



DOSSIER PROFESSIONNEL (DP)

MINISTÈRE CHARGÉ
DE L'EMPLOI

Nom de naissance

- Gadoumi Tarik
-
- Tarik
- 10 rue des petites maries, 13001 Marseille , boite aux lettres n°37

Nom d'usage

Prénom

Adresse

Titre professionnel visé

Concepteur(trice) Développeur(se) Informatique

MODALITÉ D'ACCÈS :

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

DOSSIER PROFESSIONNEL (DP)

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 des composants d'interface utilisateur en intégrant les recommandations de sécurité	p. 5
▸ Papotage Paradis Maquetter une Application	p. p.
▸ Papotage Paradis	p. p.
▸ Jeu + CLI	p. p.
Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité	p.
▸ Papotage Paradis	p. p.
▸ Blog e-learning	p. p.
▸	p. p.
Concevoir et développer la persistance des données en intégrant les recommandations de sécurité	p.
▸ Papotage Paradis	p. p.
▸	p. p.
▸ I	p. p.
Titres, diplômes, CQP, attestations de formation (facultatif)	p.
Déclaration sur l'honneur	p.
Documents illustrant la pratique professionnelle (facultatif)	p.

DOSSIER PROFESSIONNEL^(DP)

| Annexes (*Si le RC le prévoit*)

p.

DOSSIER PROFESSIONNEL^(DP)

EXEMPLES DE PRATIQUE

PROFESSIONNELLE

DOSSIER PROFESSIONNEL (DP)

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

Exemple n°1 - chat app et blog e-learning - Maquetter une application

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

Mobile

Pendant mes études à La Plateforme, j'ai entrepris le développement d'une application mobile de chat en utilisant React Native. Mon objectif principal était de créer une plateforme intuitive et centrée sur l'utilisateur, permettant aux individus de se connecter et d'interagir facilement. J'ai imaginé l'application comme une plateforme universelle, sans cibler un public spécifique, facilitant des conversations détendues et simples.

L'application mobile a été développée en utilisant le robuste et polyvalent cadre de React Native. Pour la partie back-end, j'ai utilisé Node.js et diverses bibliothèques pour alimenter les fonctionnalités de l'application.

Les dépendances front-end de l'application, détaillées dans le fichier package.json, révèlent la complexité du projet. Parmi les dépendances notables, "@apollo/client" pour gérer les requêtes réseau vers un serveur GraphQL, les bibliothèques "react-navigation" pour implémenter la navigation dans l'application, et "expo" pour une expérience de développement et de test fluide.

Sur le backend, l'application utilise Apollo Server pour créer un serveur GraphQL, JWT pour l'authentification, et Mongoose pour gérer les opérations de la base de données avec MongoDB. La partie serveur du projet a été développée dans un environnement Node.js.

En conclusion, le développement de cette application mobile de chat a été une expérience éclairante qui a aiguisé mes compétences en React Native et Node.js, et m'a donné des aperçus précieux sur le développement d'applications mobiles. C'est un projet dont je suis particulièrement fier et que je pense démontre mes capacités en tant que développeur.

Web

Après l'application mobile de chat, je me suis lancé dans la création d'un blog d'apprentissage en ligne en utilisant React pour le front-end et deux APIs REST pour le backend, une construite avec Lumen et l'autre avec Node.js Express. L'objectif était de proposer une plateforme d'apprentissage en ligne interactive et efficace, en mettant l'accent sur l'expérience utilisateur et le suivi fluide du contenu.

Le blog d'apprentissage en ligne permet aux utilisateurs authentifiés d'ajouter du contenu à une liste de lecture et de sauvegarder leurs notes, commentaires, critiques, et listes dans la base de données. Ce processus d'apprentissage interactif est conçu pour améliorer l'engagement des utilisateurs et enrichir

DOSSIER PROFESSIONNEL (DP)

l'expérience d'apprentissage.

L'application web exploite le robuste et polyvalent cadre de React pour le front-end, avec des dépendances notables comme "axios" pour la gestion des requêtes réseau, "react-router-dom" pour la navigation de l'application, et "react-query" pour la gestion des requêtes asynchrones. Pour le backend, Lumen et Node.js Express ont été utilisés pour alimenter les fonctionnalités, se concentrant sur l'authentification et les opérations de base de données.

En conclusion, le développement de ce blog d'apprentissage en ligne a affiné mes compétences en React, Lumen et Node.js Express, et a apporté des aperçus précieux sur le développement d'applications web. C'est un autre projet dont je suis particulièrement fier, et je crois qu'il souligne ma capacité d'adaptation et mes capacités en tant que développeur.

2. Précisez les moyens utilisés :

Mobile

Pour la conception de l'application mobile de chat, j'ai utilisé plusieurs outils et techniques pour assurer un processus de développement efficace et bien organisé.

Dans la phase de conception de l'architecture de l'application, j'ai utilisé des Modèles Conceptuels de Données (MCD), des Modèles Logiques de Données (MLD), et des diagrammes de flux. Ces outils m'ont aidé à visualiser la structure de l'application et à comprendre comment les données circulent et interagissent au sein de celle-ci.

