



DOSSIER PROFESSIONNEL (DP)

<i>Nom de naissance</i>	➤ SYLLA
<i>Nom d'usage</i>	➤ SYLLA
<i>Prénom</i>	➤ Ibrahim
<i>Adresse</i>	➤ 31 RUE DANTON 13003 Marseille

Titre professionnel visé

*Titre RNCP 6 :
Concepteur Developpeur Application*

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 ^(DP)

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 la persistance des données en intégrant les recommandations de sécurité			p.	5
▶ Projet Blog en Symfony	p.	p.		5
▶ Intitulé de l'exemple n° 2	p.	p.		
▶ Intitulé de l'exemple n° 3	p	p.		
Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité			p.	14
▶ Calculator 9000	p.	p.		14
▶ My Sokoban	p.	p.		20
▶ Intitulé de l'exemple n° 3	p	p.		
Concevoir et développer une application multicouches répartie en intégrant les recommandations de sécurité			p.	21
▶ Akiba Town	p.	p.		25
▶ Tests de l'application Akiba Town (fonctionnels et unitaires)	p.	p.		43
▶ Intitulé de l'exemple n° 3	p	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.	

EXEMPLES DE PRATIQUE PROFESSIONNELLE

Activité-type 1 Concevoir et développer la persistance des données en intégrant les recommandations de sécurité

Exemple n°1 - Blog (Symfony)

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

Dans le cadre de ma formation de Concepteur développeur d'application j'ai dû réaliser un Blog à l'aide du Framework de PHP: **SYMFONY**.

Symfony est un ensemble de composants PHP ainsi qu'un framework MVC libre écrit en PHP. Il fournit des fonctionnalités modulables et adaptables qui permettent de faciliter et d'accélérer le développement d'un site web.

Il est **open source** et permet de créer des sites ou applications Web complexes, robustes, fiables, évolutifs, maintenables et performants.

Comme une grande « bibliothèque de solutions efficaces et prêtes à l'emploi », Symfony regroupe de nombreux composants qui facilitent le développement des produits digitaux. L'utilisation d'un tel framework est de :

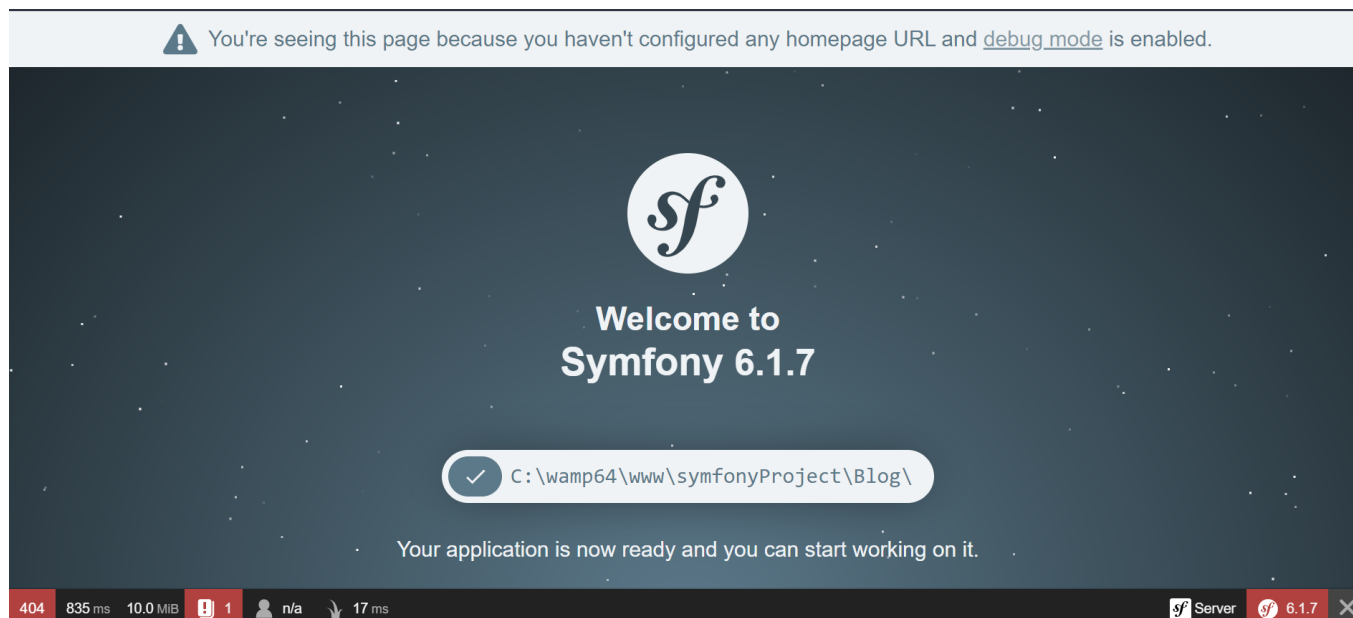
- ❖ Limiter le codage fastidieux,
- ❖ Limiter le codage sans valeur ajoutée,
- ❖ Réduire le temps de développement,
- ❖ Guider le développeur et fiabiliser son travail.

Ce projet m'a permis d'apprendre une manière révolutionnaire de développer une webApp, d'où l'avantage d'utiliser symfony. La base de données est entièrement créée à l'aide de lignes de commandes (Terminal). Le projet est généré avec les dossiers et toutes les dépendances nécessaires (composer, twig etc..) sont installées avec une ligne de commande:

Génération du projet:

- ❖ **composer create-project symfony/website-skeleton my_project_name** : crée un nouveau projet Symfony (avec "**website-skeleton**" on précise que l'on veut une architecture pour une WebApp).
- ❖ **symfony server:start** : lance le serveur web intégré pour le développement sur un port libre du pc (ex: <https://127.0.0.1:8000>).

Si au lancement du serveur il y a cette page:



C'est que le projet a été correctement généré. On peut y voir la version de symfony utilisée (6.1.7 pour ce projet).

Construction de la Base de données:

- ❖ **symfony make:entity** : génère une nouvelle entité.

A l'aide de cette commande on génère un modèle de donnée (table), ce qui va représenter notre base de données à l'intérieur du projet. on utilise la même commande pour toutes les tables dont nous avons besoin et on se laisse guider par les instructions affichées dans le terminal (Voir Entité en annexe: "lien").

- ❖ **symfony doctrine:migrations** : génère un nouveau fichier de migration (Voir annexe: "lien").

Avec cette commande on génère un fichier de migration qui va permettre en quelque sorte d'enregistrer la version actuelle de la base de données pour que l'on puisse garder un historique et, en cas de problème, revenir à la version souhaitée.

- ❖ **symfony doctrine:migrations:migrate** : exécute la migration.

Avec cette commande on effectue la migration qui est effectuée à l'aide du fichier de migration vers un serveur local de base de données (ex: phpMyAdmin).

- ❖ **symfony make:controller** : génère un nouveau contrôleur.

Avec cette commande on génère un nouveau controller à partir de l'entité choisie pour pouvoir développer les fonctionnalités et aller chercher les données nécessaires auprès des models pour le bon fonctionnement du site.

- ❖ **symfony make:form** : génère un nouveau formulaire.

Avec cette commande on génère un formulaire (pour l'inscription par exemple). Le formulaire est généré et stylisé automatiquement à l'aide de classes Bootstrap (Bootstrap est le style par défaut).

- ❖ **symfony make:user** : génère une nouvelle classe utilisateur avec l'authentification.

Cette commande est très pratique car elle permet de générer un module de connexion complet et sécurisé. Elle génère l'entité Users (la table utilisateurs), le controller pour le traitement de l'inscription et celui pour le traitement de l'authentification.

- ❖ **symfony debug:router** : affiche une liste de toutes les routes enregistrées dans l'application.
- ❖ **symfony cache:clear** : efface le cache de l'application.
- ❖ **symfony security:check** : vérifie les dépendances pour les vulnérabilités de sécurité connues.

Consignes

La base de données devait être mise en place à l'aide de lignes de commande et être composée de :

- ❖ Une table **Users**
- ❖ Une table **Articles**
- ❖ Une table **Comments**

Le blog devait avoir les fonctionnalités suivantes :

- ❖ Se **créer un compte** et **se connecter**
- ❖ **Poster des articles**
- ❖ **Poster des commentaires** sur des articles

Le blog devait contenir :

- ❖ Une page de **login**
- ❖ Une page avec **tous les articles**
- ❖ Une page de **détail pour l'article cliqué**, avec les commentaires associés en dessous et un formulaire pour en rentrer un nouveau
- ❖ Une **page profil** sur laquelle on peut modifier nos informations

Chaque modèle (Entité) :

- Un **controller**
- Un **fichier Twig** de base (qui sera extends), ainsi qu'un fichier twig pour chacune de vos pages (vue utilisateurs).

