],
    ['.', '.', '.', '.', 'W', '.', '.', '.', 'W',
     '.', '.', '.', '.', '.', '.', '.', '.'],
    ['.', '.', 'W', 'W', 'W', '.', 'P', '.', 'W',
     'W', '.', '.', '.', '.', '.', '.', '.'],
    ['.', '.', 'W', '.', '.', '.', '.', 'B', '.',
     'W', '.', '.', '.', '.', '.', '.', '.'],
    ['W', 'W', 'W', '.', 'W', '.', 'W', 'W', '.',
     'W', '.', '.', 'W', 'W', 'W', 'W', 'W'],
    ['W', '.', '.', '.', 'W', '.', 'W', 'W', '.',
     'W', 'W', 'W', 'W', '.', '.', 'H', 'W'],
    ['W', '.', '.', '.', 'B', '.', '.', 'H', 'W',
     '.', '.', '.', '.', '.', 'H', 'W'],
    ['W', 'W', 'W', 'W', '.', '.', '.', 'H', 'W',
     'B', '.', '.', '.', '.', 'H', 'W'],
    ['.', '.', 'W', 'W', 'B', 'W', 'W', 'W',
     '.', '.', 'W', 'W', '.', '.', 'H', 'W'],
    ['.', '.', 'W', 'W', 'W', 'W', 'W', 'W',
     '.', '.', 'W', 'W', 'W', 'W', 'W'],
    ['.', '.', 'W', 'W', 'W', 'W', 'W', 'W',
     '.', '.', 'W', '.', '.', '.', '.'],
    ['.', 'W', 'W', '.', '.', '.', '.', '.',
     'W', 'W', '.', '.', '.', '.', '.'],
]
screen = pygame.display.set_mode((WIDTH, HEIGHT))
```

# DOSSIER PROFESSIONNEL <sup>(DP)</sup>

Compétences REAC couvertes par ce projet :

- Développer une interface utilisateur de type desktop

## 2. Précisez les moyens utilisés :

OS : MacOS

Language (Librairies) : Python (Pygames)

## 3. Avec qui avez-vous travaillé ?

Seul

## 4. Contexte

Nom de l'entreprise, organisme ou association ▶ La Plateforme\_

Chantier, atelier, service ▶ Sokoban

Période d'exercice ▶ Du : 07/03/2023 au : 08/03/2023

## 5. Informations complémentaires (facultatif)

## Activité-type 1 Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Exemple n° 3 - Quiz Connect

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

Dans le cadre du projet pour l'association le Poivre de l'espoir, nous avons développé un panel Admin web en **NextJs**.

Ce projet valide les compétences :

- Développer des composants d'accès aux données
- Développer la partie front-end d'une interface utilisateur
- Développer la partie back-end d'une interface utilisateur

Cette interface contient un accès à un **CRUD** pour gérer les étudiants, les cours, les forums avec leurs messages, les thèmes et les QCMs.

On a choisi NextJs parce qu'il nous permet de prendre en charge le rendu côté serveur, la génération statique et le rendu côté client dans une seule application. Il intègre également un routeur, il nous suffit de créer un fichier dans le dossier **page**.

Si on veut aller sur la route `"/users"`, il faut créer un fichier `users.js`. Et quand on veut faire une page élément, par exemple ici, on veut modérer les messages liés à un Forum en allant sur la route `messages/id_topic`, il va falloir créer un dossier `messages` et dedans un fichier `[id].js`. Grâce à la méthode `getServerSideProps()`, on va pouvoir récupérer `id_topic` dans l'url. Cette méthode récupère les données pour chaque requête.