Pour le maquettage et la conception de l'interface utilisateur, j'ai utilisé Figma. C'est un outil excellent qui a permis de créer des prototypes interactifs de l'interface utilisateur, me permettant d'avoir une visualisation claire du design avant le début du développement.

Pour générer des images, j'ai utilisé MidJourney. C'est un outil puissant pour créer des images de haute qualité qui ont été utilisées dans l'application.

Pour gérer le flux de travail et suivre les tâches pendant le processus de développement, j'ai utilisé Trello. C'est un outil de gestion de projet qui m'a permis de créer des tickets pour chaque tâche et de suivre leur progression.

Enfin, pour le contrôle des versions et la collaboration avec d'autres développeurs, j'ai utilisé Git. C'est un outil essentiel qui a facilité le suivi des modifications et a permis une collaboration efficace.

Web

Pour le blog d'apprentissage en ligne, le processus de conception était similaire. J'ai utilisé MCD, MLD et des diagrammes de flux pour la conception de l'architecture. Figma a été utilisé pour le maquettage de

DOSSIER PROFESSIONNEL (DP)

l'interface utilisateur, tandis que Midjourney a été utilisé pour la génération d'images.

J'ai également utilisé Trello pour le suivi des tâches et Git pour le contrôle des versions, ce qui a permis une gestion efficace du projet et une collaboration fluide.

3. Avec qui avez-vous travaillé ?

seul

4. Contexte

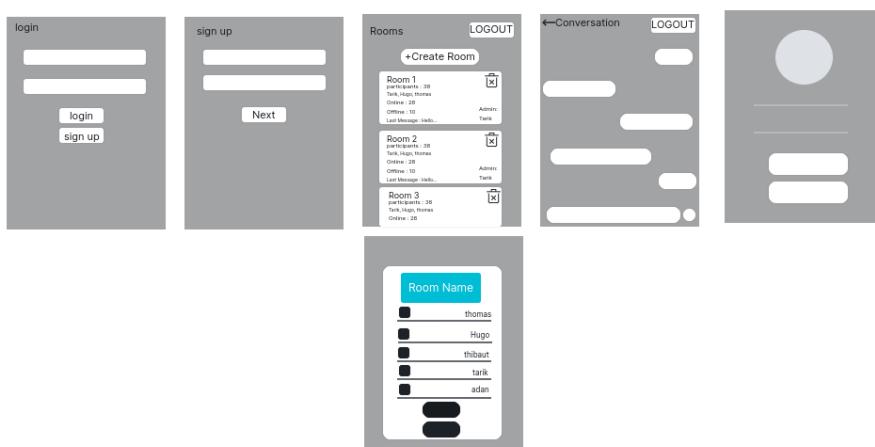
Nom de l'entreprise, organisme ou association ➤ **La plateforme**

