



# DOSSIER PROFESSIONNEL (DP)

*Nom de naissance* ▶ *Meca*  
*Nom d'usage* ▶  
*Prénom* ▶ *Geoffrey*  
*Adresse* ▶ *9 allée Courbet 13470 Carnoux*

## Titre professionnel visé

*Concepteur Développeur D'applications*

### 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

<b>Concevoir et développer des composants d'interface utilisateur</b>	<b>p.</b>	<b>5</b>
- Développement d'une application forum sur le code – CodeHub	p. p.	5
- Intitulé de l'exemple n° 2	p. p.	
<b>Concevoir et développer la persistance des données</b>	<b>p.</b>	<b>7</b>
- Développement d'une application forum sur le code – CodeHub	p. p.	7
- Intitulé de l'exemple n° 2	p. p.	
<b>Concevoir et développer une application multicouche</b>	<b>p.</b>	<b>10</b>
- Développement d'une application forum sur le code – CodeHub	p. p.	10
- Intitulé de l'exemple n° 2	p. p.	
<b>Titres, diplômes, CQP, attestations de formation</b> <i>(facultatif)</i>	<b>p.</b>	<b>15</b>
<b>Déclaration sur l'honneur</b>	<b>p.</b>	<b>16</b>
<b>Documents illustrant la pratique professionnelle</b> <i>(facultatif)</i>	<b>p.</b>	<b>17</b>
<b>Annexes</b> <i>(Si le RC le prévoit)</i>	<b>p.</b>	<b>18</b>

# **EXEMPLES DE PRATIQUE PROFESSIONNELLE**

## Activité-type 1 Développer des composants d'interface

Exemple n°1 - Développement d'une application forum sur le code – CodeHub

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

Dans le projet, l'une de mes tâches était de mettre en place la navigation de l'application. Pour ce faire, j'ai utilisé la librairie react navigation qui m'a permis de mettre en place un menu drawer afin de pouvoir naviguer entre les différentes pages de l'application.

### 2. Précisez les moyens utilisés :

Tout d'abord, j'ai installé React Navigation dans mon projet. J'ouvre mon terminal et exécute la commande suivante :

```
npm install @react-navigation/native
```

Ensuite, j'ai ajouté les dépendances spécifiques à la navigation en utilisant la commande suivante :

```
npm install react-native-reanimated react-native-gesture-handler react-native-screens  
react-native-safe-area-context @react-native-community/masked-view
```

Maintenant que j'ai installé React Navigation, j'ai créé un nouveau fichier pour mon Drawer Navigator. Je vais l'appeler "AppNavigator.js" et l'ouvrir dans mon éditeur de code.

Dans ce fichier, je vais importer les composants nécessaires de React Navigation et créer mon Drawer Navigator. Voici un exemple de code pour le fichier "AppNavigator.js" :

```
import React from 'react';  
import { createDrawerNavigator } from '@react-navigation/drawer';  
import { NavigationContainer } from '@react-navigation/native';  
  
import HomeScreen from './HomeScreen';  
import SettingsScreen from './SettingsScreen';  
  
const Drawer = createDrawerNavigator();  
  
const AppNavigator = () => {  
  return (  
    <NavigationContainer>  
      <Drawer.Navigator>  
        <Drawer.Screen name="Accueil" component={HomeScreen} />  
        <Drawer.Screen name="Paramètres" component={SettingsScreen} />  
      </Drawer.Navigator>  
    </NavigationContainer>  
  )  
}
```

```
);  
};
```

```
export default AppNavigator;
```

Dans cet exemple, j'ai créé un Drawer Navigator avec deux écrans : "HomeScreen" et "SettingsScreen". Il est possible d'ajouter autant d'écrans que souhaité en utilisant la méthode Drawer.Screen.

