



# DOSSIER PROFESSIONNEL (DP)

*Nom de naissance* ➤ Entrez votre nom de naissance ici.  
*Nom d'usage* ➤ Entrez votre nom d'usage ici.  
*Prénom* ➤ Entrez votre prénom ici.  
*Adresse* ➤ Entrez votre adresse ici.

## Titre professionnel visé

*Cliquez ici pour entrer l'intitulé du titre professionnel visé.*

### 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>Développer une application sécurisée</b>	<b>p.</b>	<b>5</b>
- Développement du site web - RushMaster	p.	5
- Développement d'une API de configuration - PowerCard	p.	9
<b>Concevoir et développer une application sécurisée organisée en couches</b>	<b>p.</b>	
- Conception et développement du site web - RushMaster	p.	11
- Conception et développement de la page d'administration - PowerCard	p.	15
- Conception et développement d'un gestionnaire de mot de passe	p.	18
<b>Préparer le déploiement d'une application sécurisée</b>	<b>p.</b>	
- Préparer le déploiement du site web - RushMaster	p.	20
<b>Titres, diplômes, CQP, attestations de formation</b> <i>(facultatif)</i>	<b>p.</b>	<b>24</b>
<b>Déclaration sur l'honneur</b>	<b>p.</b>	<b>25</b>
<b>Documents illustrant la pratique professionnelle</b> <i>(facultatif)</i>	<b>p.</b>	<b>26</b>
<b>Annexes</b> <i>(Si le RC le prévoit)</i>	<b>p.</b>	<b>27</b>

# **EXEMPLES DE PRATIQUE PROFESSIONNELLE**

## Activité-type 1 Développement d'une application sécurisée

Exemple n°1 - Développement du site web - RushMaster

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

#### Contexte :

RushMaster est une application web full-stack développée dans le cadre d'un projet de fin d'études (Titre RNCP niveau 6 – Concepteur Développeur d'Applications).

L'objectif était de proposer un outil en ligne permettant aux joueurs de League of Legends de :

- Consulter leurs statistiques de jeu en temps réel
- Accéder à l'historique de leurs parties
- Analyser leurs performances
- Comparer leurs résultats avec ceux d'autres joueurs

Le projet a été réalisé en **binôme** sur six mois, avec une méthodologie **Agile** (sprints hebdomadaires, réunions Discord, gestion des tâches sur Trello).

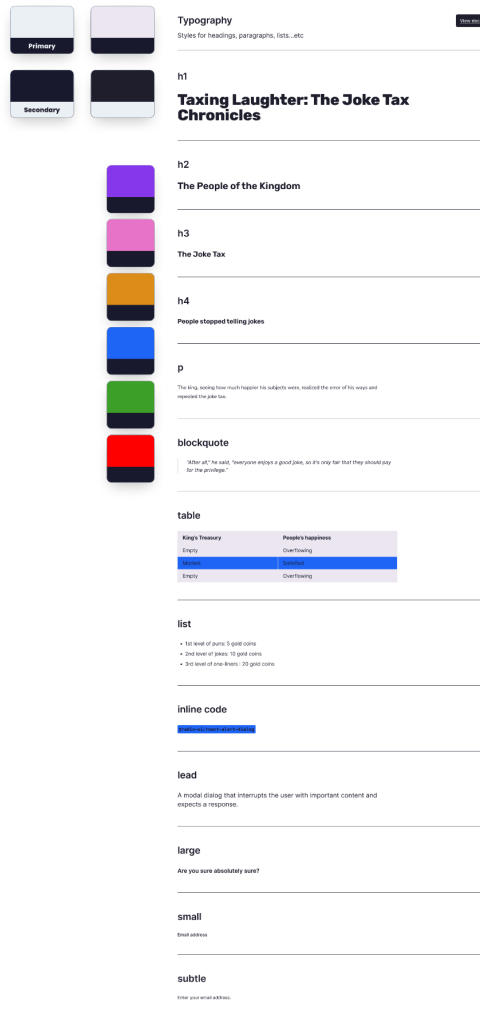
L'intégration avec l'API Riot Games nécessitait une gestion sécurisée des clés API et un traitement fiable des données.