Chantier, atelier, service ➤ **Papotage Paradis, E-learning Blog**

Période d'exercice ➤ Du : **01/03/2023** au : **01/06/2023**

5. Informations complémentaires (*facultatif*)

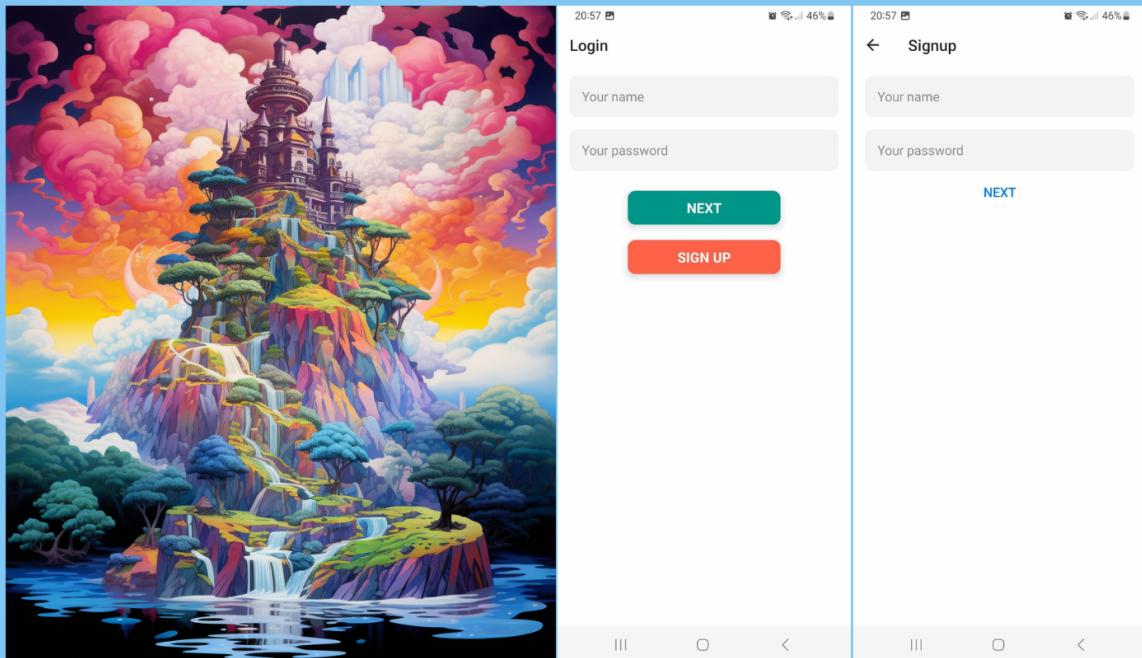
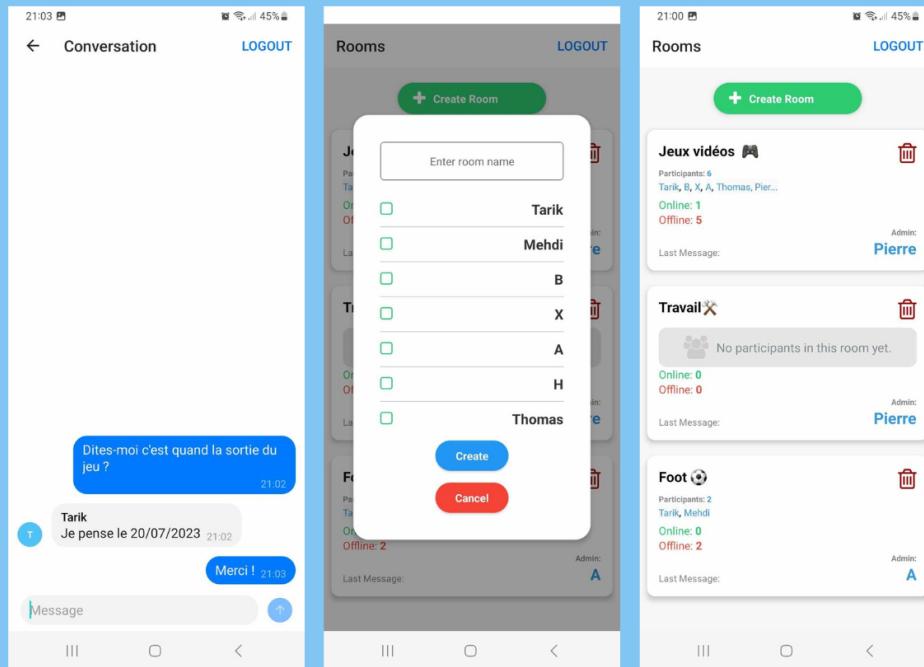
WireFrame:



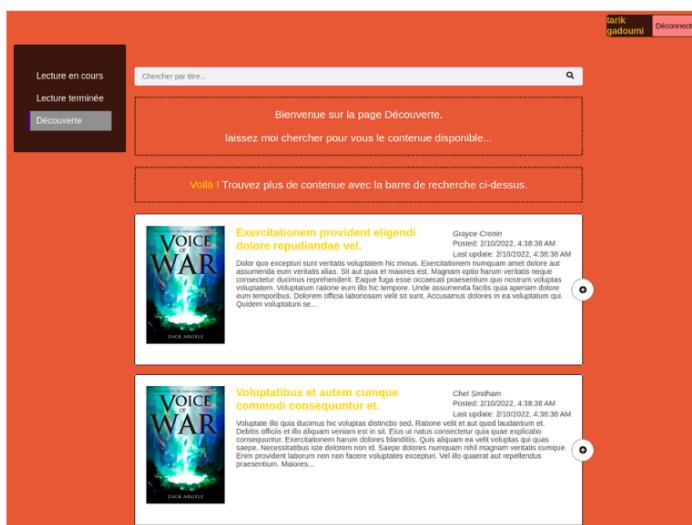
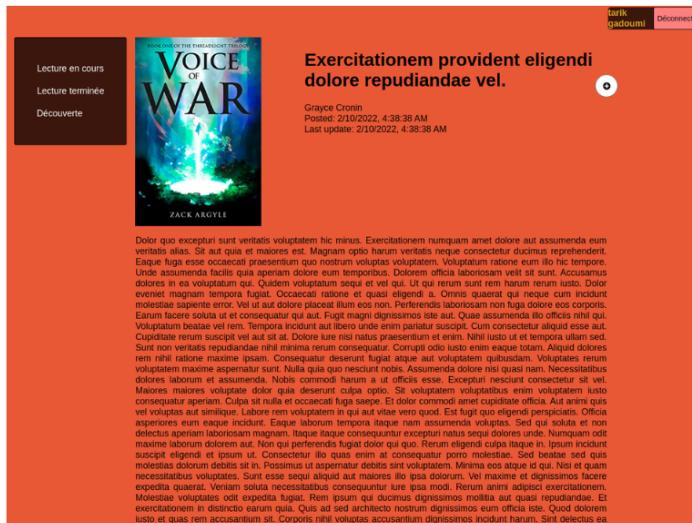
DOSSIER PROFESSIONNEL ^(DP)

Maquette :

DOSSIER PROFESSIONNEL (DP)



DOSSIER PROFESSIONNEL (DP)



DOSSIER PROFESSIONNEL (DP)



Activité-type 2

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

Exemple n° 1 - Le jeu du Sokoban - Développer une interface utilisateur de type desktop

DOSSIER PROFESSIONNEL (DP)

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

Ce projet valide les compétences :

- *Développer une interface utilisateur de type desktop*

Dans le cadre de ma formation, j'ai développé le jeu du Sokoban en Python.