Maintenant que j'ai créé mon Drawer Navigator, je vais l'utiliser dans mon composant racine de l'application. Je vais ouvrir mon fichier "App.js" et le modifier comme suit :

```
import React from 'react';  
import AppNavigator from './AppNavigator';
```

```
const App = () => {  
  return <AppNavigator />;  
};
```

```
export default App;
```

Maintenant, lorsque j'exécute mon application, je vois mon menu Drawer fonctionner correctement. Lorsque je fais glisser mon doigt de gauche à droite sur l'écran, le menu Drawer s'ouvre et affiche les écrans "Accueil" et "Paramètres".

### 3. Avec qui avez-vous travaillé ?

J'ai travaillé seul sur cette activité

### 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)

## Activité-type 2 Développer la persistance des données

**Exemple n° 1** - Développement d'une application forum sur le code – CodeHub avec API Platform.

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

Dans ce projet, nous avons utilisé API Platform, un framework web utilisé pour générer des API REST, se basant sur le patron de conception MVC, tout en offrant des fonctionnalités avancées telles que la documentation automatique de l'API, la validation des données, la sécurité, etc. La partie serveur du framework est écrite en PHP et basée sur le framework Symfony, tandis que la partie client est écrite en JavaScript et TypeScript.

Plusieurs routes étaient nécessaires afin de sécuriser et de contrôler la persistance des données. Dans l'application un fichier nommé api.js concentre toutes les requêtes en utilisant la librairie Axios.

### 2. Précisez les moyens utilisés :

Pour créer les 3 entités j'ai utilisé la commande:

`bin/console make:entity User` pour l'entité user car elle permet de mieux gérer la sécurité et l'authentification d'un utilisateur.

Et la commande `bin/console make:entity "Nom de l'entité"` pour créer les 2 autres.

Pour configurer les entités, j'ouvre les fichiers de classe générés pour chaque entité et spécifie les propriétés (champs) dont j'ai besoin. Par exemple, pour l'entité User, je peux ajouter des propriétés comme `id`, `username`, `email`, etc., en les annotant avec les annotations Doctrine appropriées. Je peux également définir les relations entre les entités. Par exemple, pour créer une relation entre User et Article, j'ajoute une propriété `OneToMany` dans la classe User, qui référence les articles écrits par l'utilisateur.

Pour générer la base de données,

j'utilise la commande `doctrine:schema:update --force` pour générer les tables correspondant à mes entités dans la base de données.

Symfony Platform utilise Doctrine ORM pour gérer les opérations de base de données, donc la configuration de la connexion à la base de données doit être correctement spécifiée dans le fichier `config/packages/doctrine.yaml`.

Pour effectuer une requête vers l'API, une méthode request a été créée qui gère les réponses des requêtes.

```
const request = async (method, url, data, callback) => {
  await getJwtToken();
  header = { 'Content-type': 'application/json', }
  if (method == "patch") {
    header = { 'Content-type': 'application/merge-patch+json' }
  }

  return api({
    method: method,
    url: url,
    data: data,
    headers: header
  }).then(res => {
    return callback(res);
  })
  .catch(error => {
    console.log(error.message)
    console.log(`Erreur ${error.response.data.code}`)
    console.log(error.response.data.message)
    if (error.response.data.message == 'Expired JWT Token') {
      SecureStore.deleteItemAsync('jwt').then(() => {
      })
      Alert.alert('Votre session a expiré', 'Veuillez vous re-connecter pour continuer', [
        { text: 'OK', onPress: () => { } }
      ])
    }
    return null
  })
  return callback(error.response)
};
```

La fonction request ne requiert que 3 paramètres, la première consiste à définir la méthode (get/post/patch/delete), la seconde la route, la troisième les paramètres de la requête si il y en a, et enfin elle renvoie en callback la réponse (ou l'erreur).

Pour ensuite appeler votre requête dans n'importe quelle fichier, il suffit d'importer le fichier api.js ainsi que la méthode que vous avez définis.

Si paramètre il y a, il suffit de les envoyer à la méthode en premier temps, et ensuite récupérer la réponse en fonction fléché anonymes.

