



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Nom de naissance ▶ SPRIGGS
Nom d'usage ▶
Prénom ▶ Julia
Adresse ▶ 458 cours de la libération, appt D101, 33400 Talence

Titre professionnel visé

Concepteur développeur d'applications

MODALITE D'ACCES :

- 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.

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



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

DOSSIER PROFESSIONNEL ^(DP)

Sommaire

Exemples de pratique professionnelle

Développer une application sécurisée

p. 4

▶ Intitulé de l'exemple n° 1 : Installation et configuration de l'environnement pour le projet de formation

p. 4

▶ Intitulé de l'exemple n° 2

p. p.

▶ Intitulé de l'exemple n° 3

p p.

Concevoir et développer une application sécurisée organisée en couches

p. 12

▶ Intitulé de l'exemple n° 1 : Conception et mise en place d'une base de données relationnelles pour le projet de formation

p. 12

▶ Intitulé de l'exemple n° 2

p. p.

▶ Intitulé de l'exemple n° 3

p p.

Préparer le déploiement d'une application sécurisée

p. 20

▶ Intitulé de l'exemple n° 1 : Mise en place et documentation du déploiement pour le projet de formation

p. 20

▶ Intitulé de l'exemple n° 2

p. p.

▶ Intitulé de l'exemple n° 3

p p.

Titres, diplômes, CQP, attestations de formation *(facultatif)*

p. 27

Déclaration sur l'honneur

p. 28

Documents illustrant la pratique professionnelle *(facultatif)*

p. 29

Annexes *(Si le RC le prévoit)*

p. 30

EXEMPLES DE PRATIQUE PROFESSIONNELLE

Activité-type 1 Développer une application sécurisée

Exemple n°1 ▶ *Installation et configuration de l'environnement pour le projet de formation*

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

Dans le cadre de ma formation, je travaille sur mon ordinateur personnel portable. En formation alors, je travaille dans un environnement Linux.

Cette formation étant entièrement à distance, il était primordial qu'on organise notre projet du meilleur moyen possible afin d'éviter d'éventuels problèmes.

Notre formateur était responsable de nous créer des dépôts Github à partir de l'organisation de notre école. Alors la première question pour nous était mono dépôt ou double dépôt ?

J'ai choisi un double dépôt car avec un double dépôt, il est plus facile de contrôler les accès et éviter des problèmes de sécurité.

Comme il était prédéterminé par notre formation d'utiliser Github pour l'hébergement de notre code, il était évident d'utiliser Git pour le versionnage de notre code pour le projet.

Pour l'environnement général, j'ai décidé d'utiliser :

- Typescript, car comme il inclut un typage statique, cela permettra une définition de types de données pour nos variables, fonctions, et objets.
- GraphQL, car il est flexible lorsqu'on fait des requêtes, et la combinaison de GraphQL et Typescript crée du code auto-documenté
- Apollo Studio, car il est facile d'utilisation et permet de tester les builds et la gestion des APIs avec GraphQL
- Jest; car il fonctionne bien avec Typescript et la mise en place de tests est très facile, avec assez peu de configuration nécessaire
- Cypress, car il est très facile à mettre en place, permet les tests end-to-end (E2E), Cypress recharge automatiquement après chaque changement de code, et les outils pour débbugger sont faciles à utiliser
- Docker, car il est facile à utiliser, et comme chaque membre de notre groupe travaillait sur un système différent, un système de containerisation comme LXC aurait été plus compliqué pour ceux qui ne travaillent pas sous Linux
- Dockerhub, car comme on utilise docker pour la containerisation, la plateforme pour héberger les images docker est très simple d'utilisation
- Visual Studio Code, car comme on travaillait à distance, avec 2 personnes à Bordeaux, 1 personne à Nantes et la dernière à Paris, VS Code rend facile le partage des lives lorsqu'on travaillait ensemble sur du code.

DOSSIER PROFESSIONNEL (DP)

NB : comme on travaillait en groupe pendant une période de 6 mois, j'ai créé un compte spécifique à notre groupe sur dockerhub, et ai pu partager l'utilisateur et mot de passe en privé pour que tout le monde puisse y avoir accès.