Lorsqu'on arrive sur le site, on atterrit sur la page d'accueil avec le titre du site et tous les liens dont-on aura besoin pour se déplacer sur le site :

DOSSIER PROFESSIONNEL (DP)

Blog Symfony Home Blog Login Register

Bienvenue sur le Blog
Inscrivez-vous et postez des articles



200 @ app_main 90 ms 6.0 MiB 1 n/a 17 ms

Sf Server 6.1.7

En cliquant sur “Blog” dans la barre de navigation on aura accès à la page où tous les articles sont affichés:

Blog Symfony Home Blog Login Register

ibraa

rtuhijo.

Voir l'article

Baki Hanma

Fils de Yujiro Hanma.

Voir l'article

ibraa

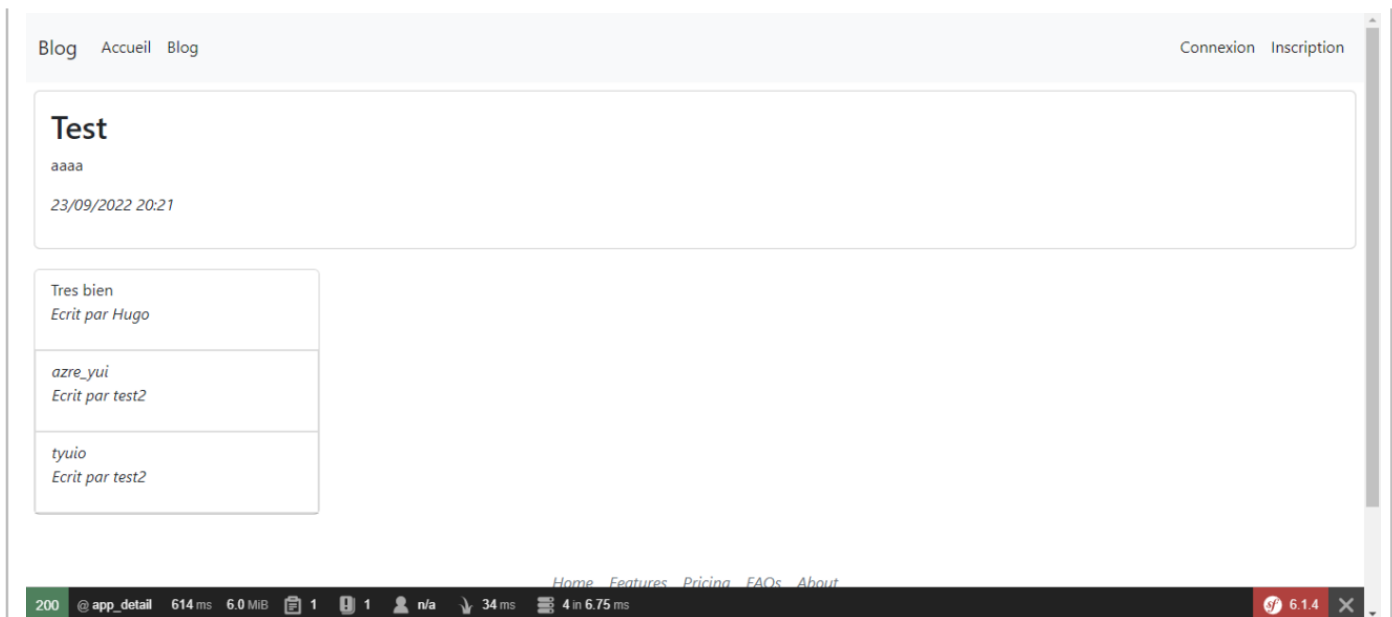
hhhh.

Voir l'article



Si l'on clique sur “Voir l'article” alors on a l'article accompagné de la date à laquelle il a été posté avec le nom de l'auteur accompagné des commentaires liés à l'article:

DOSSIER PROFESSIONNEL (DP)



Dans le cas où l'on souhaite poster un article, alors il faudra se connecter via la page de connexion:

Blog Symphony Home Blog Login Register

Please sign in

Email

ibrahim.sylla27@outlook.fr

Password

.....

Sign in

f t G @ in

Si l'on a pas de compte alors on a la possibilité de s'inscrire en cliquant sur le lien "Register":

DOSSIER PROFESSIONNEL (DP)

Blog Symfony Home Blog Login Register

Inscription

Firstname

Lastname

Email

Password

R g p d consent ☐

Register



200 @ app_register 166 ms 4.0 MiB 1 1 5 n/a 57 ms

Sf Server 6.1.7 X

Une fois connecté, on aura la possibilité de créer un article via le formulaire de création d'article en cliquant sur le lien:

Blog Symfony Home Blog Mon compte Me deconnecter Creer un Articles

info

Title

Article

Valider



200 @ app_articles 283 ms 4.0 MiB 1 1 3 ibrahim.sylla27@outlo... 51 ms 1 in 2.08 ms

Sf Server 6.1.7 X

On a aussi la possibilité de modifier ses coordonnées en cliquant sur l'onglet mon compte:

DOSSIER PROFESSIONNEL (DP)

[Blog](#) [Accueil](#) [Blog](#) [Créer un article](#) [Profil](#) [Deconnexion](#)

Pseudo : test2

E-mail : test@gmail.com

Pseudo

Adresse email

[Modifier](#)

[Modifier Mot de Passe](#)

[Home](#) [Features](#) [Pricing](#) [FAQs](#) [About](#)

© 2022 Company, Inc

200

@ app_user

533 ms

6.0 MiB

1

1

2

test2

39 ms

1 in 2.86 ms

6.1.4

Ce projet m'a permis d'acquérir les compétences du référentiel:

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

2. Précisez les moyens utilisés :

❖ Environnement de développement intégré (IDE)

J'ai utilisé Visual Studio Code pour la rédaction de mon code en général et il me permet la prise en charge native de divers langages d'où React JS qui est un Framework JS.

❖ Git ,GitHub et Github Desktop

Ces 3 outils m'ont permis de gérer les versions de mon code. Ils m'ont permis de sauvegarder chacune des fonctionnalités de mon projet les unes après les autres, dans le cas où une erreur me force à revenir à une version antérieure.

❖ Symfony

Pour la rédaction plus optimisée de mon code grâce aux principes de composant, et toutes les autres fonctionnalités liées à ce framework

❖ Bootstrap

Pour le design épuré de l'application

DOSSIER PROFESSIONNEL ^(DP)

❖ **MySql**

Pour la création et gestion de la base de données

❖ **Google font**

Pour l'ajout de police spécifique

❖ **Trello**

Pour la gestion de mon travail. Le meilleur moyen de bien développer est de s'organiser le mieux possible, soit de différencier les tâches et les mettre régulièrement à jour

❖ **Documentation**

Je me suis documenté sur Symfony, voici deux sources qui me paraissent pertinentes :

❖ <https://symfony.com/doc/current/index.html>

Qui est la documentation officielle de Symfony. Elle met à disposition une documentation complète sur les bonnes pratiques et l'utilisation de Symfony de son installation à son utilisation.

❖ <https://openai.com/blog/chatgpt>

Qui est un IA répondant à tous types de questions, concernant des sujets divers et variés. Cela m'a permis de répondre à mes questions et de corriger des erreurs que je ne comprenais pas forcément.

3. Avec qui avez-vous travaillé ?

4. Contexte

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

Chantier, atelier, service ▶ *Projet Blog*

Période d'exercice ▶ Du : *19/09/2022* au : *24/09/2022*

5. Informations complémentaires (facultatif)

Activité-type 2 Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Exemple n° 1 - Calculator 9000 (calculatrice JS)

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

Dans le cadre de ma formation à la Plateforme, pour l'obtention de mon titre RNCP Concepteur Développeur Application, j'ai réalisé une calculatrice en utilisant le frameworks de Javascript: **React.JS**.

Le but de ce projet était de recréer les fonctionnalités d'une calculatrice tout en pouvant enregistrer les résultats de nos calculs.

Cette WebApp devait comporter diverses fonctionnalités dictées par des consignes précises:

- ❖ Des fonctions basiques telles que : Additionner, Soustraire, Multiplier, Diviser.
- ❖ Prendre en charge les valeurs décimales.
- ❖ Fonctionnalité de formatage: suppression complète du calcul.
- ❖ Modification des calculs
- ❖ Fonctionnalité de limite : qui affiche un texte lorsque la valeur du résultat est supérieure à 9000.
- ❖ Fonctionnalité de Sauvegarde : permettant de sauvegarder les résultats et les calculs en base de données

Ce projet semblait simple mais comportait une multitude de détails rendant sa conception plus complexe. L'application a été développée sur le framework JS: **React.JS** qui m'a beaucoup servi notamment dans l'optimisation du code en utilisant des composants pour construire l'application.