```
export async function getServerSideProps(context) {  
  return {  
    props: {  
      id: context.query.id_topic  
    }  
  }  
}
```

```
const [messagesItem, setMessagesItem] = useState([]);

useEffect(() => {
  getTopicMessage(props.id_topic).then((data) => {
    setMessagesItem(data.Messages);
  });
}, []);
```

On va récupérer les messages liés au forum via un appel à l'API et l'afficher côté front. On le fait dans un **useEffect** parce qu'on récupère la donnée en asynchrone puis on la stocke dans un **state**. La fonction côté front API se présente comme tel :

```
import axiosInstance from '../config'

export const getTopicMessage = async (topicID) => {
  const { data } = await axiosInstance.get(`topics/${topicID}?include=Message`)
  return data
}
```

On a créé une instance axios sur une Url précisé dans le fichier de fichier config.js

```
import axios from "axios";

const axiosInstance = axios.create({
  baseURL: process.env.NEXT_PUBLIC_API_URL,
  responseType: "json",
});

axiosInstance.interceptors.request.use(
  function (request) {
    const token = localStorage.getItem("user");
    if(token){
      request.headers.Authorization= `Bearer ${token}`
    }
    return request;
  },
);
export default axiosInstance;
```

Le point d'entrée de l'API est stocké dans une variable d'environnement, pour optimiser le code et

## DOSSIER PROFESSIONNEL <sup>(DP)</sup>

n'avoir qu'un seul endroit où changer les paramètres. De ce fait, on a créé un fichier .gitignore pour éviter d'éventuelles failles de sécurité en le pushant sur github.

Le panel admin se présente comme tel avec différents onglets où l'on peut modérer :

- les étudiants (créer / modifier / supprimer leur compte)