### 3. Avec qui avez-vous travaillé ?

Rany Alo - Christophe Calmes - Mathieu Ruiz



---

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

---

## 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)

## Activité-type 3 Développer une application

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

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

Le forum Code Hub est une application mobile Android, développée pour permettre aux utilisateurs de publier des articles et de commenter ceux des autres.

L'application permet aux visiteurs sans compte de consulter les articles et les commentaires, mais ils ne peuvent pas en ajouter eux-mêmes.

Les utilisateurs avec un compte peuvent ajouter des articles et des commentaires. Ils ont également la possibilité de modifier leur profil et de supprimer leur compte si nécessaire.

Les administrateurs ont les mêmes droits qu'un utilisateur standard, mais disposent également de fonctionnalités supplémentaires.

Ils peuvent modifier, supprimer et ajouter des utilisateurs au forum.

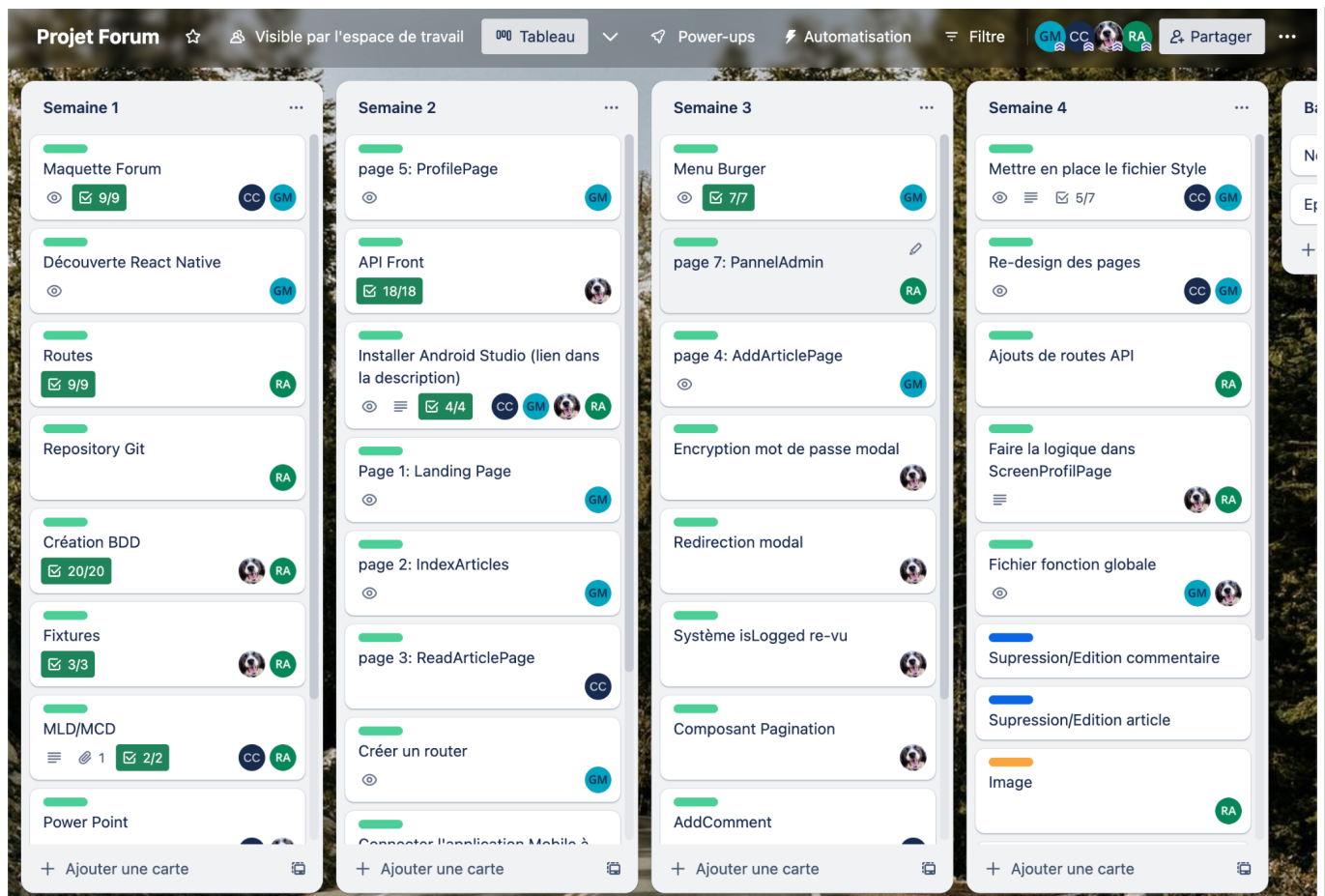
De plus, ils peuvent supprimer des articles ou des commentaires en cas de violation des règles du forum.