Une attention particulière a été portée à la **sécurité** (authentification OAuth2, protection des endpoints backend, validation des entrées) et à la **performance** (mise en cache local des données, optimisation des requêtes).

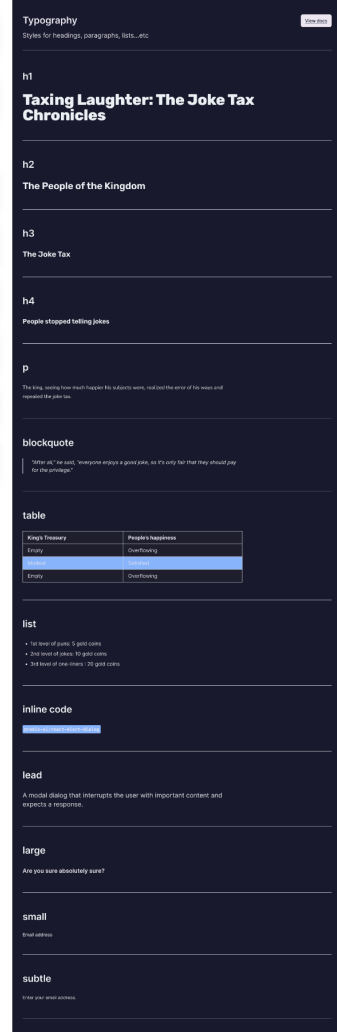
#### Tâches réalisées :

- **Mise en place de l'environnement de développement sécurisé :**
  - Installation de **Docker** et **Docker Compose** pour isoler le frontend (React), le backend (Spring Boot) et la base de données (PostgreSQL)
  - Configuration des réseaux Docker pour limiter l'exposition des services
  - Gestion des variables sensibles (clés API, mots de passe) via fichiers **.env** non versionnés
- **Développement du frontend sécurisé :**
  - Création d'une interface responsive avec **React** et **Material-UI**

