

DOSSIER PROFESSIONNEL (DP)

Nom de naissance ▶ MARZAK
Nom d'usage ▶ MARZAK
Prénom ▶ Ali
Adresse ▶ 1 Rue des Chevesnes, 67540 Ostwald

Titre professionnel visé

Développeur web et web mobile niveau 5

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>

Sommaire

Exemples de pratique professionnelle

Développer la partie front-end d'une application web ou web mobile sécurisée p.

- ▶ Exercice de Formation ELAN -- Exercice Beergarden p. 5
- ▶ Exercice de Formation ELAN -- Projet SESSION p. 7
- ▶ Projet de Formation ELAN -- PROSER p. 9

Développer la partie back-end d'une application web ou web mobile sécurisée p.

- ▶ Exercice de Formation ELAN -- Projet Forum..... p. 11
- ▶ Projet de Stage -- Garage LCM p. 13
- ▶ Projet de Formation ELAN -- PROSER p. 15

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

Développer la partie front-end d'une application web ou web mobile sécurisée

Exemple n°1 ► Exercice de Formation ELAN -- Exercice Beergarden

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

L'exercice Beergarden est un projet de formation axé sur l'intégration d'une landing page à partir d'un cahier des charges fourni par les formateurs. Pour ce qui est des tâches effectuées, j'ai :

- ✓ Prise en main du cahier des charges contenant principalement le maquetage.
- ✓ Création de la structure HTML : 'header', 'main' -> 'sections', 'footer'.
- ✓ Respect des standards de la W3C
- ✓ Mise en place du design responsive : Utilisation de media Queries pour adapter selon différentes résolutions + Utilisation des unités flexibles (% , vh, vw). (Voir **Annexe 1.4**)
- ✓ Utilisation de Flexbox et Grid pour organiser les éléments de manière flexible et ordonnées. (Voir **Annexe 1.1**)
- ✓ Intégration du Menu burger pour la version mobile avec des interactions CSS & JS pour l'ouvrir et fermer. (Voir **Annexe 1.2**)
- ✓ Mise en place d'un effet Parallax sur une section du site donnant un dynamisme visuel lors du défilement. (Voir **Annexe 1.3**)
- ✓ Création d'une galerie d'images responsive + ajout d'animation lors de survol d'image.
- ✓ Affichage d'une Map avec l'ajout d'une balise « iframe ». (Voir **Annexe 1.3**)

2. Précisez les moyens utilisés :

- En suivant attentivement le maquetage fourni par les formateurs, j'ai structuré les pages en utilisant HTML5 et appliqué la mise en forme graphique ainsi que les animations avec CSS3. Pour améliorer le dynamisme du site, j'ai utilisé du JavaScript.
- Editeur de code : VSCode
- Hébergement & Contrôle de Version : Github (Git)

DOSSIER PROFESSIONNEL (DP)

3. Avec qui avez-vous travaillé ?

J'ai réalisé cet exercice de formation en autonomie, sous la supervision de mes formateurs.

4. Contexte

Nom de l'entreprise, organisme ou association		ELAN Formation
Chantier, atelier, service		▶ Elan Formation DWWM
Période d'exercice	▶ Du	15/03/2024 au 19/03/2024

5. Informations complémentaires (facultatif)

Texte

Activité-type 1

Développer la partie front-end d'une application web ou web mobile sécurisée

Exemple n°2 ► Exercice de Formation ELAN -- **Projet SESSION**

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

Le projet Session est une application permettant de gérer des sessions organisées par un formateur et pouvant accueillir des stagiaires. Pour ce qui est des tâches effectuées, j'ai :

- ✓ Prise en main du cahier des charges contenant les grands points à prendre en compte.
- ✓ Définit les **différentes étapes du projet** en créant et organisant les tâches sur Trello
- ✓ Conception du maquetage avec **Figma** : réalisation des **MockUps** en utilisant l'approche **Mobile First**, en commençant par la version mobile pour optimiser **le référencement naturel (SEO)**. (Voir **Annexe 2.1 et 2.2**)
- ✓ Mise en place du design responsive : Utilisation **des Media Queries** pour adapter à tous types de résolutions d'écrans, en combinant **des Unités flexibles** (% , vh, vw) pour assurer une mise en page fluide. (Voir **Annexe 2.3**)
- ✓ Développement des **templates front-end en utilisant Twig**, le moteur de templates de Symfony, pour générer dynamiquement les pages HTML de manière efficace et modulaire.
- ✓ Application de **transitions** sur les éléments interactifs (boutons, liens, images) pour améliorer **l'expérience utilisateur (UX)**.
- ✓ Intégration des formulaires d'inscription et Connexion et **stylisations avec du CSS**. (Voir **Annexe 2.5**)
- ✓ Mise en place de la mise en page du site en utilisant les propriétés **Flexbox** pour garantir une mise en page harmonieuse et responsive. (Voir **Annexe 2.3**)
- ✓ Respect des standards de la **W3C**
- ✓ Utilisation de **FormType** incluant le Filtrage et la validation des entrées.