Pour cela, j'ai utilisé la librairie Pygames, qui me permet de créer des éléments graphiques ainsi que d'écouter sur des événements bien précis comme sur celui des touches du clavier. Ce projet a été codé en Orienté Objet pour pouvoir au mieux faire interagir mes éléments et une plus grande lisibilité du code.

J'ai commencé par initier le jeu avec la fonction init de Pygame et définir la surface sur laquelle on va jouer. J'ai rajouté une fonction drawMap et drawPlayer pour dessiner le fond de mon interface afin d'être sur de la position de mes murs et de mon joueur.

```
# initialisation de Pygame
pygame.init()

# création de la fenêtre
screen = pygame.display.set_mode((LARGEUR, HAUTEUR))
pygame.display.set_caption(TITRE)

# chargement de l'image de fond
background = pygame.image.load("images/back.jpg")

# création de la grille et du joueur
_grille = Grille("levels/lv1.txt")
_grille.drawMap(screen)
_player = Player(_grille)
_player.drawPlayer(screen)

# affichage de la fenêtre
pygame.display.flip()
```

Ensuite, j'ai créé une classe Player pour gérer les différents événements de mon player.

DOSSIER PROFESSIONNEL (DP)

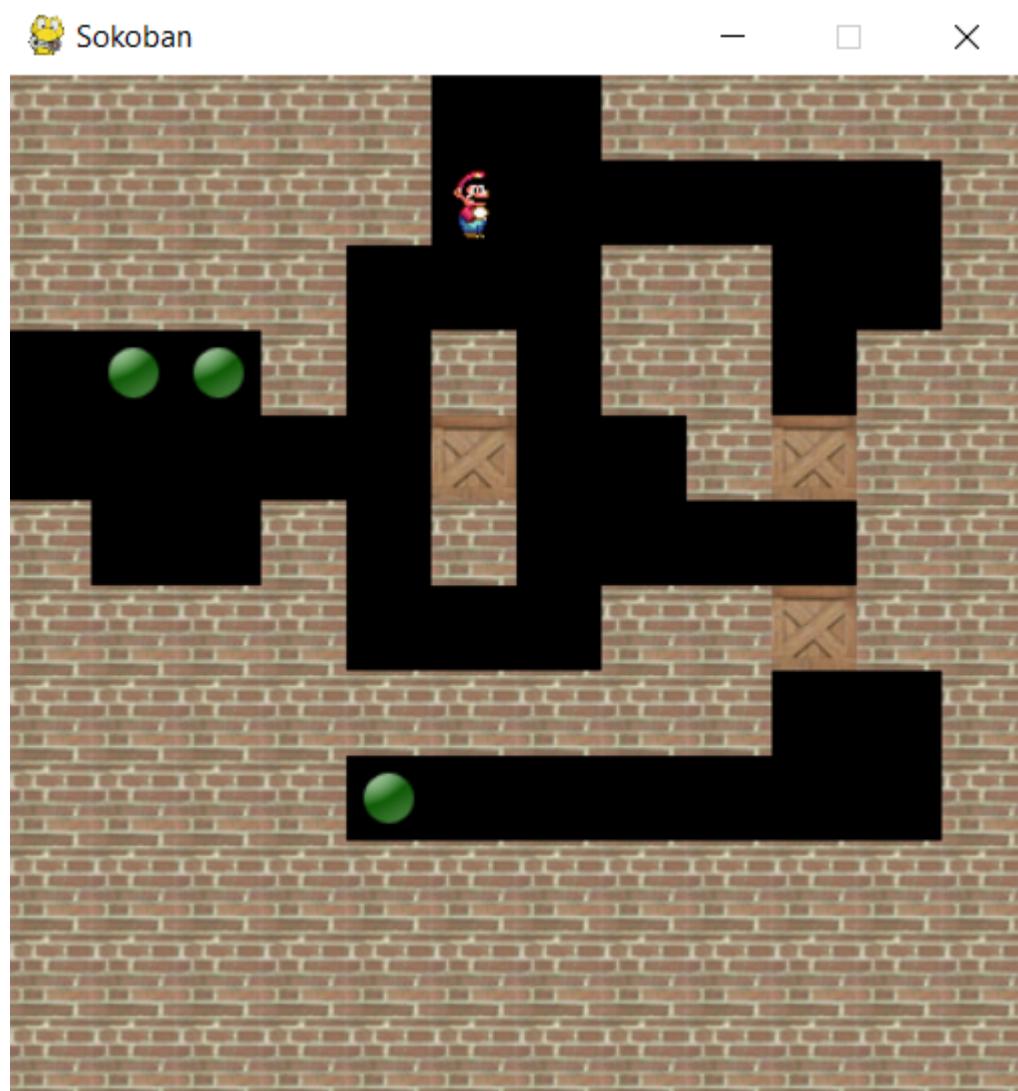
```
class Player:  
    def __init__(self, grille):  
        pygame.mixer.music.load("sounds/mvrasseli_play_the_game.mp3")  
        pygame.mixer.music.set_volume(0.5) # volume réglé à 50%  
        pygame.mixer.music.play()  
  
        self.gauche = pygame.image.load("images/mario_gauche.gif")  
        self.droite = pygame.image.load("images/mario_droite.gif")  
        self.bas = pygame.image.load("images/mario_bas.gif")  
        self.haut = pygame.image.load("images/mario_haut.gif")  
  
        self.position = self.droite  
  
        self.grille = grille  
        self.pos = self.grille.getPlayerPosition(self.grille)  
  
        self.x = self.pos[0] // SIZE  
        self.y = self.pos[1] // SIZE  
        self.id_joueur = None  
        self.score = 0 # initialisation du score à 0
```