## GraphChart (Light Theme)



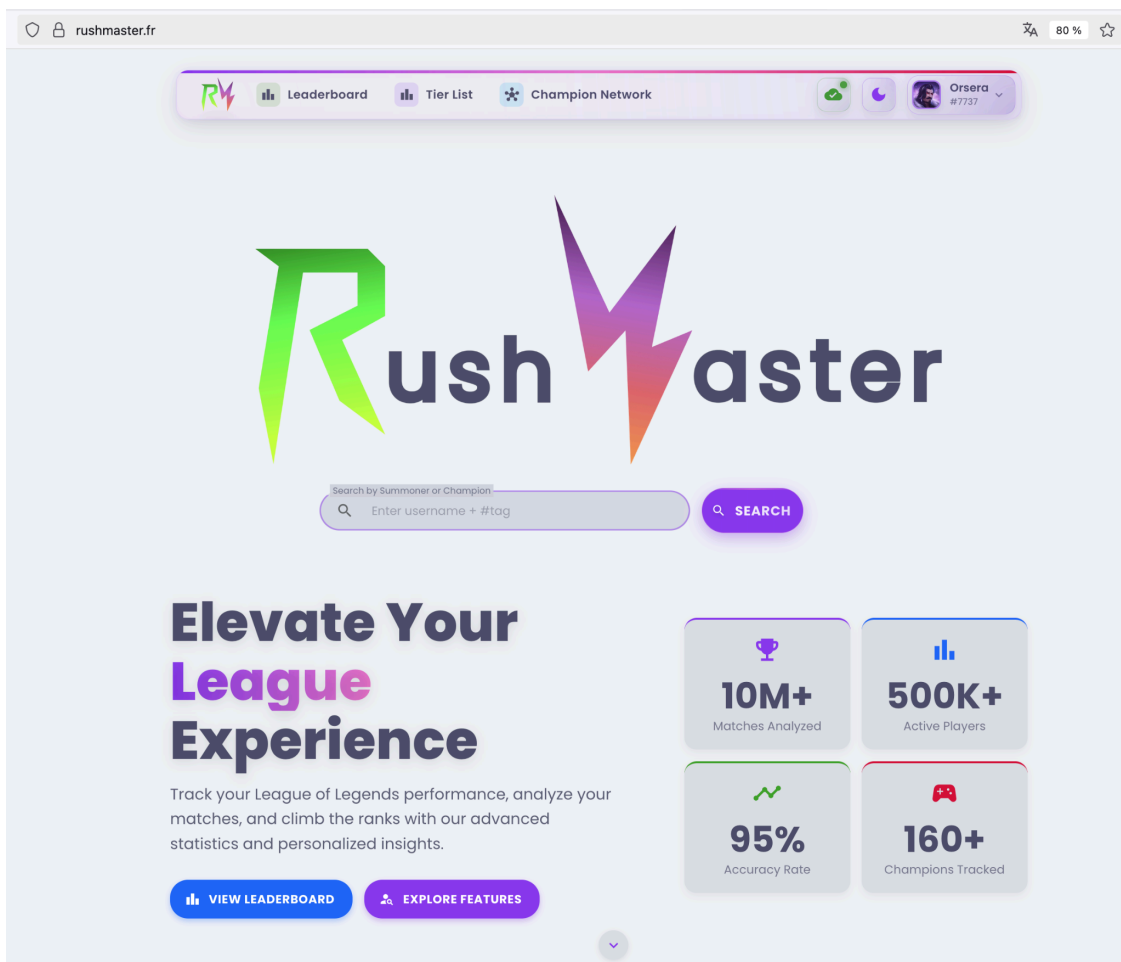
## GraphChart (Dark Theme)



- Composants réutilisables pour la recherche de joueurs, affichage de statistiques, historique des parties
- Validation côté client des entrées utilisateur pour éviter les injections et requêtes invalides
- Implémentation d'un système de **mode clair/sombre**
- **Développement du backend sécurisé :**
  - Création d'une **API REST** avec **Spring Boot** exposant des endpoints sécurisés

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

- Intégration avec l'API Riot Games pour récupérer les données de joueurs et de champions
- Implémentation de l'authentification **OAuth2** fournie par Riot Games
- Gestion des accès via **Spring Security** (rôles, autorisations)
- Mise en cache des données pour éviter les appels API excessifs et améliorer la performance



## 2. Précisez les moyens utilisés :

Gestion de version : Git (GitHub)

# DOSSIER PROFESSIONNEL (DP)

**Gestion de projet** : Trello (Kanban), Notion (documentation), GitHub Issues (tickets/bugs)

**Communication** : Discord, e-mails, messagerie instantanée

**Conception graphique** : Figma (maquettes, logo)

**IDE** : IntelliJ IDEA (Java backend & React frontend), Visual Studio Code (POC Python)

**Conteneurisation** : Docker, Docker Compose

**Hébergement** : VPS Hostinger KVM4 (16 Go RAM, 200 Go stockage)

## 3. Avec qui avez-vous travaillé ?

Collaboration en binôme avec **Clément Marin**, étudiant dans la même promotion de bachelor IT à La Plateforme, spécialisation logiciel.

## 4. Contexte

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

Chantier, atelier, service ▶ *Dans le cadre de la formation : Développement d'une application web.*

Période d'exercice ▶ Du : *20/12/24* au : *30/07/25*

## 5. Informations complémentaires (facultatif)



## Activité-type 1 Développement d'une application sécurisée

Exemple n° 2 - Développement d'une API de configuration - PowerCard

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

Lors de mon alternance chez **Acpqualife**, j'ai travaillé pendant 8 mois sur une fonctionnalité visant à **redévelopper la gestion de nos passerelles SMS**. En effet, certains de nos clients, notamment les processeurs bancaires, hébergent parfois plusieurs passerelles afin de garantir la redondance ou de gérer différents flux de communication.