Pour le back-end, j'ai décidé d'utiliser :

- NodeJs, car il fonctionne bien avec Typescript/Javascript, l'écosystème de NPM (node package manager) offre un nombre de bibliothèques et frameworks imbattables, et il supporte une programmation asynchrone, ce qui est primordiale pour la performance et réactivité d'une application.
- ExpressJs, car c'est un framework minimaliste qui avec du middleware robuste pour facilement ajouter des fonctionnalités d'authentification
- TypeORM, car il permet de définir et gérer les relations entre entités facilement, avec des options de Eager ou Lazy Loading, et l'utilisation des décorateurs pour définir les entités rend le code plus facile à lire, ce qui aide avec la construction de la base de données
- TypeGraphQL, car utilise des décorateurs pour définir les types, queries, et mutations GraphQL, ce qui rend le code plus lisible, il génère automatiquement les schémas GraphQL basés sur les classes et décorateurs Typescript, évitant le besoin d'écrire des schémas à la main

Pour la base de données, j'ai choisi d'utiliser PostgreSQL, car il a des fonctionnalités avancées, il permet de définir des types de données, opérateurs, et fonctions personnalisables, et il supporte des fonctionnalités JSON, ce qui était important, comme un de nos attributs allait utiliser du JSON.

Pour le front-end, j'ai choisi d'utiliser :

- Next, un framework de React, qui s'appuie sur NodeJs, car il peut réduire la charge sur les navigateurs, il fournit une sécurité accrue, et il gère facilement le routage
- Material-UI : car c'est une bibliothèque basée sur du CSS, qui est facile d'utilisation et permet une personnalisation profonde grâce à une large gamme de paramètres de styles

Pour le versioning avec Git, j'ai voulu qu'on ait 3 branches : main, dev, et branche perso (avec le nom de cette branche étant la fonctionnalité sur laquelle on travaillait). De cette façon, avec chaque push or pull request, on peut tester les nouvelles fonctionnalités sur dev avant de mettre sur notre branche main.

2. Précisez les moyens utilisés :

Pour ce projet, j'avais besoin de :

- Une connexion internet fiable
- Un ordinateur (le mien est sous linux debian)
- Accès à discord pour la communication entre les membres de mon groupe et moi
- Visual Studio Code, qui est mon éditeur de code, mais aussi pour le partage des lives et je l'ai utilisé pendant la configuration du projet

DOSSIER PROFESSIONNEL (DP)

- Docker pour la containerisation du projet et afin de s'assurer que tout le groupe aurait les mêmes versions des outils lors du lancement de projet
- Un navigateur internet pour les tests du rendu (j'ai utilisé principalement Firefox)

3. Avec qui avez-vous travaillé ?

J'ai configuré mon système seule, mais le projet était en collaboration avec un groupe de trois autres personnes dans ma promo de formation en alternance.

4. Contexte

Nom de l'entreprise, organisme ou association	Wild Code School
Chantier, atelier, service	▶ Groupe Oros
Période d'exercice	▶ Du 04/2024 au 10/2024

5. Informations complémentaires (facultatif)

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)



DOSSIER PROFESSIONNEL ^(DP)

Activité-type 1 Développer une application sécurisée

Exemple n°2 ▶ Cliquez ici pour entrer l'intitulé de l'exemple

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

Cliquez ici pour taper du texte.

2. Précisez les moyens utilisés :

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

3. Avec qui avez-vous travaillé ?

Cliquez ici pour taper du texte.

4. Contexte

Nom de l'entreprise, organisme ou association

Cliquez ici pour taper du texte.

Chantier, atelier, service

▶ Cliquez ici pour taper du texte.

Période d'exercice

▶ **Du** Cliquez ici **au** Cliquez ici

5. Informations complémentaires *(facultatif)*

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

Activité-type 1 Développer une application sécurisée

Exemple n°3 ▶ Cliquez ici pour entrer l'intitulé de l'exemple

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

Cliquez ici pour taper du texte.

2. Précisez les moyens utilisés :

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

3. Avec qui avez-vous travaillé ?

Cliquez ici pour taper du texte.

4. Contexte

Nom de l'entreprise, organisme ou association Cliquez ici pour taper du texte.

Chantier, atelier, service Cliquez ici pour taper du texte.

Période d'exercice Du Cliquez ici au Cliquez ici

5. Informations complémentaires *(facultatif)*

Cliquez ici pour taper du texte.

Activité-type 2

Concevoir et développer une application sécurisée organisée en couches

Exemple n°1 ▶ *Conception et mise en place d'une base de données relationnelles pour le projet de formation*

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

Dans le cadre de ma formation en alternance, j'ai travaillé sur un projet d'avril jusqu'en octobre 2024, dans un rythme d'environ une semaine sur quatre, faisant partie d'un groupe de quatre personnes. Cette formation a été suivie entièrement à distance.