Dans le sujet, il était demandé de sauvegarder à l'aide d'un bouton "SAVE" le calcul et le résultat affiché. Pour ce faire, j'ai utilisé PHP, SQL et MySQL.

Pour commencer, l'utilisateur arrive sur la calculatrice, design simple.

Calculator9000

0					
0					
0	1	2	3	4	+
					-
5	6	7	8	9	*
					/
=	C	Save			.
Calcul			Resultat		

- ❖ Comme sur une calculatrice normale, on peut insérer des valeurs et ajouter un ou plusieurs opérateurs pour effectuer son calcul.

Calculator9000

100+100					
200					
0	1	2	3	4	+
					-
5	6	7	8	9	*
					/
=		C		Save	
					.
Calcul			Resultat		

Si l'on fait un calcul, le résultat sera affiché juste en dessous, j'ai réalisé ça en créant un deuxième écran.

- ❖ Dans le cas de figure où l'on décide de réaliser une sauvegarde de son résultat, on appuie sur le bouton 'Save' à cet effet pour enregistrer son calcul en base de données. Et par la suite la base de données renvoie toutes les données sauvegardées dans un tableau prévu à cet effet:

Calculator9000

15+15					
30					
0	1	2	3	4	+
					-
5	6	7	8	9	*
					/
=		C		Save	
				.	
Calcul			Resultat		
15+15			30		

La calculatrice fonctionne à l'aide de différents composants ayant chacun un but bien précis:

- ❖ Composant BeautifulScreen:
Le Component représentant l'écran d'affichage de la calculatrice. Il doit afficher différentes choses telles que l'état du calcul en cours et le résultat.
- ❖ Composant AmazingNumberButton:
Un composant simple qui servira à représenter les touches numériques de la calculatrice.
- ❖ Composant GreatOperationButton:
Le cœur et l'âme du calculatrice ... Ce component servira à représenter les boutons d'opérations telles que +, -, *, /.
- ❖ Composant MagnificentEqualButton:
Ce component-ci devra être le plus beau de tous puisqu'il représente la touche "=" de la calculatrice qui est le meilleur bouton.
- ❖ Composant Calculator:
Le component principal, il sera composé des composants ci-dessus, entre autres.
- ❖ Composant ItSOverNineThousand.
- ❖ Ce component devra être invisible par défaut mais afficher "It's Over 9000 !!!" quand le résultat d'un calcul est supérieur à 9000.

Ce projet m'a permis d'acquérir les compétences du référentiel:

- ☒ **Développer une interface utilisateur de type desktop**
- ☒ **Développer des composants d'accès aux données**
- ☒ **Développer la partie front-end d'une interface utilisateur web**
- ☒ **Développer la partie back-end d'une interface utilisateur web**

2. Précisez les moyens utilisés :

- ❖ **Environnement de développement intégré (IDE)**
J'ai utilisé Visual Studio Code pour la rédaction de mon code en général et il me permet la prise en charge native de divers langages d'où React JS qui est un Framework JS.
- ❖ **Git ,GitHub et Github Desktop**
Ces 3 outils m'ont permis de gérer les versions de mon code. Ils m'ont permis de sauvegarder chacune des fonctionnalités de mon projet les unes après les autres, dans le cas où il y aurait une erreur.

DOSSIER PROFESSIONNEL ^(DP)

❖ React JS

Pour la rédaction plus optimisée de mon code, grâce au principes de composant, et toutes les autres fonctionnalités liées à ce framework

❖ PHP

Pour la gestion de la partie backend du projet, la récupération des données via la base de données

❖ MySql

Pour la création et la gestion de la base de données

❖ Google font

Pour l'ajout de police spécifique

❖ Trello

Pour la gestion de mon travail. Le meilleur moyen de bien développer est de s'organiser le mieux possible ou de différencier les tâches et les mettre régulièrement à jour

❖ Documentation

Je me suis documenté sur Reactjs, voici deux sources qui me paraissent pertinentes :

- ❖ <https://legacy.reactjs.org/docs/getting-started.html>
Qui est la documentation officielle de React Js. Elle met à disposition une documentation complète sur les bonnes pratiques et l'utilisation de Reactjs, de son installation à son utilisation.
- ❖ <https://blog.eleven-labs.com/fr/optimiser-son-application-react/>
- ❖ Qui s'adresse aux développeurs dans l'optimisation de leur code. Ce site met à disposition des conseils du Framework pour optimiser au maximum son code.

3. Avec qui avez-vous travaillé ?

J'ai réalisé ce projet seul

4. Contexte

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

Chantier, atelier, service ▶ *Projet en centre de formation*

Période d'exercice ▶ Du : *12/09/2022* au : *17/09/2022*

Activité-type 2 Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Exemple n° 2 - My Sokoban

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

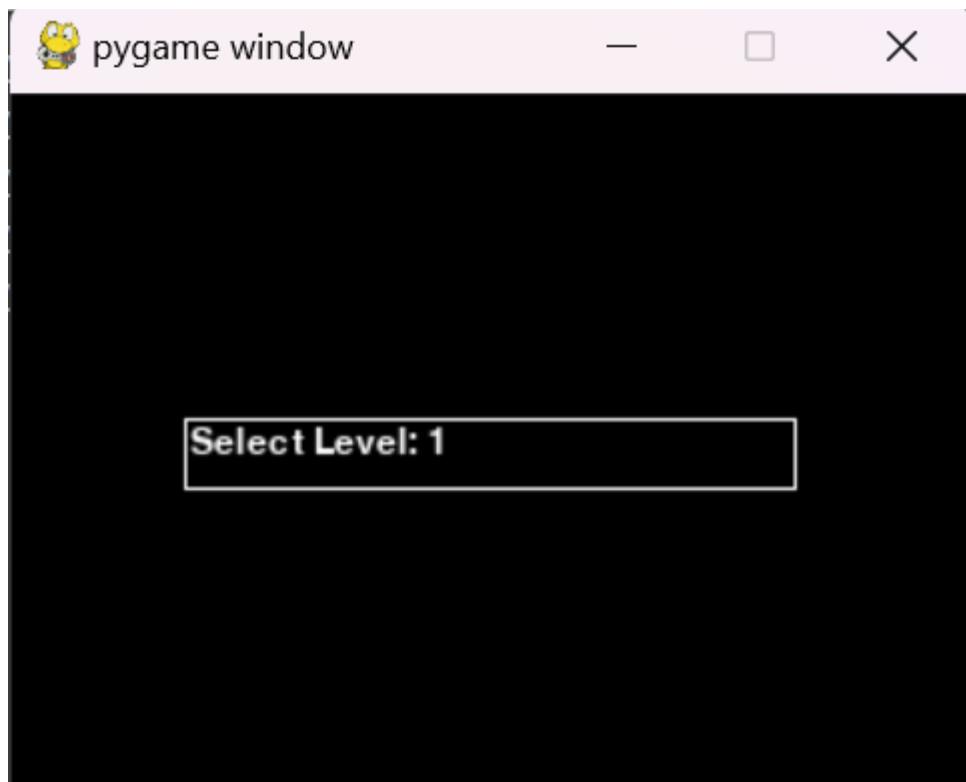
Dans le cadre de ma formation de développeur et concepteur d'application, et ma présentation de fin d'année, j'ai dû réaliser une application de type desktop. Le projet My Sokoban est un jeu où le contexte est de déplacer des caisses jusqu'à l'endroit désigné sur la map.