L'implémentation existante présentait plusieurs limitations : pour chaque mise à jour ou chaque nouvel enrôlement de client, il était nécessaire de **redéployer l'intégralité du produit**, entraînant une **interruption de service** pour toutes les passerelles hébergées par la banque concernée.

Ma mission a donc été découpée en trois objectifs principaux :

1. Permettre une **livraison asynchrone**, indépendante des livraisons mensuelles habituelles.
2. Offrir la possibilité de **configurer une passerelle par banque ou par produit**, et non plus une unique passerelle par client.
3. Mettre en place une **interface graphique** dédiée à la configuration de ces passerelles.

Sur une période de huit mois, j'ai mené à bien ces trois volets.

Pour le premier point, j'ai déplacé les propriétés auparavant stockées dans les fichiers d'environnement Java vers la base de données. Ainsi, une simple requête SQL permet désormais de modifier la configuration sans nécessiter un redéploiement complet.

Pour le second, j'ai effectué un travail important sur le **code legacy**, incluant une restructuration et une adaptation de l'architecture afin de supporter la gestion multi-passerelles.

Enfin, pour le dernier point, j'ai développé une **page Angular** permettant d'administrer facilement les configurations via une interface ergonomique.

### 2. Précisez les moyens utilisés :

J'ai dû intervenir sur **trois volets principaux** :

- **La base de données** sous **Oracle SQL**, pour la gestion et la structuration des données.

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

- **Le backend** en **Java Spring**, pour la logique métier et les services applicatifs.
- **Le frontend** avec **Angular TypeScript**, pour l'interface utilisateur et les interactions côté client.

L'ensemble de ces développements a été réalisé en respectant les **exigences de couverture de tests** imposées par **Jenkins** et **SonarLint**, garantissant ainsi la qualité et la maintenabilité du code.

### 3. Avec qui avez-vous travaillé ?

Pour ce projet, j'ai été le seul développeur en charge du développement. Cependant, j'ai pu bénéficier tout au long du processus des conseils précieux d'un profil senior de l'entreprise, Hervé Milesi, qui m'a non seulement guidé sur ce projet, mais m'a également beaucoup aidé à monter en compétences.

### 4. Contexte

Nom de l'entreprise, organisme ou association ▶ *AcpQualife (groupe HPS)*

Chantier, atelier, service ▶ *Dans le cadre de l'entreprise: feature*

Période d'exercice ▶ Du : *03/2024* au : *Cliquez ici 12/2024*

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

## Activité-type 2

### Concevoir et développer une application sécurisée organisée en couches

Exemple n° 1 - Conception et développement du site web - RushMaster

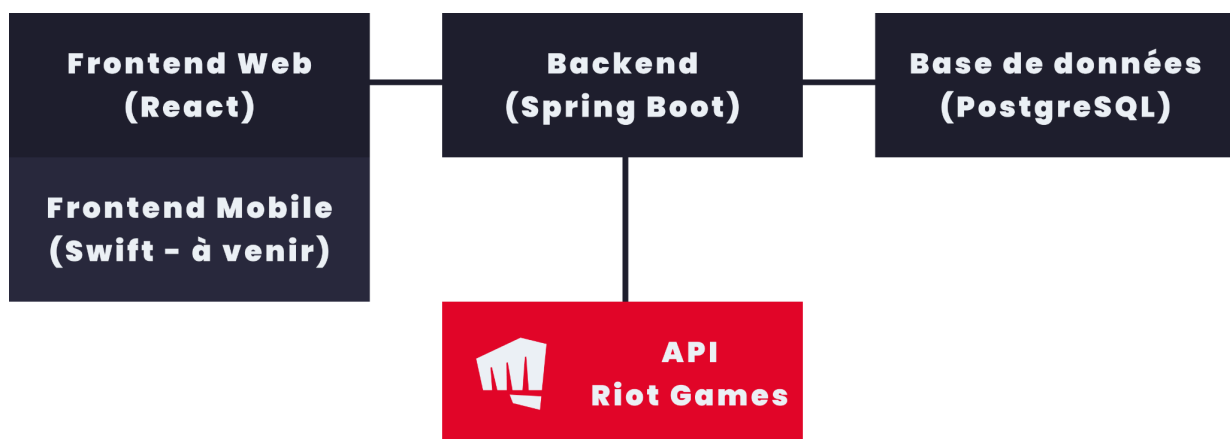
#### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