Lorsque nous avons commencé à travailler sur le projet, nous avons procédé ainsi pour la conception :

- Le formateur, qui avait créé un serveur discord privé pour notre classe, nous a attribué des canaux vocaux et écrits pour chaque groupe.
- Lors de la discussion en vocal, un de nos membres a créé un google drive commun, pour le partage des documents.
- Nous avons décidé de travailler sur un projet, où notre entreprise fictive voulait louer des biens pour du sport de montagne (skis, VTT, etc.).
- J'ai créé des phrases clés, et un utilisant ces phrases, cela nous a aidé à commencer à travailler sur la Merise.
- J'ai créé nos 5 Entités à partir de ces idées (Session, User, Reservation, Material, et Category). En faisant le Modèle Conceptuel de Données, ainsi qu'en ajoutant les cardinalités et les maximalités, j'ai constaté que nous allions avoir un lien Many-to-Many entre les Entités Reservation et Material, alors lorsque j'ai créé modèle logique de données, j'ai ajouté un tableau intermédiaire, qui devient notre Entité ReservedMaterial.
- Le MLD nous a permis de réfléchir aux attributs pour chaque Entité que nous allions créer, ainsi que voir où on aura des clés étrangères. L'Entité ReservedMaterial a, par exemple, 2 clés étrangères comme elle fait intermédiaire entre Reservation et Material.
- Le MPD nous a permis de décider quel type de donnée chaque attribut allait avoir. La plupart de nos attributs sont des VARCHARS, mais par exemple dans l'entité User, j'ai créé un attribut « role », qui est enum, avec les deux possibilités étant user ou admin. Cela est essentiel, car selon le rôle, l'utilisateur aura (ou non) l'accès au back-office.
- Après la Merise, nous avons travaillé sur plusieurs diagrammes de l'Unified Modeling Language. Comme pour la Merise, j'ai créé des phrases afin de nous aider à réfléchir comment créer notre diagramme de classe.
- Après avoir pris les classes que j'ai déjà créées pour la Merise, ainsi que leurs attributs, j'ai réfléchi à leurs multiplicités. Si nous prenons l'exemple de User, entre Session et User, la multiplicité est de 1, car un utilisateur ne peut ouvrir qu'une session à la fois. C'est la même chose pour User et Reservation, car une réservation est faite par un seul utilisateur.
- Après les multiplicités, j'ai pris en compte les portées. Il y a un seul attribut que j'ai mis en privé, c'est le mot de passe pour un utilisateur, pour les raisons de sécurité. Pour le reste des attributs, je les ai laissés en public.

DOSSIER PROFESSIONNEL (DP)

- J'ai également créé un diagramme de cas d'utilisation ainsi qu'un diagramme de séquence. Pour le diagramme de cas d'utilisation, j'ai clarifié surtout comment un utilisateur allait interagir avec notre application. C'est-à-dire qu'un simple visiteur peut parcourir notre catalogue, mais s'il veut faire une réservation, il faut créer un compte et se logger afin de pouvoir créer cette réservation. Un admin a des pouvoirs sur ce qui apparaît sur le site, il pourra créer, mettre à jour ainsi que supprimer des matériaux. Le diagramme de séquence, quant à lui, était concentré sur les interactions entre l'utilisateur loggé et notre site de location. Un utilisateur, quand il effectue une recherche, devrait recevoir les résultats correspondant à cette recherche. Optionnellement, il peut aussi voir la description d'un matériel qui lui intéresse et l'ajouter à son panier. Et s'il veut, il peut après procéder avec un achat.

Après la conception, j'ai travaillé sur la mise en place de la base de données sur notre projet :