Pour réaliser ce projet j'ai utilisé les technos suivantes:

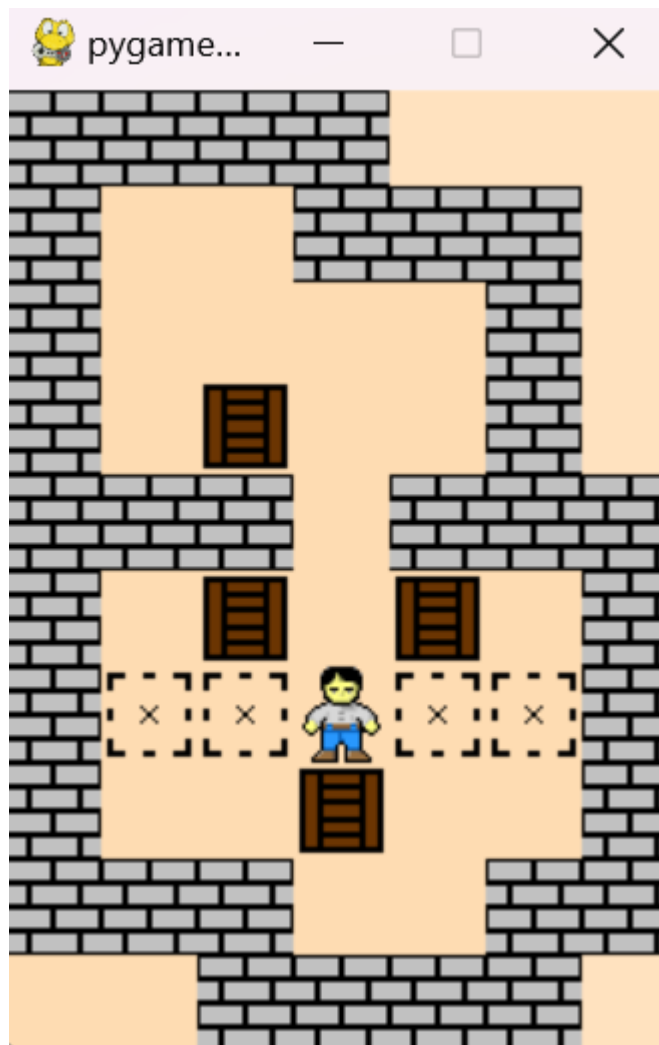
- ❖ **Python** pour programmer l'appli
- ❖ **Pygames** pour l'interface graphique

Pour ce faire, il fallait créer une architecture avec un modèle de donnée (une classe). Dans cette classe j'ai créé des fonctions qui gèrent les événement de clavier, le déplacement des caisses et du personnage, la création de la map en fonction du niveau choisi.

Lorsque l'on démarre l'appli on doit choisir un niveau entre 1 et 50:

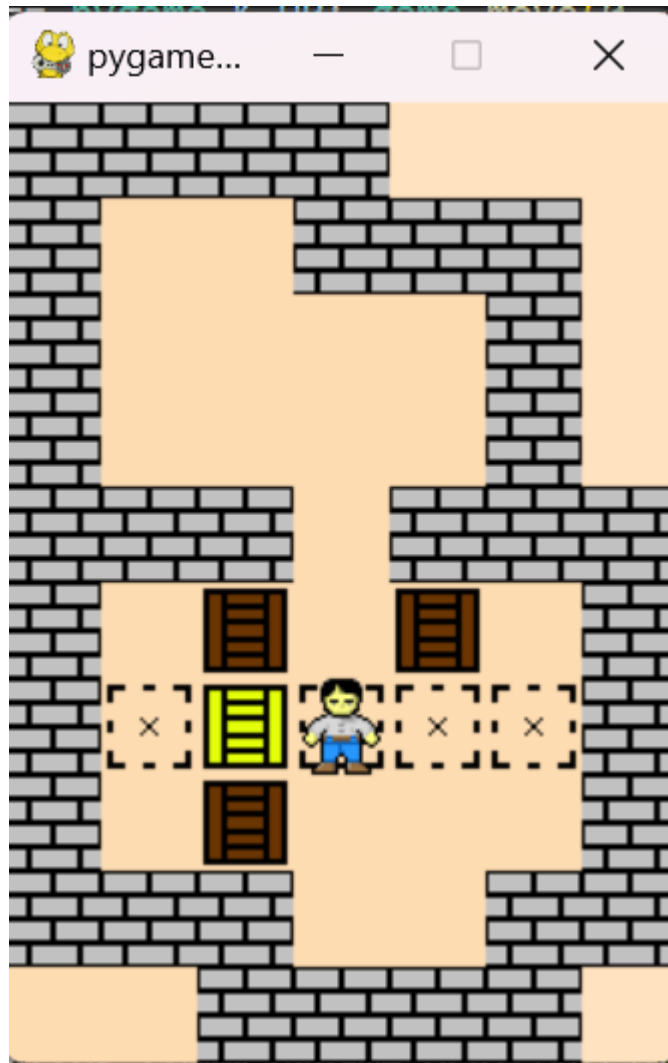


Ensuite, une fois que l'on a choisi le niveau (en l'occurrence le niveau 1), on peut voir apparaître le niveau choisi:



On peut voir la map avec les caisses, le personnage qui va déplacer les caisses et leurs emplacements.

Une fois que l'on a placé une caisse à son objectif, elle change de couleur pour devenir dorée:



Il ne reste plus qu'à placer les autres caisses jusqu'à réussir le niveau.

DOSSIER PROFESSIONNEL ^(DP)

2. Précisez les moyens utilisés :

Pour réaliser ce projet j'ai utilisé:

- ❖ Python pour la programmation
- ❖ La librairie Pygames pour l'interface graphique
- ❖ Des images pour les caisses, les murs et l'ouvrier.

3. Avec qui avez-vous travaillé ?

J'ai réalisé ce projet seul.

4. Contexte

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

Chantier, atelier, service ▶ My Sokoban

Période d'exercice ▶ Du : 17/04/2023 au : 22/04/2023

5. Informations complémentaires *(facultatif)*

Activité-type 3 Concevoir et développer une application multicouches répartie en intégrant les recommandations de sécurité

Exemple n° 1 - Akiba Town

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

Dans le cadre de ma formation de développeur et concepteur d'application, et ma présentation de fin d'année, j'ai dû réaliser un projet regroupant toutes les compétences que j'ai pu acquérir durant mon cursus. Le projet est un projet Forum sous forme d'application mobile, ce projet semblait très complet aussi bien dans la conception que dans le développement.

Pour ce projet nous avons utilisé :

- ❖ **React Native** pour la partie Frontend
- ❖ **Node.js** comme environnement de développement coté serveur (Backend)
- ❖ **Express**, un framework de Node.js

Un framework de développement mobile est l'un des éléments les plus essentiels pour créer des applications mobiles riches en fonctionnalités et hautement performantes. Le choix du framework a un impact direct sur la qualité du développement et l'expérience des utilisateurs. **React Native** est l'un des principaux frameworks de développement mobile utilisés par les développeurs du monde entier aujourd'hui. Il offre des fonctionnalités robustes et un nombre important de fonctions puissantes pour les applications iOS et Android.

- ❖ React est un framework de développement d'applications basé sur JavaScript.
- ❖ React Native est une plateforme permettant aux utilisateurs de développer des applications mobiles natives et multiplateformes.

La bibliothèque JavaScript React.js au cœur de React Native est utilisée pour créer une couche d'interface utilisateur robuste. Cette bibliothèque est conforme aux principes fondamentaux et à la syntaxe de React, ce qui la rend facile à apprendre. Cependant, React.js et React Native sont quelque peu différents, car le code du navigateur React est rendu par le DOM virtuel, mais React Native utilise des API natives pour le rendu des composants mobiles. React est utilisé pour le codage avec HTML mais la syntaxe native de React doit être apprise. Les composants des applications peuvent être animés en utilisant l'API animée fournie avec React Native.

Concernant la partie BackEnd de notre projet, nous avons utilisé MySQL et Sequelize pour gérer notre base de données :

- ❖ **MySQL** est un **système de gestion de bases de données relationnelles open-source**, c'est-à-dire un logiciel qui permet de stocker, d'organiser et de récupérer des données structurées. Il est

utilisé pour gérer des bases de données dans de nombreux types d'applications, telles que les sites web, les systèmes de gestion de contenu, les applications d'entreprise et bien d'autres encore. MySQL est l'un des systèmes de gestion de bases de données les plus populaires au monde, car il est rapide, fiable, facile à utiliser et prend en charge de nombreuses fonctionnalités avancées. Voilà pourquoi nous avons utilisé ce système.

- ❖ **Sequelize** est une **bibliothèque JavaScript open source** qui permet d'interagir avec une base de données relationnelle à l'aide d'objets JavaScript plutôt que d'écrire directement du code SQL. Elle est utilisée pour la gestion des modèles de données et des requêtes associées dans des applications Node.js basées sur des bases de données relationnelles telles que MySQL, PostgreSQL, SQLite, MSSQL, Oracle, etc. Sequelize fournit également des fonctionnalités telles que la migration de la base de données, la validation des données et les associations entre les tables. En résumé, Sequelize simplifie l'accès aux bases de données relationnelles en fournissant une interface conviviale et basée sur des objets pour interagir avec la base de données.

Pour ce projet que nous devons réaliser entièrement, de la conception au développement nous avons dû dans un premier temps :

- ❖ Définir et comprendre toutes les fonctionnalités de l'application :
 - ❖ Créer un compte, Se connecter
 - ❖ Modifier son Profil
 - ❖ Consulter la totalité des sujet lancés
 - ❖ Créer un sujet
 - ❖ Réagir à un sujet (Aimer, mettre en favori, Commenter)
 - ❖ Faire une recherche par rapport à l'aide de filtres

à cela nous avons décidé d'ajouter quelques pages permettant de compléter les fonctionnalités et la navigation pour l'utilisation.