##### Contexte :

L'architecture de RushMaster a été pensée dès le départ pour être **modulaire, maintenable et sécurisée**.

L'application suit une organisation en **trois couches** distinctes :

1. **Présentation (Frontend)** – Interface utilisateur en React
2. **Logique métier (Backend)** – API REST sécurisée en Spring Boot
3. **Accès aux données (Base de données)** – PostgreSQL avec Spring Data JPA



Cette séparation permet de limiter les dépendances entre couches, d'isoler les responsabilités et de faciliter la maintenance.

La conception a commencé par une **analyse fonctionnelle détaillée** et la réalisation de **maquettes interactives** avec **Figma**, validées avant le développement.

L'intégration avec l'API Riot Games a nécessité la mise en place d'un système d'accès aux données optimisé, capable de gérer les quotas API et de mettre en cache les résultats.

##### Tâches réalisées :

- **Analyse des besoins et conception initiale :**

- Recueil et formalisation des besoins fonctionnels et non-fonctionnels
- Création de maquettes haute-fidélité avec Figma (versions desktop et mobile)
- Définition de la charte graphique et du thème clair/sombre

- **Définition de l'architecture logicielle :**

- Découpage en 3 couches indépendantes (Frontend / Backend / Base de données)
- Conception d'un **modèle relationnel** pour PostgreSQL avec scripts SQL d'initialisation
- Séparation des environnements (dev, test, prod) avec configurations spécifiques

(DP)



## Données Riot Games

- **Développement backend :**

- Implémentation d'une API REST avec **Spring Boot**
- Intégration sécurisée de l'API Riot Games avec gestion des quotas et mise en cache
- Accès aux données via **Spring Data JPA**
- Validation côté serveur des entrées utilisateur
- Gestion des rôles et permissions via **Spring Security**

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

- **Développement frontend :**

- Création de composants React réutilisables et testés individuellement
- Affichage dynamique des statistiques, historique de parties et profils joueurs
- Gestion du state global avec **Redux** pour fiabiliser les données affichées
- Filtres et tris dynamiques sur les pages leaderboard et tier list

## 2. Précisez les moyens utilisés :

**Gestion de version :** Git (GitHub)

**Gestion de projet :** Trello (Kanban), Notion (documentation), GitHub Issues (tickets/bugs)

**Communication :** Discord, e-mails, messagerie instantanée

**Conception graphique :** Figma (maquettes, logo)

**IDE :** IntelliJ IDEA (Java backend & React frontend), Visual Studio Code (POC Python)

**Conteneurisation :** Docker, Docker Compose

**Hébergement :** VPS Hostinger KVM4 (16 Go RAM, 200 Go stockage)

## 3. Avec qui avez-vous travaillé ?

Collaboration en binôme avec **Clément Marin**, étudiant dans la même promotion de bachelor IT à La Plateforme, spécialisation logiciel.