- J'ai mis en place TypeORM afin d'établir les paramètres de connexion à la base de données dans le fichier de configuration, avec des détails comme le type de base de données (PostgreSQL), l'hôte, le port, l'utilisateur, le mot de passe, ainsi que les Entités établies lors de notre phase de conception.
- Pour les entités, chacune a été créée sous forme de classe, en utilisant des TypeORM pour indiquer les colonnes, les types de données et les relations entre elles, ainsi que des décorateurs TypeGraphQL pour rendre ces entités accessibles via l'API GraphQL. L'utilité de ces décorateurs rend facile la mise en place des tableaux pour la base de données. Par exemple, le décorateur @ObjectType permet de déclarer la classe comme un type GraphQL, alors que @PrimaryGeneratedColumn et @Column spécifient les colonnes de la base de données (@PrimaryGeneratedColumn étant normalement utilisé pour déclarer l'id pour nos classes). Un décorateur comme @OneToMany ou @ManyToOne permet d'indiquer la relation entre deux entités. Les types de données sont aussi indiqués (par exemple id: number, ou name: string).
- Les opérations CRUD ont été créées dans un dossier service, chaque entité ayant son fichier service attribué (comme material.service.ts, category.service.ts, etc). Comme j'ai prévu la structure des services lors de la création de notre diagramme de classe, la mise en place de notre CRUD a été assez simple.
- Enfin, j'ai créé un dossier resolvers, où comme pour les services, chaque entité a son fichier attribué. Les resolvers GraphQL sont utilisés pour les opérations côté serveur. Ils permettent de récupérer les entités de TypeORM pour les requêtes et de manipuler les données, Chaque resolver est lié à une opération spécifique.
- Une fois tout est mis en place, j'ai pu aussi installer Apollo Server, pour pouvoir tester la création, édition et suppression des données et s'assurer que notre base de données est fonctionnelle.

2. Précisez les moyens utilisés :

DOSSIER PROFESSIONNEL (DP)

J'ai utilisé Miro pour la création de nos modèles et diagrammes.

Pour coder, j'ai utilisé VS Code comme éditeur de code. Pour la base de données, j'utilise PostgreSQL. Pour la mise en place de la base de données, j'ai utilisé TypeORM, TypeGraphQL. GraphQL a été utilisé pour créer les requêtes, et Apollo Server a été utilisé pour tester ces requêtes.

3. Avec qui avez-vous travaillé ?

Ce projet était en collaboration avec trois autres personnes de ma formation en alternance.

4. Contexte

Nom de l'entreprise, organisme ou association	Wild Code School
Chantier, atelier, service	▶ Groupe Oros
Période d'exercice	▶ Du 04/2024 au 10/2024

DOSSIER PROFESSIONNEL (DP)

5. Informations complémentaires (*facultatif*)

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

Activité-type 2

Concevoir et développer une application sécurisée organisée en couches

Exemple n°2 ▶ Cliquez ici pour entrer l'intitulé de l'exemple

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

Cliquez ici pour taper du texte.

2. Précisez les moyens utilisés :

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

3. Avec qui avez-vous travaillé ?

Cliquez ici pour taper du texte.

4. Contexte

Nom de l'entreprise, organisme ou association Cliquez ici pour taper du texte.

Chantier, atelier, service Cliquez ici pour taper du texte.

Période d'exercice Du Cliquez ici au Cliquez ici

5. Informations complémentaires *(facultatif)*

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

Activité-type 2

Concevoir et développer une application sécurisée organisée en couches

Exemple n°3 ▶ Cliquez ici pour entrer l'intitulé de l'exemple

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

Cliquez ici pour taper du texte.

2. Précisez les moyens utilisés :

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

3. Avec qui avez-vous travaillé ?

Cliquez ici pour taper du texte.

4. Contexte

Nom de l'entreprise, organisme ou association Cliquez ici pour taper du texte.

Chantier, atelier, service Cliquez ici pour taper du texte.

Période d'exercice Du Cliquez ici au Cliquez ici

5. Informations complémentaires *(facultatif)*

Cliquez ici pour taper du texte.

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

Exemple n°1 ▶ *Mise en place et documentation du déploiement pour le projet de formation*

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

Dans le cadre de ma formation, mon école nous a fourni un VPS afin qu'on puisse faire un déploiement nous-mêmes. On était contraints à travailler sur le déploiement pendant nos jours de formation en alternance, car les jours où on était en entreprise, l'école mettait hors tension les machines virtuelles élèves de leur infrastructure.

Notre formateur nous a donné les accès nécessaires au serveur, et par la suite, j'ai pu me connecter à ce VPS par SSH (Secure SHell). J'ai ensuite installé et utilisé byobu afin de rester connectée au serveur, ainsi que mieux gérer les onglets différents dans le serveur lorsque je travaillais à faire le déploiement.