Les tâches réalisées :

► La première étape dans la réalisation d'un projet informatique est de construire une équipe solide de développeurs qui peuvent collaborer efficacement pour atteindre les objectifs du projet, nous avons constitué une équipe de quatre développeurs et utilisé les méthodes de **Kanban** et **Trello** pour le suivi des tâches.

Le projet a été divisé en petites tâches pour faciliter le travail et assurer une bonne gestion du temps. Nous avons également utilisé **Github** pour le suivi des versions et le partage du code.

# DOSSIER PROFESSIONNEL (DP)

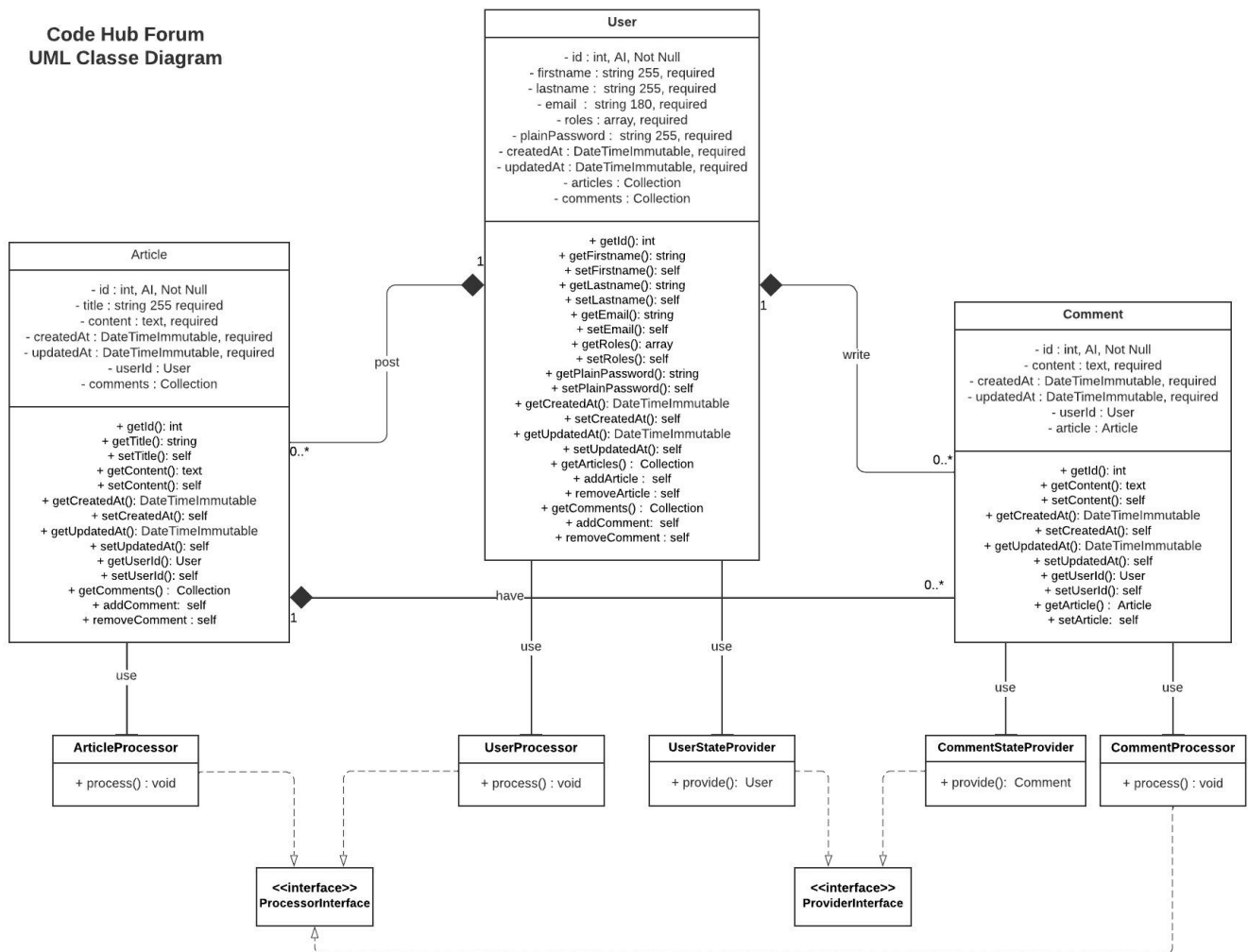


► L'analyse des besoins est une étape cruciale dans la conception d'une application, car elle permet de déterminer les exigences et les attentes des utilisateurs. Dans le cadre de notre projet, nous avons reçu un cahier des charges de notre école qui a servi de base pour identifier les fonctionnalités à implémenter.