- les cours (créer / modifier / supprimer le cours)
- les forums (créer / modifier / supprimer les messages)
- les thèmes (créer / modifier / supprimer les thèmes)
- les QCM (créer / modifier / supprimer les questions)

### 2. Précisez les moyens utilisés :

J'ai utilisé :

- VScode
- Next js
- Node js
- sequelize
- les librairies react utilisées : MUI, Jwt-decode, React-to-print, axios

### 3. Avec qui avez-vous travaillé ?

J'ai collaboré avec Fatima Elmouhine et Ifanl Ibrahim durant ce projet.

### 4. Contexte


Nom de l'entreprise, organisme ou association ➤ *L'association Le Poivre de l'espoir*

Chantier, atelier, service ➤ *Quiz Connect*

Période d'exercice ➤ Du : *01/01/2023* au : *31/01/2023*

### 5. Informations complémentaires (facultatif)

# DOSSIER PROFESSIONNEL (DP)







Quiz Connect

- Étudiants
- Cours
- Forums
- Thèmes
- QCM

DÉCONNEXION

## Liste des étudiants



<input type="checkbox"/>	ID	Prénom	Nom	E-mail	Rôle	Photo de profil	Date de création	Date de modification
<input type="checkbox"/>	1	admin2	admin	admin@admin.com	<a href="#">ROLE_STUDENT</a> <a href="#">ROLE_ADMIN</a>		27/05/2023	14/06/2023 15:1'
<input type="checkbox"/>	2	user	user	user@user.com	<a href="#">ROLE_TUTOR</a>		27/05/2023	27/05/2023 11:0'
<input type="checkbox"/>	3	user2	user2	user2@user2.com	<a href="#">ROLE_STUDENT</a>		27/05/2023	27/05/2023 11:0'

## Liste des étudiants

<input type="checkbox"/>	ID	Prénom
<input type="checkbox"/>	1	admin
<input type="checkbox"/>	2	user
<input type="checkbox"/>	3	user2

### Ajouter un utilisateur

Prénom

Nom

E-mail

Mot de passe

Confirmer le mot de passe

Rôle

[ENREGISTRER](#) [ANNULER](#)







Photo de profil	Date de création	Date de modification
	27/05/2023	27/05/2023 11:01:02
	27/05/2023	27/05/2023 11:01:02