## 4. Contexte

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

Chantier, atelier, service ▶ *Dans le cadre de la formation : Développement d'une application web.*

Période d'exercice ▶ Du : *20/12/24* au : *30/07/25*

## 5. Informations complémentaires (facultatif)

## Activité-type 2 Concevoir et développer une application sécurisée organisée en couches

*Exemple n° 2* - Conception et développement de la page d'administration - PowerCard

---

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

Lors de mon alternance chez **Acpqualife**, j'ai travaillé pendant 8 mois sur une fonctionnalité visant à **redévelopper la gestion de nos passerelles SMS**. En effet, certains de nos clients, notamment les processeurs bancaires, hébergent parfois plusieurs passerelles afin de garantir la redondance ou de gérer différents flux de communication.

L'implémentation existante présentait plusieurs limitations : pour chaque mise à jour ou chaque nouvel enrôlement de client, il était nécessaire de **redéployer l'intégralité du produit**, entraînant une **interruption de service** pour toutes les passerelles hébergées par la banque concernée.

Ma mission a donc été découpée en trois objectifs principaux :

4. Permettre une **livraison asynchrone**, indépendante des livraisons mensuelles habituelles.
5. Offrir la possibilité de **configurer une passerelle par banque ou par produit**, et non plus une unique passerelle par client.
6. Mettre en place une **interface graphique** dédiée à la configuration de ces passerelles.

Sur une période de huit mois, j'ai mené à bien ces trois volets.

Pour le premier point, j'ai déplacé les propriétés auparavant stockées dans les fichiers

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

d'environnement Java vers la base de données. Ainsi, une simple requête SQL permet désormais de modifier la configuration sans nécessiter un redéploiement complet.

Pour le second, j'ai effectué un travail important sur le **code legacy**, incluant une restructuration et une adaptation de l'architecture afin de supporter la gestion multi-passerelles.

Enfin, pour le dernier point, j'ai développé une **page Angular** permettant d'administrer facilement les configurations via une interface ergonomique.

## 2. Précisez les moyens utilisés :

J'ai dû intervenir sur **trois volets principaux** :

- **La base de données** sous **Oracle SQL**, pour la gestion et la structuration des données.
- **Le backend** en **Java Spring**, pour la logique métier et les services applicatifs.
- **Le frontend** avec **Angular TypeScript**, pour l'interface utilisateur et les interactions côté client.

L'ensemble de ces développements a été réalisé en respectant les **exigences de couverture de tests** imposées par **Jenkins** et **SonarLint**, garantissant ainsi la qualité et la maintenabilité du code.

## 3. Avec qui avez-vous travaillé ?

Pour ce projet, j'ai été le seul développeur en charge du développement. Cependant, j'ai pu bénéficier tout au long du processus des conseils précieux d'un profil senior de l'entreprise, Hervé Milesi, qui m'a non seulement guidé sur ce projet, mais m'a également beaucoup aidé à monter en compétences.

## 4. Contexte

Nom de l'entreprise, organisme ou association ▶ *AcpQualife (groupe HPS)*

Chantier, atelier, service ▶ *Dans le cadre de l'entreprise: feature*

Période d'exercice ▶ Du : *03/2024* au : *Cliquez ici 12/2024*

## 5. Informations complémentaires (facultatif)



---

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

---

---

## Activité-type 2 Concevoir et développer une application sécurisée organisée en couches

**Exemple n° 3** - Conception et développement d'un gestionnaire de mot de passe

---

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

Dans le cadre de la validation des compétences requises pour l'obtention du titre de **Concepteur Développeur d'Applications (CDA)**, j'ai développé un **gestionnaire de mots de passe** en **Python**, utilisant **MongoDB** pour stocker de manière sécurisée les mots de passe préalablement hachés.

Ce projet, inscrit dans un contexte purement scolaire, a été intégralement conçu et réalisé en autonomie. Sa mise en œuvre m'a demandé un peu moins d'une semaine, depuis la phase de conception jusqu'aux tests finaux.