- ❖ Consulter un sujet en particulier
 - ❖ Consulter un profil d'utilisateur
 - ❖ Consulter une liste regroupant tous les sujets
 - ❖ Consulter un thème en particulier (tous les sujets d'un thème)
 - ❖ Consulter un annuaire (Modérateur, Administrateur)
 - ❖ Consulter un résumé des dernières activités de l'appli (nouvel utilisateur, nouveau commentaire, nouveau sujet etc...)
- ❖ Établir un cahier des charges permettant de bien résumer le projet et se mettre d'accord sur tous les besoins de l'application
- ❖ Réaliser une maquette permettant d'avoir un visuel de l'application
- ❖ La réalisation d'un MCD et MLD pour conceptualiser la base de données et faciliter sa création
- ❖ Répartir les tâches à l'aide d'un tableau Trello pour nous permettre de mieux nous organiser dans le développement de l'application
- ❖ Mettre en place l'environnement de travail, c'est à dire :
 - ❖ S'assurer des versions des outils sur lesquelles on travaille pour éviter les problèmes de versions
 - ❖ Faire toutes les installations de dépendances de bibliothèques et librairies) nécessaires

au bon fonctionnement de l'application

- ❖ Création des API pour stocker les données d' utilisateurs et de sujets, récupération d'API externes pour générer les sélecteurs dynamiques

Après avoir réalisé toutes ces tâches, nous avons enfin pu commencer le développement.

Lors de la rédaction du cahier des charges, nous avons décidé de créer différents types d'utilisateurs :

- ❖ **Utilisateur** : un utilisateur lambda pouvant créer un article, like, commenter et mettre en favori d'autres articles.
- ❖ **Modérateur** : un super utilisateur pouvant modifier et supprimer des articles pour telle ou telle raison.
- ❖ **Administrateur** : un super modérateur ayant les mêmes avantages qu'un modérateur mais aussi ayant accès à un historique des utilisateurs, un historique des articles mais aussi supprimer ou modifier des utilisateurs.

Nous avons aussi décidé de donner certaines permissions aux utilisateurs qui ne sont pas connectés ou n'ayant pas encore de comptes :

- ❖ Pour les utilisateurs non connectés :
 - Accéder à la **page d'accueil** qui réceptionnera les utilisateurs non connectés pour leur laisser le choix de se connecter, s'inscrire ou directement accéder au forum



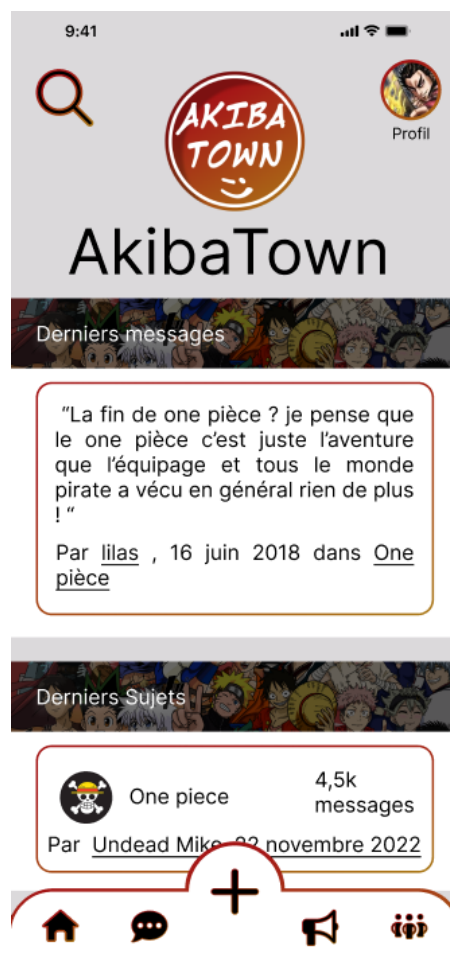
- Accéder à la page de connexion pour se connecter



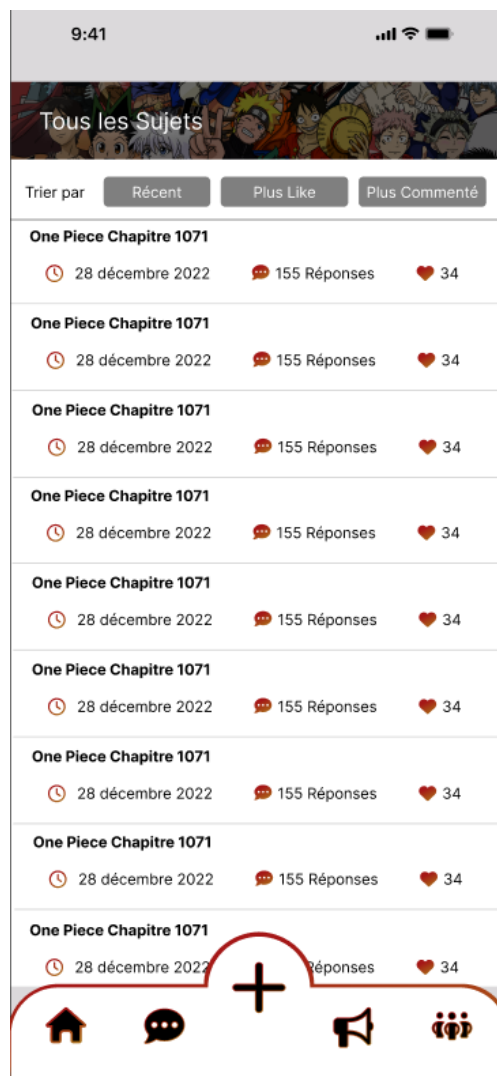
- Accéder à la page d'inscription pour s'inscrire



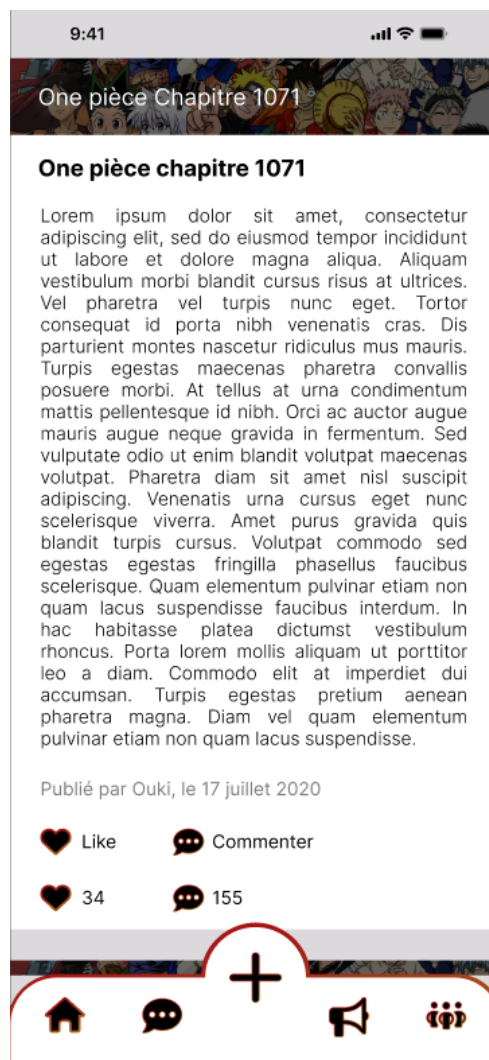
- ❖ Accéder à la landing page permettant de réceptionner tous les utilisateurs connectés ou non. Cette page résume toutes les dernières activités du site telles que :
 - ❖ Les derniers messages (réponse d'article)
 - ❖ Les Derniers articles publiés
 - ❖ Les statistiques de l'application (Nombre de membres, nombres d'articles, nombres de réponses)



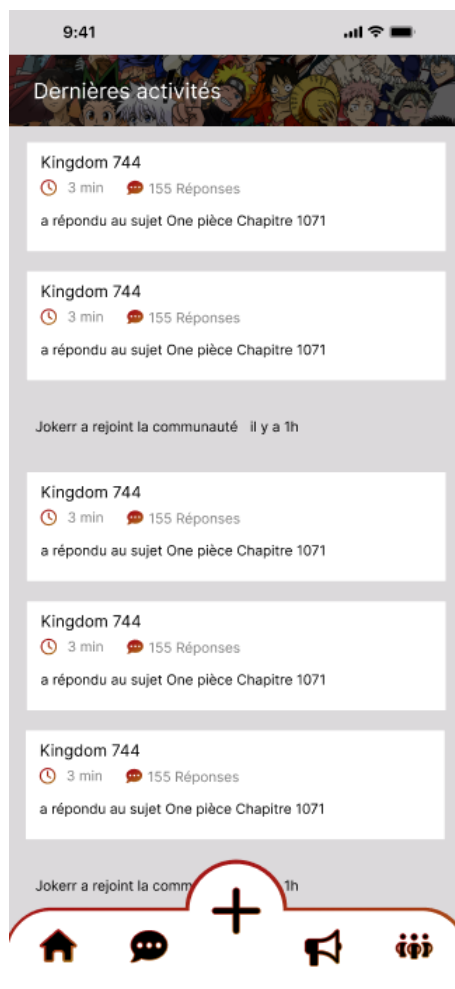
- ❖ Accéder à la page de forum regroupant l'intégralité des forums publiés et y accéder à un article en particulier