Voici les étapes que j'ai fait lorsque je me suis connectée au serveur pour la première fois :

- J'ai mis à jour le serveur
- J'ai changé le port de connexion de 22 à 2269 afin de ne pas avoir un port de défaut et mieux sécuriser notre serveur
- J'ai installé et configuré fail2ban afin d'avoir une protection supplémentaire, si quelqu'un essaie de se connecter à notre serveur sans succès
- J'ai installé et configuré Caddy (en mettant en place un Caddyfile) comme reverse proxy pour gérer les requêtes HTTP sur notre nom de domaine qui nous a été attribué
- J'ai installé docker sur le serveur afin de ne pas être obligée de faire l'installation de tous les logiciels nécessaires à notre application à la main
- J'ai créé un compte sur Dockerhub spécifique à notre projet, et puis 2 répos dessus, 1 pour le back, et 1 pour le front
- Pour chaque dépôt github, j'ai ajouté le username et le token de du dépôt dockerhub sous les secrets du dépôt
- J'ai créé 2 fichiers workflow (un pour le back, et un pour le front) où si les tests qui ont été mis en place passent, la nouvelle image devrait être poussé sur le dépôt attribué de dockerhub
- J'ai créé 2 fichiers docker-compose – un que j'ai appelé staging (ou pré-production), et l'autre prod (pour production)
- J'ai aussi créé un fichier nginx.conf pour rediriger les requêtes entrantes de /graphql vers notre back-end, et les autres requêtes sont envoyées au front-end
- J'ai créé un script bash simple pour qu'on puisse lancer en une commande un redémarrage des services docker compose avec les dernières images qui sont sur dockerhub.
- J'ai installé webhook sur le serveur, et créé webhook.conf, où j'ai créé deux webhooks, un pour mettre à jour notre staging, et l'autre pour production.

DOSSIER PROFESSIONNEL (DP)

- J'ai mis à jour le Caddyfile, pour avoir notre domaine principal (production), un sous-domaine pour staging, et un autre sous-domaine pour les webhooks (ops).
- Sur dockerhub, j'ai ajouté sous l'onglet webhooks, l'adresse domaine pour le webhook que j'ai créé pour le staging. Je n'ai pas ajouté le webook pour la production, car je voulais qu'on teste en staging avant de déclencher une mise à jour sur production.

Pour la documentation, j'ai rédigé un README en Markdown sur notre dépôt Github (sur le front), détaillant toutes les étapes que nous avons prises pour déployer notre projet. Je l'ai rédigé en anglais afin que ce soit le plus accessibles pour le plus grand nombre de personnes.

2. Précisez les moyens utilisés :

Cliquez ici pour taper du texte.

J'avais besoin de :

- Le VPS de l'école, pour le déploiement sur leur serveur
- Dockerhub et github, pour stocker les images, et github pour l'hébergement de notre code ainsi que les actions pour déclencher
- Visual Studio Code, pour l'écriture du code
- La documentation a été écrite dans VS Code et mis sur notre dépôt github sur le front

3. Avec qui avez-vous travaillé ?

DOSSIER PROFESSIONNEL (DP)

Ce projet était en collaboration avec trois autres personnes de ma formation en alternance.

4. Contexte

Nom de l'entreprise, organisme ou association	▶	Wild Code School
Chantier, atelier, service	▶	Groupe Oros
Période d'exercice	▶	Du 04/10 au 10/2024

5. Informations complémentaires (facultatif)

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

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

Exemple n°2 ▶ Cliquez ici pour entrer l'intitulé de l'exemple

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

Cliquez ici pour taper du texte.

2. Précisez les moyens utilisés :

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

3. Avec qui avez-vous travaillé ?

Cliquez ici pour taper du texte.

4. Contexte

Nom de l'entreprise, organisme ou association

Cliquez ici pour taper du texte.

Chantier, atelier, service

▶ Cliquez ici pour taper du texte.

Période d'exercice

▶ **Du** Cliquez ici **au** Cliquez ici

5. Informations complémentaires *(facultatif)*

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

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

Exemple n°3 ▶ Cliquez ici pour entrer l'intitulé de l'exemple

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

Cliquez ici pour taper du texte.

2. Précisez les moyens utilisés :

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

3. Avec qui avez-vous travaillé ?

Cliquez ici pour taper du texte.

4. Contexte

Nom de l'entreprise, organisme ou association

Cliquez ici pour taper du texte.

Chantier, atelier, service

▶ Cliquez ici pour taper du texte.

Période d'exercice

▶ **Du** Cliquez ici **au** Cliquez ici

5. Informations complémentaires *(facultatif)*

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL ^(DP)

Titres, diplômes, CQP, attestations de formation

(facultatif)

Intitulé	Autorité ou organisme	Date
Full-stack JavaScript Web Development	Ironhack – Web Development Bootcamp	Novembre 2022
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une 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] *Julia Spriggs* ,
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 *Talence*

le *14 Octobre 2024*

pour faire valoir ce que de droit.

Signature :



DOSSIER PROFESSIONNEL ^(DP)

ANNEXES

(Si le RC le prévoit)