### 2. Précisez les moyens utilisés :

**Python 3.x** – Langage principal pour la logique applicative

**PyMongo** – Bibliothèque Python pour interagir avec **MongoDB**

**MongoDB** – Base de données NoSQL utilisée pour stocker les identifiants et mots de passe

### 3. Avec qui avez-vous travaillé ?

Ce projet, réalisé en autonomie complète, a été mené du début à la fin – de la conception jusqu'aux tests finaux – en un peu moins d'une semaine.

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

## 4. Contexte

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

Chantier, atelier, service ▶ *Dans le cadre de la formation : Développement d'une application.*

Période d'exercice ▶ Du : *24/07/25* au : *30/07/25*

## 5. Informations complémentaires *(facultatif)*

## Activité-type 3 Préparer le déploiement d'une application sécurisée

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 :

#### Contexte :

Le déploiement de RushMaster a été conçu pour être **automatisé, reproductible et sécurisé**. L'application est hébergée sur un **VPS Hostinger** (4 vCPU, 16 Go RAM, 200 Go SSD NVMe) sous **Linux Ubuntu**, avec conteneurisation via Docker et orchestration via Docker Compose.

Un pipeline CI/CD a été mis en place avec **GitHub Actions** pour automatiser les tests et les déploiements.

La sécurité est assurée par l'usage de certificats SSL **Let's Encrypt** et par la gestion stricte des variables sensibles.

#### Tâches réalisées :

- **Préparation de l'environnement serveur :**
  - Installation et configuration d'Ubuntu Server
  - Mise en place du pare-feu (UFW) avec ouverture des ports nécessaires (22, 80, 443)
  - Création d'un utilisateur dédié au déploiement sécurisé en SSH
- **Conteneurisation :**
  - Création de fichiers **Dockerfile** distincts pour le frontend et le backend
  - Fichier **docker-compose.yml** orchestrant les services (frontend, backend, base PostgreSQL)
  - Configuration réseau interne Docker pour limiter l'exposition externe des conteneurs
- **Mise en place du SSL et sécurité :**

- Obtention et configuration des certificats SSL Let's Encrypt pour rushmaster.fr
- Configuration Nginx dans le conteneur frontend pour forcer le HTTPS
- Gestion des permissions sur les certificats pour éviter les accès non autorisés
- **Automatisation du déploiement :**
  - Création d'un pipeline **GitHub Actions** déclenché à chaque push sur **main**
  - Étapes automatiques :
    1. Validation des fichiers Docker Compose
    2. Connexion SSH au VPS
    3. Pull du code et des sous-modules
    4. Reconstruction et redémarrage des conteneurs
    5. Renouvellement automatique des certificats SSL si nécessaire

The screenshot shows the GitHub Actions interface for the repository 'clement-ramos / Rush\_Master'. The 'Actions' tab is selected, displaying a list of workflow runs for the 'Main CI/CD' workflow. The runs are listed in a table with columns for Event, Status, Branch, and Actor. All runs are successful (green checkmark) and were triggered by 'Manually run by Marin-Clement' on the 'main' branch. The runs are ordered by time, with the most recent at the top. The table shows 91 workflow runs in total.

Event	Status	Branch	Actor
Manually run by Marin-Clement	Success	main	Marin-Clement
Manually run by Marin-Clement	Success	main	Marin-Clement
Manually run by Marin-Clement	Success	main	Marin-Clement
Manually run by Marin-Clement	Success	main	Marin-Clement
Manually run by Marin-Clement	Success	main	Marin-Clement
Manually run by Marin-Clement	Success	main	Marin-Clement
Manually run by Marin-Clement	Success	main	Marin-Clement
Manually run by Marin-Clement	Success	main	Marin-Clement
Manually run by Marin-Clement	Success	main	Marin-Clement
Manually run by Marin-Clement	Success	main	Marin-Clement

- **Documentation et maintenance :**

- Rédaction d'un guide complet pour installer, configurer et mettre à jour l'application
- Procédures de sauvegarde régulière de la base de données
- Mise en place de la supervision post-déploiement (logs et monitoring API)

## Deployment Automation

This document explains the automated deployment process set up for the Rush Master application.