Cette analyse nous a permis de comprendre les besoins des utilisateurs, d'anticiper les problématiques et d'élaborer une application adaptée.

► Création de l'UML Diagramme de classes : Le diagramme de classes UML permet de visualiser la structure de notre modèle de données et de faciliter la communication entre les membres de l'équipe de développement.

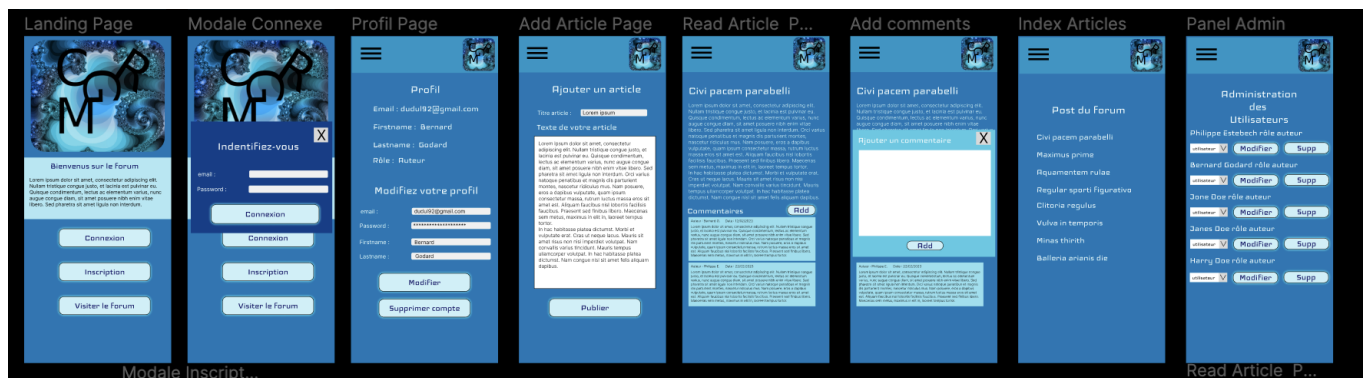
**Code Hub Forum**  
**UML Classe Diagram**



► Développement de notre application mobile Code Hub organisée en quatre couches :

1. La couche de présentation : Cette couche comprend toutes les vues de notre application, c'est-à-dire les écrans que les utilisateurs voient et avec lesquels ils interagissent. Cette couche est constituée des écrans React Native de votre application. Les écrans permettent aux utilisateurs de voir les articles, les commentaires et les utilisateurs (que pour l'admin), de créer de nouveaux articles et commentaires, et de mettre à jour ou supprimer les articles et les commentaires existants.