	27/05/2023	27/05/2023 11:01:02



### Activité-type 2 Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité

Exemple n° 1 - Quizz Connect

---

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

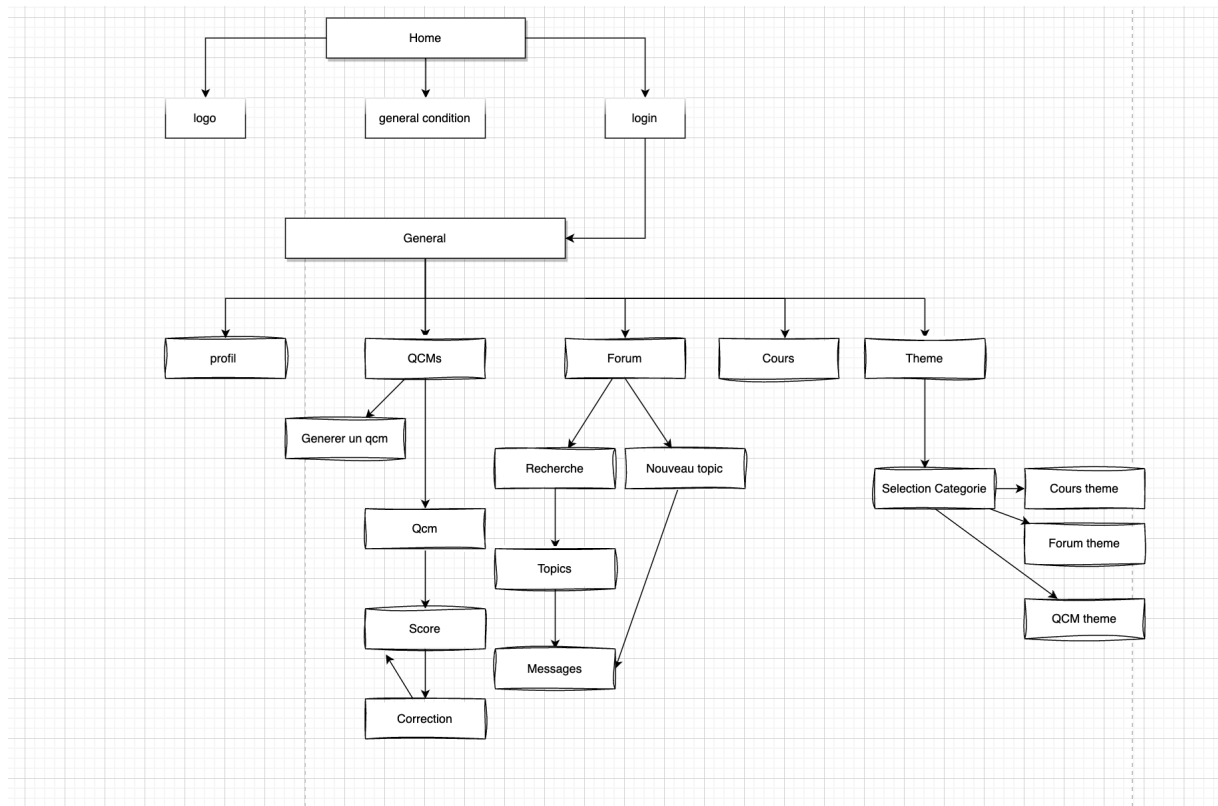
Après avoir pris en note dans le cahier des charges les besoins de l'association (voir exemple 1), il a été décidé de réaliser une application en **React Native** qui serait composée :

- d'un espace de connexion
- d'un espace profil
- d'un espace forum
- d'un espace cours
- d'un espace qcm
- d'un espace statistique

Ce projet me permet de valider les compétences :

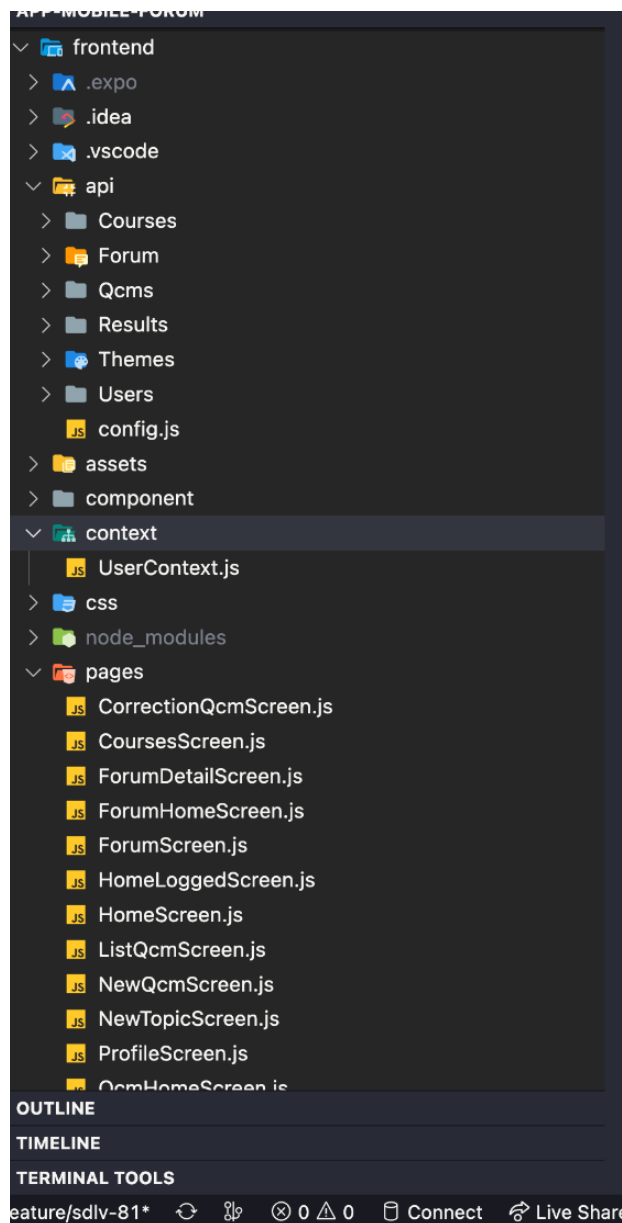
- **Développer une application mobile**
- **Développer des composants métier**
- **Concevoir une application**
- **Construire une application organisée en couches**

Nous avons architecturé l'application mobile comme représenté dans le schéma suivant :



On ne peut pas accéder au reste de l'application si l'on n'est pas connecté.

Nous avons utilisé **Expo**, qui est à la fois un **framework** et une plateforme qui simplifient la création et le déploiement d'applications mobiles avec **React Native**. Expo embarque de nombreux outils utiles et des librairies natives pour React Native. Il gère aussi la mise à jour de ces librairies.



Côté Frontend, nous avons créé différents dossiers afin de mieux nous organiser . Dans le dossier **api**, on aura tout ce qui est appelé api qui communique avec la base de données. Dans le fichier config, on a mis en place une instance **Axios** qui nous permet :

- de personnaliser les paramètres pour toutes les requêtes effectuées à partir de cette instance,
- d'éviter la répétition de code et de rendre notre code plus lisible.