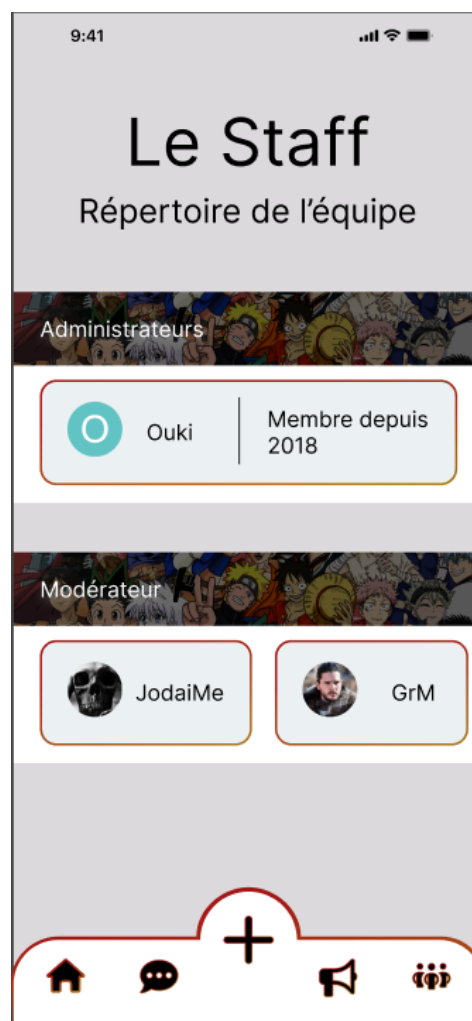
- ❖ Accéder à un Sujet en particuliers **sans pour autant y réagir**



- ❖ Accéder à la page d'actualités permettant de voir toutes les dernières notifications de l'application

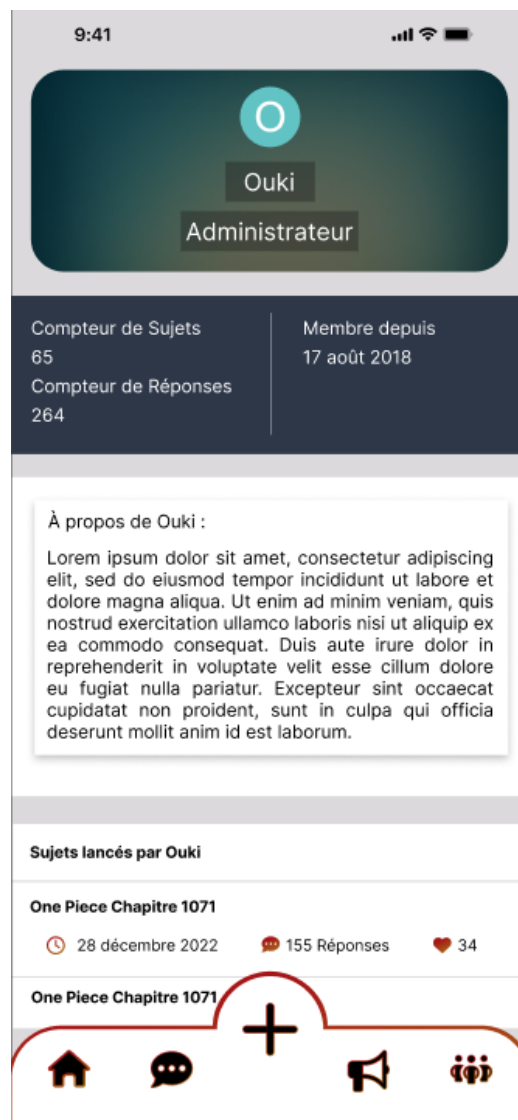


- ❖ Accéder à l'annuaire permettant de voir le membre du staff de l'application c'est à dire l'équipe de modération et celle d'administration

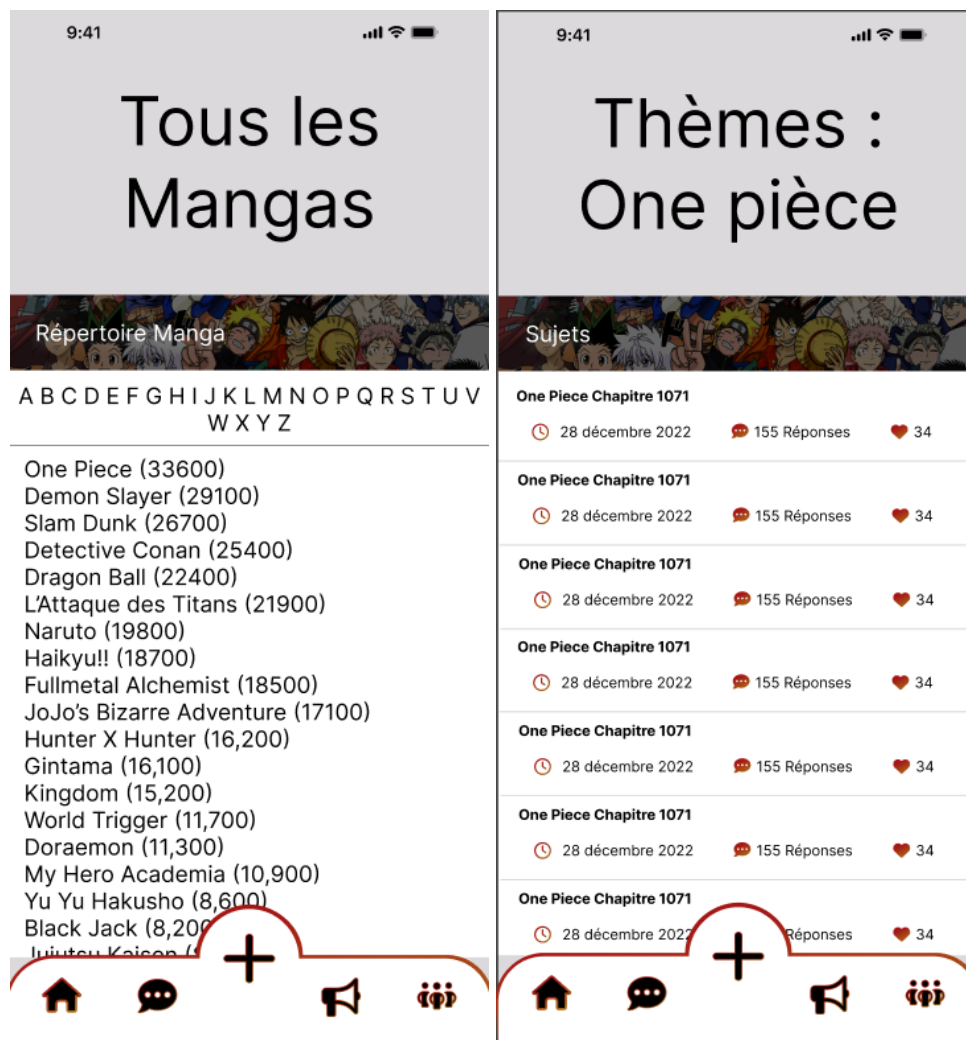


- ❖ Accéder aux pages Profil d'utilisateur permettant de voir les informations sur un utilisateur

DOSSIER PROFESSIONNEL (DP)



- ❖ Accéder aux pages de recherche et de classement par thèmes, pour respectivement lister tous les thèmes et faciliter la recherche concernant des thèmes bien précis.