Il ne me reste plus qu'à lancer une boucle qui mettra le jeu à jour avec les nouvelles donnée à chaque actions possible dans le jeux selon la touche pressé que j'aurai au préalable configuré.

DOSSIER PROFESSIONNEL (DP)

```
# boucle principale du jeu
continuer = True
while not _grille.is_fini():
    for event in pygame.event.get():
        if event.type == QUIT:
            # sauvegarde du score dans la base de données avant de quitter
            _grille.set_niveau(1) # définir le niveau atteint
            _player.set_id_joueur("joueur1") # définir l'identifiant du joueur
            query = "INSERT INTO scores (id_joueur, niveau, score) VALUES (%s, %s, %s)"
            params = (_player.id_joueur, _grille.niveau, _player.get_score())
            c.execute(query, params)
            conn.commit()
            sys.exit()
        if event.type == KEYDOWN:
            _player.move(event.key)
            # incrémente le score à chaque mouvement du joueur
            _player.set_score(_player.get_score() + 1)
            if event.key == K_r:
                # régénération de la grille et du joueur
                _grille = Grille("levels/lv1")
                _grille.drawMap(screen)
                _player = Player(_grille)
                _player.drawPlayer(screen)
```

DOSSIER PROFESSIONNEL (DP)



2. Précisez les moyens utilisés :

Pour développer le jeu du Sokoban en Python, j'ai utilisé plusieurs outils et méthodologies.

Tout d'abord, j'ai choisi le langage de programmation Python pour sa syntaxe claire et concise et sa polyvalence dans le développement de différents types d'applications, y compris les jeux.

Pour la conception et le développement de l'interface utilisateur, j'ai utilisé la bibliothèque

DOSSIER PROFESSIONNEL (DP)

Pygame. Pygame est un ensemble de modules Python conçus pour la création de jeux. Il fournit les fonctionnalités nécessaires pour créer des éléments graphiques et écouter des événements spécifiques, comme les touches du clavier.

Pour structurer le code du jeu, j'ai choisi une approche de programmation orientée objet. Cela a permis une meilleure interaction entre les éléments du jeu et a rendu le code plus lisible et facile à maintenir.

J'ai commencé par initialiser le jeu avec la fonction init de Pygame et définir la surface de jeu. J'ai ensuite ajouté une fonction drawMap pour dessiner le fond de l'interface, et une fonction drawPlayer pour dessiner le joueur.

Pour gérer les événements liés au personnage du jeu, j'ai créé une classe Player. Cette classe gère les actions du joueur en fonction des événements de clavier.

Enfin, pour faire tourner le jeu, j'ai mis en place une boucle de jeu qui met à jour le jeu avec de nouvelles données après chaque action possible, en fonction des touches pressées par le joueur.

Pour l'environnement de développement, j'ai utilisé un éditeur de texte de mon choix pour écrire le code, et une console Python pour exécuter le jeu et tester son fonctionnement. J'ai également utilisé Git pour le contrôle de version, ce qui m'a permis de garder une trace de chaque modification et d'éviter les problèmes lors de la modification du code.

3. Avec qui avez-vous travaillé ?

J'ai fait ce projet seul

4. Contexte

DOSSIER PROFESSIONNEL (DP)

Nom de l'entreprise, organisme ou association ▶ *LaPlateforme*

Chantier, atelier, service ▶ *MySokoban*

Période d'exercice ▶ Du : *07/03/2023* au : *22/03/2023*

5. Informations complémentaires (*facultatif*)

DOSSIER PROFESSIONNEL^(DP)

Activité-type 3 Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Exemple n° 1 - Blog e-learning et chat app