```
import axios from 'axios'
import * as SecureStore from 'expo-secure-store';
import API_IP from '../env.js'

const axiosInstance = axios.create({
  baseURL: `${API_IP}/api/`, responseType: 'json',
})

axiosInstance.interceptors.request.use(function (request) {
  SecureStore.getItemAsync('token').then((token) => {
    request.headers.Authorization = `Bearer ${token}`
  })
  return request
}, function (error) {
  console.log(error);
  return Promise.reject(error)
})

export default axiosInstance
}
```

Par le biais du **Secure Store** d'expo, on a pu stocker le token d'authentification et ainsi le passer en entête de nos requêtes. Les requêtes sont faites avec la librairie Axios comme expliqué dans l'exemple précédent. On a également mis en place **React Context** qui nous a permis de rendre les données de l'utilisateur accessibles à l'ensemble des composants de notre application.

```
export const UserContext = createContext();

const UserContextProvider = (props) => {

  const [token, setToken] = useState(null);
  const [isLoggedIn, setIsLoggedIn] = useState(false);

  const decodedToken = token ? jwtDecode(token) : null;
```

Dans un premier temps, on déclare un `UserContext` avec la fonction `createContext()` de react pour pouvoir utiliser le context provider et faire passer les informations de notre utilisateur.

```
<UserContext.Provider
  value={{userDetails, setUserDetails, userAuthentication, isLoggedIn, setIsLoggedIn, userLogout}}
>

  {props.children}

</UserContext.Provider>
```

On passe également en paramètre du provider :

- la variable `isLoggedIn` qui nous permet de savoir si l'utilisateur est connecté à partir du token d'authentification
- la fonction `setUserDetails` qui nous sert à mettre à jour le context depuis d'autres pages.
- `userAuthentication` qui permet à l'utilisateur de se connecter

## Exemple de composants métier

```
<LinearGradient
  colors={['purple', '#02254F', '#2D84EA']}
  style={styles.containerGradient}
>
  <Image source={require("../assets/logo_fond.png")} style={styles.bgTop} />
  <View style={styles.profileImgContainer}>...
  </View>
  <Text style={styles.sectionTitle}>Mes derniers cours</Text>
  <ScrollView style={{ display: "flex", flexDirection: "column" }}>
    <View style={styles.sectionLatestCourse}>
      {courses.length !== 0 &&
        courses.map((course, i) => {
          return <CourseCardComponent key={i} course={course} />;
        })
      }
    </View>
  </ScrollView>
</LinearGradient>
```

## 2. Précisez les moyens utilisés :

J'ai utilisé :

- VScode

# DOSSIER PROFESSIONNEL <sup>(DP)</sup>

- React Native
- Node js
- sequelize
- Expo
- les librairies react utilisées : react native paper, Jwt-decode, axios

## 3. Avec qui avez-vous travaillé ?

Fatima Elmouhine et Ifanl Ibrahim

## 4. Contexte

Nom de l'entreprise, organisme ou association ▶ *La Plateforme*

Chantier, atelier, service ▶ *Quizz Connect*

Période d'exercice ▶ Du : 01/01/2023 au : 31/01/2023

## 5. Informations complémentaires (facultatif)

## Activité-type 3

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

Exemple n° 1 - Quizz Connect

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

Pour ce projet, nous avons créé une base de données pour répondre à la demande du client.

Ce projet valide les compétences :

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

Sur cette base de données, on va pouvoir enregistrer les différentes informations liées :

- à l'utilisateur
- aux cours
- aux QCM et questions liées
- aux qcm répondus par l'utilisateur
- aux thèmes
- au forum et sujet associés
- aux messages des sujet

On a utilisé node Js ainsi que la librairie sequelize qui prend en charge MySQL et offre une prise en charge solide des transactions et des relations. Sequelize est un ORM ("Object Relational Mapping") qui sert à mettre à disposition des classes objet permettant de manipuler les bases de données relationnelles.

Nous nous sommes appuyés sur la méthode “Merise” pour concevoir notre base de données SQL. On a commencé par réaliser sur l'outil **draw.io**, le **MCD** (modèle conceptuel de données) et enfin le **MLD** (modèle logique de données) comme on peut le voir en Annexe 1 et 2.

```
const { Sequelize, DataTypes } = require('sequelize')
const sequelize = new Sequelize('mysql::memory:');

module.exports = (sequelize) => {
  const User = sequelize.define('User', {
    // Model attributes are defined here
    firstName: {
      type: DataTypes.STRING,
```

Dans le dossier model, on retrouve un fichier pour chaque table comme par exemple pour ce fichier user.

On définit chaque champ et lui spécifie son type et s'il peut être nul en base de données.