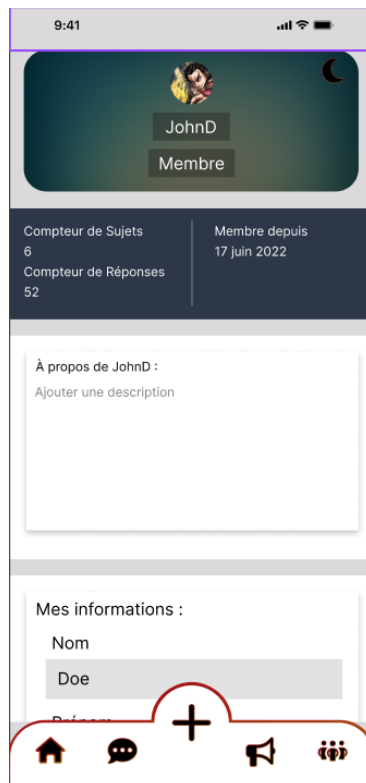


- ❖ Pour les utilisateur connectés :
 - ❖ Toutes les pages précédentes sont accessibles par les utilisateurs connectés sauf les pages Inscription, connexion et landing page
 - ❖ Les réactions sont disponibles pour ces derniers, ils peuvent maintenant Liker, commenter et mettre un poste en favori.
 - ❖ Accéder à la page de création de post pour créer un post et interagir avec la communauté

DOSSIER PROFESSIONNEL (DP)

The screenshot shows a mobile application interface for creating a new subject. At the top, the status bar displays the time 9:41 and signal indicators. Below the status bar is a header with the text "Nouveau sujet" and a background image of anime characters. The main form consists of several fields: "Titre du sujet" with a placeholder "Titre du sujet", "Choix du manga" with a placeholder "Choisir un manga" and a dropdown arrow, and "Description :" with a placeholder "Description du sujet...". At the bottom right of the form is an "Envoyer" button. Below the form is a navigation bar with five icons: a house, a speech bubble, a plus sign, a megaphone, and a group of people.

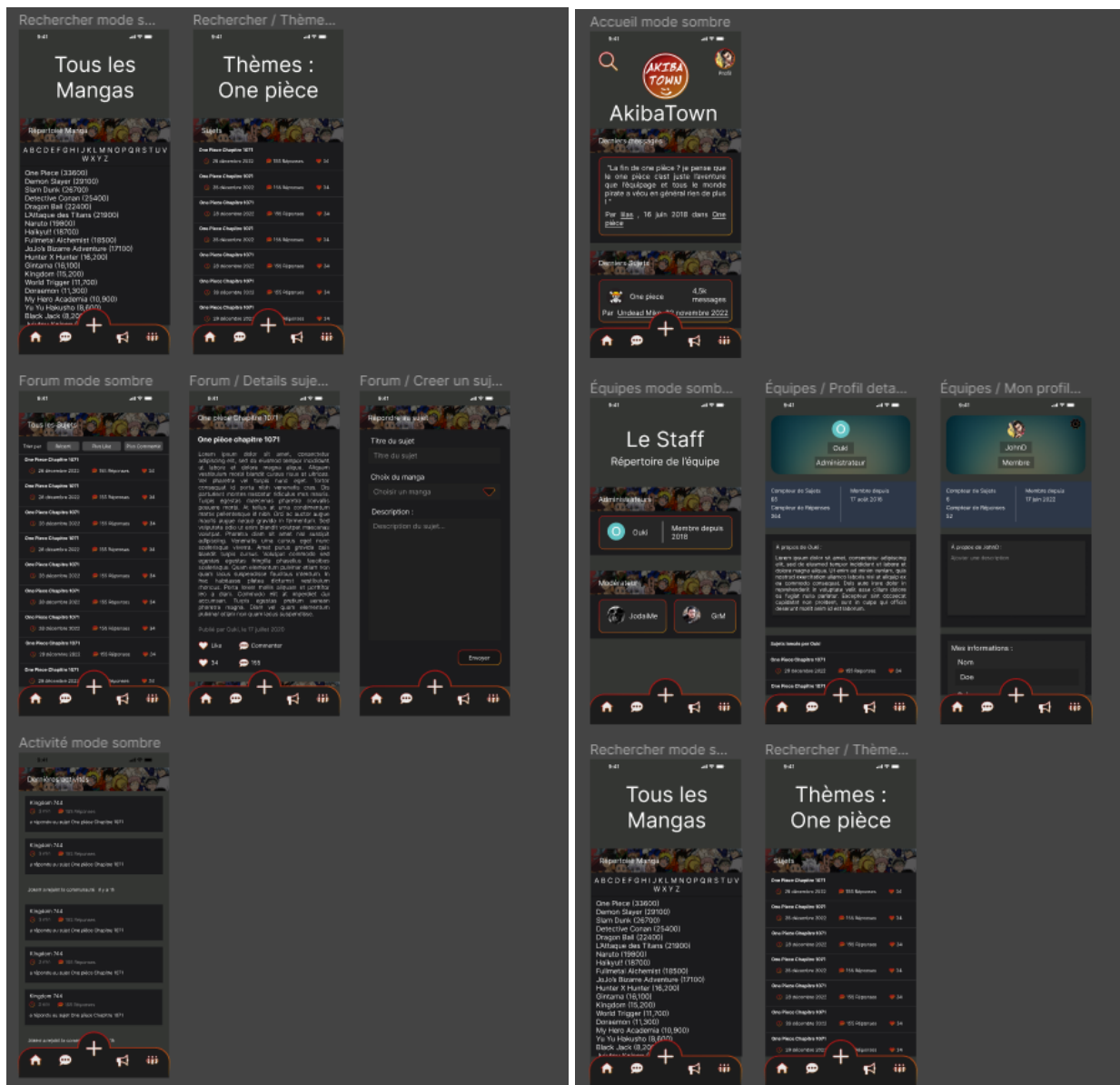
- ❖ Accéder à la page Mon profil pour accéder à leurs informations personnelles, les compléter et les modifier.



Les Modérateurs, quant à eux, ont accès à une page qui listera tous les posts, classés par thèmes pour pouvoir ensuite modifier ou supprimer un post. Les administrateurs y auront aussi accès en plus de la page listant tous les membres de l'application pour consulter, modifier, ou supprimer un utilisateur.

Nous avons aussi décidé de travailler sur le design de l'application dans le sens où nous avons intégré le changement de thèmes. En effet nous avons intégré un thème sombre pour les utilisateurs fuyant la lumière blanche :

DOSSIER PROFESSIONNEL (DP)



Ce projet nous aura permis de mieux gérer la partie conceptuelle du projet, c'est-à-dire toute la partie de brainstorming. Elle nous aura permis de beaucoup mieux gérer l'organisation du projet et par la suite la répartition des tâches.

Ce projet nous aura permis d'acquérir les compétences suivantes :

- ☒ Collaborer à la gestion d'un projet informatique et à l'organisation de l'environnement de développement
- ☒ Concevoir une application
- ☒ Développer une application mobile
- ☒ Préparer et exécuter les plans de tests d'une application
- ☒ Préparer et exécuter le déploiement d'une application

2. Précisez les moyens utilisés :

❖ Environnement de développement intégré (IDE)

J'ai utilisé Visual Studio Code pour la rédaction de mon code en général et il me permet la prise en charge native de divers langages d'où React JS qui est un Framework JS.

❖ Git ,GitHub et Github Desktop

Ces 3 outils m'ont permis de gérer les versions de mon code. Ils m'ont permis de sauvegarder chacune des fonctionnalités de mon projet les unes après les autres, dans le cas où une erreur me force à revenir à une version antérieure.

❖ React Native

Pour la rédaction plus optimisée de mon code grâce aux principes de composant, et toutes les autres fonctionnalités liées à ce framework

❖ Bootstrap

Pour le design de la partie modération et administration

❖ SQL

SQL est utilisé dans une grande variété d'applications et de systèmes de gestion de bases de données (SGBD), tels que MySQL, PostgreSQL, Oracle, Microsoft SQL Server, SQLite, etc. Il est essentiel pour stocker, récupérer et manipuler efficacement les données dans les bases de données relationnelles.

❖ MySql

MySQL est largement utilisé dans de nombreuses applications et systèmes, allant des sites web et des applications d'entreprise aux systèmes de gestion de contenu (CMS) tels que WordPress. En tant que SGBDR populaire et open source, MySQL offre une performance élevée, une stabilité et une flexibilité pour la gestion des données.

❖ Node.js

Node.js est un environnement d'exécution côté serveur basé sur le moteur JavaScript V8 de Google Chrome. Il permet d'exécuter du code JavaScript côté serveur, contrairement à son utilisation traditionnelle côté client dans les navigateurs web.

❖ Express

Express est un framework web minimaliste pour Node.js qui facilite le développement d'applications web côté serveur. Il est conçu pour être simple, flexible et performant.

❖ Google font

Pour l'ajout de police spécifique

DOSSIER PROFESSIONNEL ^(DP)

❖ Trello

Pour la gestion de mon travail. Le meilleur moyen de bien développer est de s'organiser le mieux possible soit de différencier les tâches et les mettre régulièrement à jour

❖ Figma

Qui est un outil de design graphique collaboratif basé sur le Cloud, me permettant de réaliser les visuel de la maquette de l'application

❖ Documentation

❖ <https://reactnative.dev/docs/getting-started>

Qui est la documentation officielle de React Native. Elle met à disposition une documentation complète sur les bonnes pratiques et l'utilisation de React Native de son installation à son utilisation.