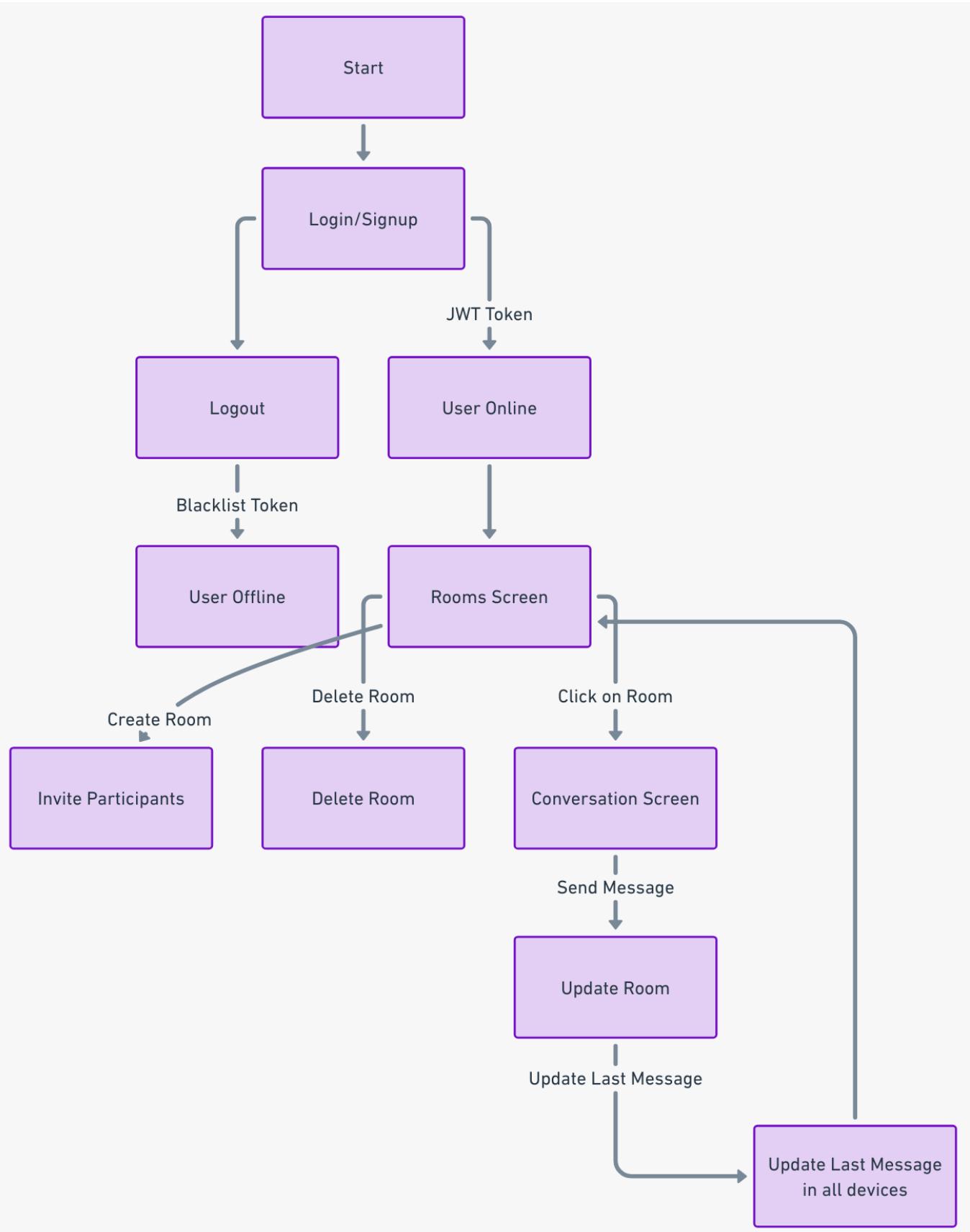
1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Ces projets valident les compétences :

- Concevoir une application
- Développer une application mobile
- Développer une application Web
- Développer des composants métier
- Construire des applications organisées en couches
- Développer des composants d'accès aux données
- Développer les parties front-end des interfaces utilisateurs
- Développer les parties back-end des interfaces utilisateurs

Chat app :

DOSSIER PROFESSIONNEL (DP)



DOSSIER PROFESSIONNEL (DP)

j'ai entrepris le développement d'un blog e-learning. Mon but était de créer une plateforme d'apprentissage en ligne interactive et efficace, en mettant l'accent sur l'expérience utilisateur et le suivi transparent du contenu.

J'ai commencé par une analyse de l'existant, en étudiant les différentes plateformes d'apprentissage en ligne existantes. Mon objectif était de comprendre les fonctionnalités clés que ces plateformes offrent et comment je pourrais intégrer de telles fonctionnalités dans mon propre projet. J'ai identifié des acteurs majeurs tels que Coursera et Udemy comme des points de référence pour mon projet.

J'ai ensuite défini les utilisateurs cibles du blog e-learning, qui sont les utilisateurs authentifiés qui peuvent suivre leur progression d'apprentissage, prendre des notes, ajouter du contenu à leur liste de lecture, évaluer les articles, et bien plus encore. Les administrateurs du blog ont également la possibilité d'ajouter du contenu et de suivre l'engagement des utilisateurs pour évaluer l'efficacité du contenu.