### Overview

The GitHub Actions workflow in `.github/workflows/main-ci.yml` has been configured to automatically deploy updates to the VPS whenever changes are pushed to the main branch or when the workflow is manually triggered.

### Workflow Steps

1. **Validation:** Checks that the Docker Compose configuration files are valid.
2. **Deployment:** Connects to the VPS and performs the following actions:
  - Navigates to `/opt/rush-master`
  - Pulls the latest changes from the main repository
  - Updates all submodules to their latest commits
  - Rebuilds and restarts the Docker containers

### Required GitHub Secrets

For the deployment to work correctly, you need to add the following secrets to your GitHub repository:

1. **VPS\_HOST:** The IP address or hostname of your VPS server.
2. **VPS\_SSH\_USERNAME:** The username for SSH access to the VPS (usually "root" or another user with appropriate permissions).
3. **VPS\_SSH\_PRIVATE\_KEY:** The private SSH key for authentication to the VPS. This must include the complete key content with BEGIN and END lines.

### How to Add Secrets to GitHub Repository

1. Go to your GitHub repository
2. Click on "Settings"
3. In the left sidebar, click on "Secrets and variables" > "Actions"
4. Click on "New repository secret"
5. For VPS\_HOST:
  - Name: VPS\_HOST
  - Value: The IP address or hostname of your VPS server (e.g., 91.108.102.201)
6. For VPS\_SSH\_USERNAME:
  - Name: VPS\_SSH\_USERNAME
  - Value: Your SSH username for the VPS
7. For VPS\_SSH\_PRIVATE\_KEY:
  - Name: VPS\_SSH\_PRIVATE\_KEY
  - Value: Copy the entire content of your private key file (e.g., `~/.ssh/id_ed25519`), including the BEGIN and END lines like:

```
-----BEGIN OPENSSH PRIVATE KEY-----  
[key content here]  
-----END OPENSSH PRIVATE KEY-----
```



## 2. Précisez les moyens utilisés :

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

**Gestion de version** : Git (GitHub)

**Gestion de projet** : Trello (Kanban), Notion (documentation), GitHub Issues (tickets/bugs)

**Communication** : Discord, e-mails, messagerie instantanée

**Conception graphique** : Figma (maquettes, logo)

**IDE** : IntelliJ IDEA (Java backend & React frontend), Visual Studio Code (POC Python)

**Conteneurisation** : Docker, Docker Compose

**Hébergement** : VPS Hostinger KVM4 (16 Go RAM, 200 Go stockage)

## 3. Avec qui avez-vous travaillé ?

Collaboration en binôme avec **Clément Marin**, étudiant dans la même promotion de bachelor IT à La Plateforme, spécialisation logiciel.

## 4. Contexte

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

Chantier, atelier, service ▶ *Dans le cadre de la formation : Développement d'une application web.*

Période d'exercice ▶ Du : *20/12/24* au : *30/07/25*

## 5. Informations complémentaires (facultatif)

---

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

---

---

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

*(facultatif)*

Intitulé	Autorité ou organisme	Date
Bachelor IT, spécialisation logiciel	LaPlateforme	2022-2025
DUT GEII	IUT Montpellier - Sète	2020-2022
Bac S.SI	Maurice Janetti	2020



### Déclaration sur l'honneur

---

Je soussigné(e) **Clément Ramos** ,  
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 **29/07/25**  
pour faire valoir ce que de droit.

Signature :

Essai gratuit



Créé par Paint S

### Documents illustrant la pratique professionnelle

*(facultatif)*

Intitulé
Cliquez ici pour taper du texte.

---

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

---

## ANNEXES

*(Si le RC le prévoit)*