# DOSSIER PROFESSIONNEL (DP)



2. La couche de logique d'application : Cette couche contient la logique métier de notre application, c'est-à-dire les règles qui déterminent comment les données sont manipulées et gérées. Cette couche est principalement gérée par l'API Platform et Symfony, qui fournissent des fonctionnalités pour gérer les entités (User, Article, Comment) et effectuer des opérations CRUD (créer, lire, mettre à jour, supprimer) sur ces entités. La couche de logique d'application utilise également des services pour effectuer des tâches spécifiques, telles que l'authentification et l'autorisation des utilisateurs.

3. La couche d'accès aux données : Cette couche est responsable de la communication avec la source de données, dans notre cas la base de données. Cette couche est également gérée par l'API Platform et Symfony, qui utilisent Doctrine pour communiquer avec la base de données et mapper les données en entités PHP.

4. La couche de services : Cette couche comprend les différents services de notre application, qui fournissent des fonctionnalités spécifiques à différentes parties de notre application. Dans notre cas, nous avons des services pour gérer l'authentification des utilisateurs et une service qui appelle l'API OpenAi pour générer un article depuis son titre. Ces services sont également gérés par l'API Platform et Symfony.

## ► Déploiement de notre application :

Nous avons utilisé Plesk pour héberger notre API. Plesk est apprécié pour sa facilité d'utilisation, sa flexibilité et sa robustesse. Il est doté d'une interface graphique intuitive qui permet de gérer facilement les différents aspects du serveur Web.

## 2. Précisez les moyens utilisés :

- Visual Studio Code (VSCode) est un éditeur de code source gratuit.
- PhpStorm est un environnement de développement intégré (IDE) conçu spécifiquement pour le développement en PHP. J'ai utilisé PhpStorm pour créer l'api symfony car il offre une gamme de fonctionnalités avancées pour faciliter le processus de développement.
- Symfony et API Platform : Symfony est un framework PHP open source. API Platform est une extension de Symfony qui permet de créer des API web RESTful. Nous avons utilisé Symfony et API Platform pour créer l'API backend de l'application.

## DOSSIER PROFESSIONNEL (DP)

- ▶ React Native est une bibliothèque JavaScript open source développée par Facebook pour créer des applications mobiles multiplateformes pour iOS et Android en utilisant une seule base de code.
- ▶ GitHub : C'est une plateforme de développement logiciel qui permet de stocker et de partager du code source. Nous avons utilisé GitHub pour le versionnage du code source et la collaboration entre les membres de l'équipe.
- ▶ Trello : C'est un outil de gestion de projet en ligne basé sur la méthode Kanban. Nous avons utilisé Trello pour organiser les différentes tâches à effectuer pour réaliser le projet et pour suivre leur progression.
- ▶ Lucidchart : est un outil en ligne de création de diagrammes et de visualisations de données. Nous avons utilisé Lucidchart pour créer les diagrammes UML.

### 3. Avec qui avez-vous travaillé ?

Rany Alo - Christophe Calmes - Mathieu Ruiz

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

---

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

*(facultatif)*

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

### Déclaration sur l'honneur

---

Je soussigné(e) [prénom et nom] [Cliquez ici pour taper du texte.](#) ,  
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 [Cliquez ici pour taper du texte.](#) le [Cliquez ici pour choisir une date](#)  
pour faire valoir ce que de droit.

Signature :



## Documents illustrant la pratique professionnelle

*(facultatif)*

Intitulé
Cliquez ici pour taper du texte.

---

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

---

### ANNEXES

*(Si le RC le prévoit)*