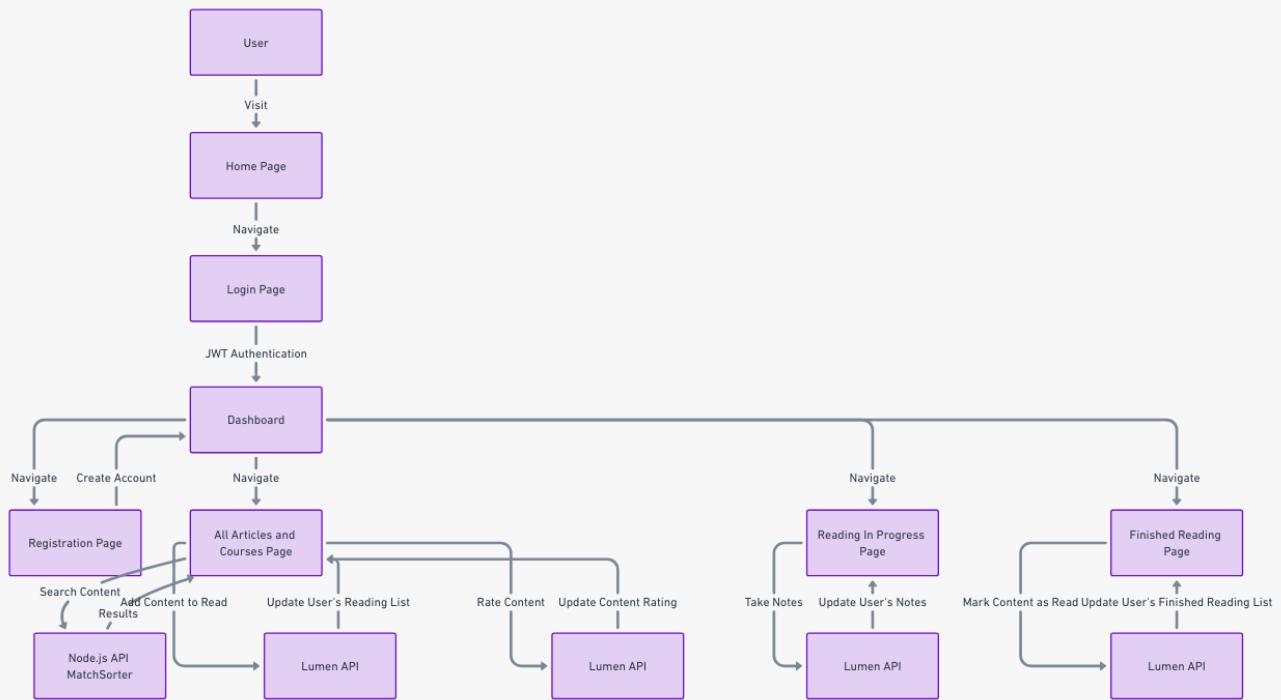
Pour le développement du blog e-learning, j'ai choisi d'utiliser JavaScript avec Node.js comme environnement de développement. J'ai également choisi d'utiliser le framework Express.js pour la création de l'API du blog, et React pour le développement front-end.

En ce qui concerne les fonctionnalités du blog, j'ai conçu plusieurs pages, notamment une page d'accueil, une page de connexion, une page d'inscription, une page qui liste tous les articles et cours, une page qui liste les éléments en cours de lecture, et une page qui liste les éléments dont la lecture est terminée. J'ai également intégré une fonctionnalité d'évaluation qui permet aux utilisateurs de noter les articles avec un système de coeurs.

Enfin, j'ai utilisé plusieurs outils pour soutenir le développement du blog. Par exemple, j'ai utilisé Visual Studio Code pour l'écriture du code, Postman pour les requêtes API, Git pour le versionnage du code, Trello pour l'organisation du travail, NPM pour l'installation des paquets, et Figma pour la conception des maquettes et de la charte graphique.

Blog e-learning :

DOSSIER PROFESSIONNEL (DP)



En parallèle du développement du blog e-learning, j'ai entrepris la réalisation d'une application de chat mobile. L'objectif était de proposer une plateforme de communication en temps réel, intuitive et conviviale.

Pour le développement de cette application mobile, j'ai opté pour le langage TypeScript, un sur-ensemble typé de JavaScript. Ce choix m'a permis de bénéficier des avantages du typage statique pour une meilleure qualité de code et une détection d'erreur plus efficace. De plus, TypeScript est largement utilisé dans le développement d'applications modernes, ce qui m'a permis d'acquérir des compétences précieuses.

J'ai également utilisé le framework React Native pour le développement de l'application mobile. Ce framework, écrit en TypeScript, permet le développement d'un seul code pour les plateformes iOS et Android, ce qui permet d'économiser beaucoup de temps et d'efforts.

Pour l'aspect communication en temps réel, j'ai fait le choix d'utiliser GraphQL, notamment avec l'utilisation d'Apollo Server. GraphQL est une alternative à REST et offre un langage de requête pour vos API ainsi qu'un temps d'exécution pour exécuter ces requêtes avec vos données existantes. L'utilisation de GraphQL permet une récupération de données plus efficace, ce qui est essentiel pour une application de chat en temps réel.

L'application de chat mobile comporte plusieurs fonctionnalités clés, notamment une page d'accueil qui redirige vers la page de connexion ou d'inscription, une page d'inscription pour

DOSSIER PROFESSIONNEL (DP)

recueillir des informations de base sur l'utilisateur, une page de connexion sécurisée, une page de chat pour la communication en temps réel, une page de profil pour gérer les informations personnelles et une page de contact qui affiche la liste de tous les utilisateurs de l'application.

En plus de TypeScript, React Native et GraphQL, j'ai utilisé une suite d'outils pour soutenir le développement de l'application de chat mobile. Ces outils comprennent Visual Studio Code pour l'écriture du code, Git pour le versionnage du code, Trello pour l'organisation du travail, NPM pour l'installation des paquets, et Expo pour émuler l'application mobile sur mon téléphone.