```
const sequelize = new Sequelize({
  dialect: 'mysql',
  username: process.env.DB_USERNAME,
  password: process.env.DB_PASSWORD,
  host: process.env.DB_HOST,
  port: process.env.DB_PORT,
  database: process.env.DB_NAME,
});
```

Pour se connecter à la base de données, on a dû créer une instance Sequelize et on lui passe en paramètre les informations de connexion qui sont dans notre .env

```
function association(sequelize){

  const {User, Qcm, UserQcm, Topic, Message, Question,
  sequelize.models;

  User.hasMany(Message, {foreignKey: 'id_user'});
  Topic.hasMany(Message, {foreignKey: 'id_topic'});
  Message.belongsTo(Topic, {foreignKey: 'id_topic'});

  Message.belongsTo(User, {foreignKey: 'id_user'});
  User.hasMany(UserQcm, {foreignKey: 'id_user'});

  Qcm.hasMany(UserQcm, {foreignKey: 'id_qcm'});
  UserQcm.belongsTo(Qcm, {foreignKey: 'id_qcm'});
```

Sequelize nous permet de faire des associations.

Il ya 4 types d'associations :

- HasOne
- BelongsTo
- HasMany
- BelongsToMany

L'étape MCD / MLD nous a aidé pour repérer plus facilement quel type de relation était le plus adapté en fonction des cardinalités .

## 2. Précisez les moyens utilisés :

J'ai utilisé :

- VScode
- Node js
- sequelize
- MySQL
- draw.io



## DOSSIER PROFESSIONNEL <sup>(DP)</sup>

### 3. Avec qui avez-vous travaillé ?

Ifanl Ibrahim et Fatima Elmouhine

### 4. Contexte

Nom de l'entreprise, organisme ou association ▶ *L'association Le Poivre de l'espoir*

Chantier, atelier, service ▶ *Quizz Connect*

Période d'exercice ▶ Du : *01/01/2023* au : *31/01/2023*

### 5. Informations complémentaires *(facultatif)*

## Activité-type 3 Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

Exemple n° 2 - Portfolio

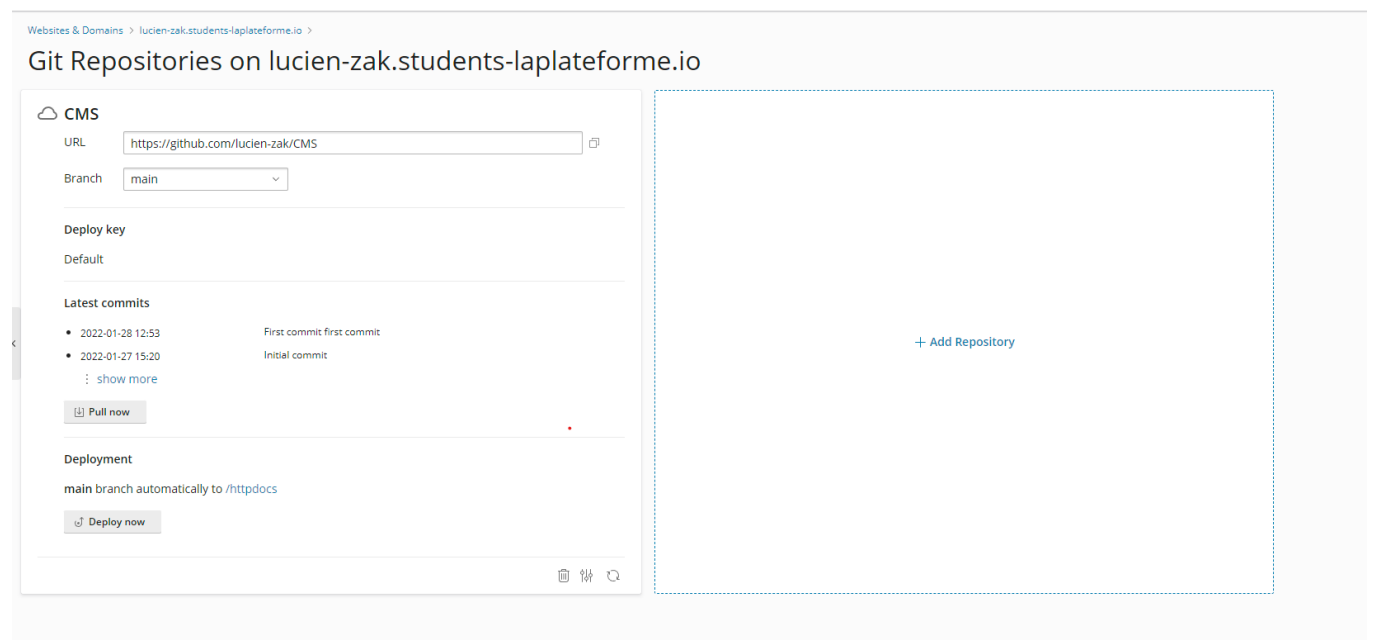
### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Dans le cadre de ma recherche d' alternance, j'ai pu réaliser un portfolio que j'ai pu déployé sur un hébergeur (plesk)

Cet exemple me permet de valider la compétence :

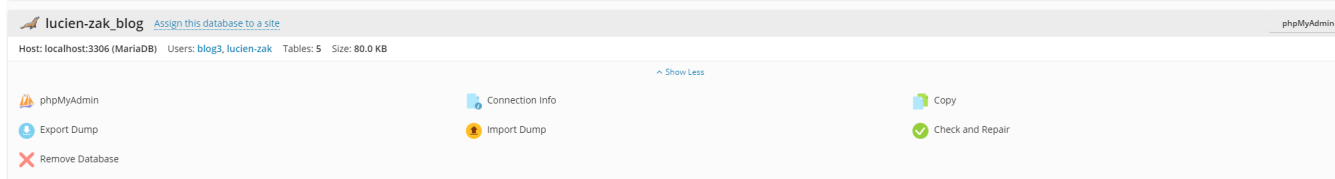
- **Préparer et exécuter le déploiement d'une application**

Je synchronise les fichiers dans le dossier /httpdocs avec mon la branche main du sur le Git pour pousser les fichiers à chaque commit :



Ensuite, j'ajoute ma base de données MySQL via l'onglet **Database**. J'ai créé un utilisateur pour me connecter. Puis j'ai changé les informations de connexion dans **mes variables d'environnement**.

# DOSSIER PROFESSIONNEL (DP)



J'indique le point d'entrée de mon domaine, mon site est maintenant en ligne

## 2. Précisez les moyens utilisés :

J'ai utilisé :

- Plesk
- Wordpress

## 3. Avec qui avez-vous travaillé ?

J'ai travaillé seul sur ce projet.

## 4. Contexte

Nom de l'entreprise, organisme ou association ▶ *La plateforme*

Chantier, atelier, service ▶ *Formation*

Période d'exercice ▶ Du : *20/09/2022* au : *30/10/2022*

## 5. Informations complémentaires (facultatif)

---

## DOSSIER PROFESSIONNEL <sup>(DP)</sup>

---

---

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

*(facultatif)*

Intitulé	Autorité ou organisme	Date

### Déclaration sur l'honneur

---

Je soussigné(e) [prénom et nom] **Lucien Zak**,  
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 **Bras** le **30/07/2024**  
pour faire valoir ce que de droit.

Signature : Zak Lucien

### Documents illustrant la pratique professionnelle

*(facultatif)*

Intitulé
Cliquez ici pour taper du texte.

---

# DOSSIER PROFESSIONNEL <sup>(DP)</sup>

---

## ANNEXES

*(Si le RC le prévoit)*