❖ <https://openai.com/blog/chatgpt>

Qui est un IA répondant à tous types de questions, concernant des sujets divers et variés. Cela m'a permis de répondre à mes questions et de corriger des erreurs que je ne comprenais pas forcément.

❖ <https://sequelize.org/docs/v6/getting-started/>

❖ Qui est la documentation officielle de Sequelize. Elle met à disposition une documentation complète sur les bonnes pratiques et l'utilisation de Sequelize de son installation à son utilisation.

3. Avec qui avez-vous travaillé ?

j'ai travaillé avec :

- Hugo Chabert
- François Niang
- Thibault Kine

4. Contexte

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

Chantier, atelier, service ➤ *Akiba Town*

Période d'exercice ➤ Du : *03/01/2023* au : *11/03/2023*

5. Informations complémentaires (facultatif)

Activité-type 3

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

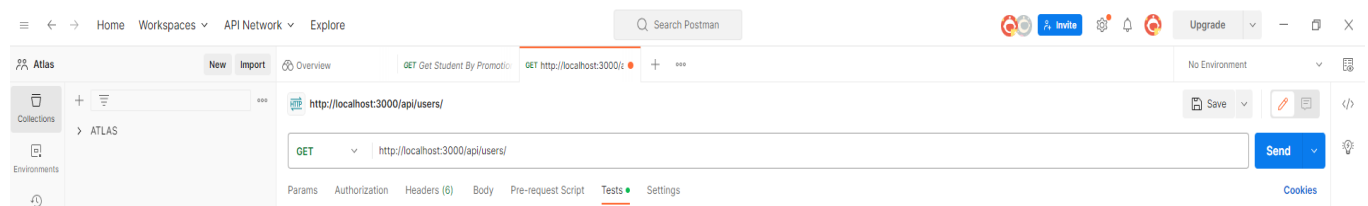
Exemple n° 2 - Tests de l'application Akiba Town (fonctionnels et unitaires)

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

Test fonctionnel d'api avec Postman

Nous souhaitons tester si l'API renvoie les données correctement . Nous avons testé nos Url d'API.

On s'est servi de postman pour réaliser cette tâche



Exemple d'url: **GET** | `http://localhost:3000/api/users/`

Pour récupérer l'ensemble des utilisateurs inscrits sur notre app mobile

Ensuite on clique sur l'onglet "Tests" juste en dessous de la zone de requête.

Dans la section des scripts de test, nous avons écrit du code JavaScript pour effectuer des assertions sur la réponse de l'API.

```
// Vérification du code de statut de la réponse
✓ pm.test("Statut de la réponse", function () {
  |   pm.response.to.have.status(200);
  | };

// Vérification du format de la réponse en JSON
✓ pm.test("Format de la réponse", function () {
  |   pm.response.to.be.json;
  | });

// Vérification du contenu de la réponse
✓ pm.test("Contenu de la réponse", function () {
  |   var jsonData = pm.response.json();

✓   pm.test("Vérification de la réponse", function () {
  |     pm.expect(jsonData).to.not.be.null;
  |     pm.expect(jsonData).to.not.be.undefined;
  |     // Autres assertions...
  |   });

✓   pm.test("Vérification des propriétés", function () {
  |     pm.expect(jsonData.data).to.have.property(0).that.is.not.null.and.not.undefined;
  |     // Autres assertions...
  |   });

✓   var expectedValue = "some value";
  |   pm.test("Vérification de la valeur", function () {
  |     pm.expect(jsonData.data[0].firstname).to.equal('Cyril');
  |     // Autres assertions...
  |   });

  |   // Vérification du nombre d'utilisateurs renvoyés
  |   pm.expect(jsonData.data).to.have.length(12);

  |   // Vérification des propriétés des utilisateurs
  |   pm.expect(jsonData.data[0]).to.have.property('id');
  |   pm.expect(jsonData.data[0]).to.have.property('firstname');
  |   pm.expect(jsonData.data[0]).to.have.property('lastname');
  | });
```

- nous vérifions que la réponse de l'API a un code de statut 200
- qu'elle est au format JSON
- on vérifie le contenu de la réponse

DOSSIER PROFESSIONNEL (DP)

- Vérification de la réponse pour voir si elle est nulle ou pas .
- On vérifie si les propriété de la réponse ne sont pas vide ou undefined
- On vérifie si réponse correspond au résultat attendu
- qu'elle contient les détails corrects pour l'utilisateur avec l'ID 1

Body Cookies Headers (8) Test Results (5/6)

All Passed Skipped Failed

PASS Statut de la réponse

PASS Format de la réponse

PASS Vérification de la réponse

PASS Vérification des propriétés

FAIL Vérification de la valeur | AssertionError: expected 'Ahmed' to equal 'Cyril'

PASS Contenu de la réponse

```
var expectedValue = "some value";
pm.test("Vérification de la valeur", function () {
  pm.expect(jsonData.data[0].firstname).to.equal('Ahmed');
  // Autres assertions...
});
```

Body Cookies Headers (8) Test Results (6/6)

All Passed Skipped Failed

PASS Statut de la réponse

PASS Format de la réponse

PASS Vérification de la réponse

PASS Vérification des propriétés

PASS Vérification de la valeur

PASS Contenu de la réponse

Test Unitaire avec jest

Nous souhaitons tester si chaque unité fonctionne correctement de manière isolée. Nous effectuons des tests unitaires. Ils garantissent la stabilité et la fiabilité du code en détectant rapidement les erreurs et en facilitant la maintenance du système. Les tests unitaires contribuent ainsi à améliorer la qualité du logiciel et à assurer un développement itératif plus efficace. Pour ce faire nous utilisons le framework Jest.

Nous voulions tester nos endpoint, voici l'exemple suivant effectué une route:

<http://localhost:3000/users/delete/:id>.

Nous testons le statut de la requête:

```
API > tests > delete.test.js > ...
1  // on importe les modules nécessaires pour le test
2  import axios from "axios";
3  import request from "supertest";
4  import server from "../App";
5
6  const { Users } = require('../src/models/UserContext.js');
7  const sequelize = require('../src/config/database.js');
8
9
10 const UserModel = sequelize.models.users
11
12 // Augmenter le délai d'attente pour chaque test
13 jest.setTimeout(10000);
14
15 // Test unitaire pour la route DELETE /users/delete/:id
16 describe('/users/delete/:id', () => {
17   it('should delete a user and return "User deleted!"', async () => {
18     // on fait une requête DELETE à l'API pour supprimer l'utilisateur de test
19     const response = await request(server).delete(`/users/delete/12`);
20
21     // on vérifie le statut de la réponse (200 OK dans ce cas)
22     expect(response.status).toBe(200);
23
24     // on vérifie le contenu de la réponse
25     expect(response.text).toBe('User deleted!');
26   });
27 }
```

Voici le résultat obtenu, la validation des tests:

```
PASS tests/delete.test.js
/users/delete/:id
  ✓ should delete a user and return "User deleted!" (138 ms)

Test Suites: 1 passed, 1 total
Tests:       1 passed, 1 total
Snapshots:   0 total
Time:        3.472 s, estimated 4 s
Ran all test suites.
```

2. Précisez les moyens utilisés :

❖ Node.js

Node.js est un environnement d'exécution côté serveur basé sur le moteur JavaScript V8 de Google Chrome. Il permet d'exécuter du code JavaScript côté serveur, contrairement à son utilisation traditionnelle côté client dans les navigateurs web.

❖ Express

Express est un framework web minimaliste pour Node.js qui facilite le développement d'applications web côté serveur. Il est conçu pour être simple, flexible et performant.

❖ Jest

Jest est un framework de test JavaScript construit sur Jasmine et maintenu par Meta. Il a été conçu et construit par Christoph Nakazawa en mettant l'accent sur la simplicité et la prise en charge des grandes applications Web.

3. Avec qui avez-vous travaillé ?

j'ai travaillé avec :

- Hugo Chabert
- François Niang
- Thibault Kine

DOSSIER PROFESSIONNEL ^(DP)

4. Contexte

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

Chantier, atelier, service ▶ *Akiba Town*

Période d'exercice ▶ Du : *03/01/2023* au : *11/03/2023*

5. Informations complémentaires *(facultatif)*

DOSSIER PROFESSIONNEL ^(DP)

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) SYLLA Ibrahim ,
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 Marseille le 26/06/2023

pour faire valoir ce que de droit.

Signature : SYLLA Ibrahim

Documents illustrant la pratique professionnelle

(facultatif)

Intitulé
Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

ANNEXES

(Si le RC le prévoit)