J'ai également utilisé JWT pour l'authentification des utilisateurs, garantissant ainsi la sécurité des échanges et la confidentialité des informations des utilisateurs.

Comme pour le blog e-learning, j'ai réalisé ce projet tout au long de mon année de formation, alternant entre des périodes en entreprise et en centre de formation. J'ai utilisé Trello pour gérer efficacement mon temps et mes tâches, ce qui m'a aidé à respecter mes échéances et à livrer une application mobile de chat fonctionnelle et efficace.

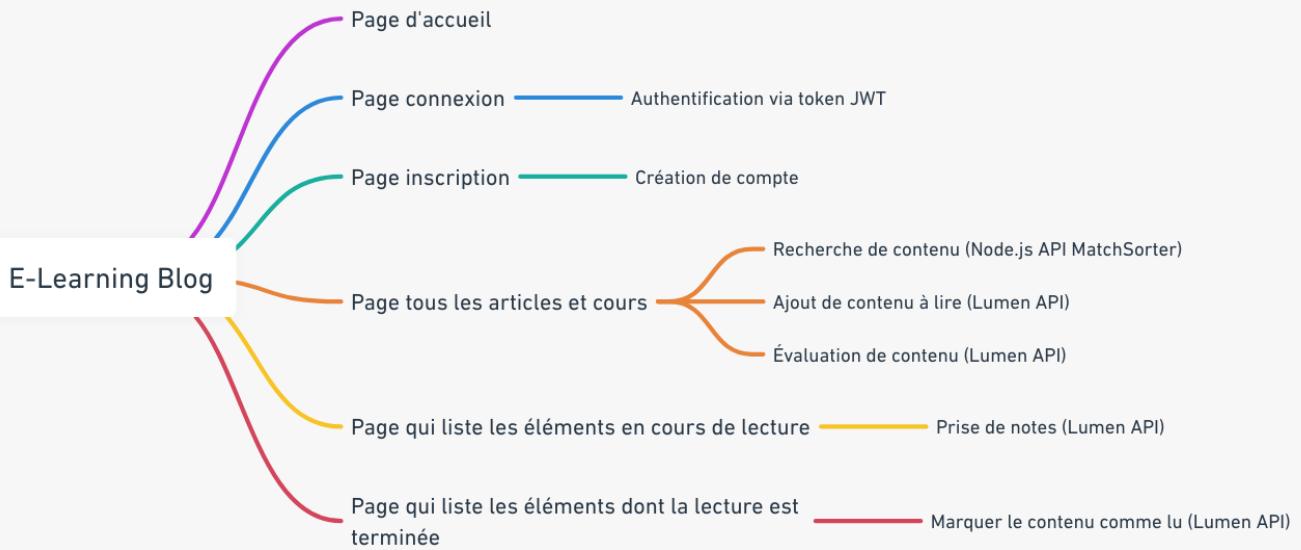
2. Précisez les moyens utilisés :

Pour réaliser le blog e-learning, j'ai mobilisé divers moyens techniques et méthodologiques. Voici une liste détaillée des outils et techniques que j'ai utilisés :

Choix de développement : J'ai choisi JavaScript avec Node.js comme environnement de développement. Ce choix a été motivé par la richesse de JavaScript et sa large utilisation, offrant de nombreux concepts et bibliothèques open-source qui facilitent le développement de certaines fonctionnalités.

Frameworks : Pour la création de l'**API du blog**, j'ai utilisé Express.js, un framework Node.js. Pour le développement front-end, j'ai choisi React pour sa flexibilité, sa performance, et son soutien communautaire.

DOSSIER PROFESSIONNEL (DP)



Logiciels et autres outils : J'ai fait usage d'une suite d'outils pour soutenir le développement du blog. Ces outils incluent :

Visual Studio Code pour l'écriture du code.

Postman pour effectuer les requêtes API.

Git pour le versionnage du code.

Trello pour l'organisation du travail.

NPM pour l'installation des paquets.

Figma pour la création de mes maquettes et de la charte graphique.

Outils personnalisés : J'ai développé et utilisé un outil CLI personnalisé, **jujutsu-FF-maker**, pour améliorer ma productivité. Cet outil permet de générer des structures de dossiers et de fichiers personnalisés, ainsi que des modèles de scripts. Il offre également une option de création assistée par intelligence artificielle pour plus d'efficacité.

Organisation du projet : L'organisation du projet a été gérée avec l'aide de Trello. Cet outil m'a permis de lister et de suivre les différentes tâches à réaliser, facilitant ainsi la gestion de mon travail et le respect de mes échéances.

3. Avec qui avez-vous travaillé ?

seul

4. Contexte

DOSSIER PROFESSIONNEL (DP)

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

Chantier, atelier, service ▶ *Blog e-learning et chat app*

Période d'exercice ▶ Du : *01/03/!2023* au : *31/06/2023*

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.

DOSSIER PROFESSIONNEL (DP)

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 :

DOSSIER PROFESSIONNEL ^(DP)

Documents illustrant la pratique professionnelle

(facultatif)

Intitulé

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

ANNEXES

(Si le RC le prévoit)