2. Précisez les moyens utilisés :

DOSSIER PROFESSIONNEL (DP)

- ▶ Gestion de Projet : Trello
- ▶ Outil de design : Figma
- ▶ Editeur de code : VSCode
- ▶ Environnement de développement : Laragon
- ▶ Langages : HTML5, CSS3, JavaScript
- ▶ Moteur de template : Twig
- ▶ Librairie d'icone : RemixIcon

3. Avec qui avez-vous travaillé ?

Exercice réalisé dans le cadre de la formation sous la supervision des Formateurs.

4. Contexte

Nom de l'entreprise, organisme ou association ▶ **ELAN Formation**

Chantier, atelier, service ▶ **Elan Formation DWWM**

Période d'exercice ▶ **Du 16/05/2024 au 29/05/2024**

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Texte

Activité-type 1

Développer la partie front-end d'une application web ou web mobile sécurisée

Exemple n°3 ► Projet de Formation ELAN -- PROSER

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

Mon Projet « ProSer » est une application de multiservice permettant de mettre en contact des clients et des Entrepreneurs. Pour ce qui est des tâches effectuées, j'ai :

- ✓ Définit les **différentes étapes du projet** en créant et organisant les tâches sur Trello
- ✓ Mis en place la conception du Maquettage : réalisation des **Wireframe** en utilisant l'approche **Mobile First**, en commençant par la version mobile pour optimiser le **référencement naturel (SEO)**. (Voir **Annexe 3.1 et 3.2**)
- ✓ Mis en place de bonne pratique pour la gestion des images permettant un meilleur référencement (SEO) : Utilisation de l'**attribut « alt »**, privilégie les images de **format « webp »**, utilisation de **balise « picture »** pour l'utilisation de bonne image selon les fonctionnalités de l'appareil et l'utilisation de l'**attribut « loading »** pour un meilleur chargement des images.
- ✓ Utilisation de **React JS** pour un front **plus interactif et plus dynamique** avec l'utilisation du langage JavaScript. Meilleure organisation du code avec un système de « **Component** » **réutilisable**.
- ✓ Utilisation de « **Tailwind CSS** » pour une **Interface Utilisateur (UI)** beaucoup plus agréable, une application optimisée avec les classes utilitaires CSS et un **responsive design** beaucoup plus simple et efficace à mettre en place.
- ✓ La combinaison de React JS et Tailwind CSS améliore considérablement l'**expérience utilisateur (UX)** et permet une meilleure maintenance. (Voir **Annexe 3.3**)
- ✓ Mise en place de la mise en page du site en utilisant les propriétés **Flexbox** pour garantir une mise en page harmonieuse et responsive. (Voir **Annexe 3.3**)
- ✓ Utilisation de la **technique AJAX** consistant à envoyer des **requêtes HTTP asynchrones** pour pouvoir communiquer avec notre base de données et pouvoir mettre à jour nos données sans avoir à recharger la page améliorant ainsi l'expérience utilisateur (UX). (Voir **Annexe 3.4**)

2. Précisez les moyens utilisés :

- Gestion de Projet : Trello
- Hébergement & Contrôle de Version : Github (Git)
- Outil de design : Figma
- Editeur de code : VSCode
- Environnement de développement : Laragon
- Langages : HTML5, CSS3, JavaScript
- Moteur de template : Twig
- Bibliothèque UI (Interface Utilisateur) : **React Js**
- Framework CSS : **Tailwind CSS**
- Librairie d'icone : FontAwesome

DOSSIER PROFESSIONNEL (DP)

3. Avec qui avez-vous travaillé ?

Ce projet de fin de formation a été réalisé en autonomie avec le soutien des formateurs du centre de formation.

4. Contexte

Nom de l'entreprise, organisme ou association ► **ELAN Formation**

Chantier, atelier, service ► **Elan Formation DWWM**

Période d'exercice ► Du **08/06/2024** au **18/10/2024**

5. Informations complémentaires (facultatif)

Texte

Activité-type

2

Développer la partie back-end d'une application web ou web mobile sécurisée

Exemple n°1 ▶

Exercice de Formation ELAN – Projet Forum

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

Le projet Forum est une application responsive permettant d'ajouter un topic dans un des catégories et de poster un message dans ce topic. Possédant un système d'inscription et d'authentification sécurisé avec une hiérarchisation selon le rôle de l'utilisateur.

Pour ce qui est des taches effectuées, j'ai :

- ✓ Conception du **MCD** (Modèle conceptuel de données) et du **MLD** (Modèle logique de données) en suivant la méthode Merise et en utilisant l'outil Looping. (Voir **Annexe 4.1 et 4.2**)
- ✓ Pris en main de l'**architecture MVC** fourni par Elan Formation.
- ✓ Création de la base de données dans Laragon avec les différentes entités contenant ces attributs et mise en place des **cardinalités** en ajoutant les clés étrangères.
- ✓ Ajout des entités, de leurs fichiers Managers et des contrôleurs dans le modèle MVC.
- ✓ Création de **requête SQL** avec une **préparation et exécution** des requêtes. (Voir **Annexe 4.3**)
- ✓ Mise en place d'un **CRUD** pour l'ajout de topic et post, la modification de post et la suppression de d'un topic.
- ✓ Ajout des **alertes** avec des messages de Succès, Warning et d'Erreur.
- ✓ Gestion des **rôles** (admin, modérateur, user) pour bloquer certains accès.
- ✓ Mise en place de la **protection des injections SQL** avec les requêtes préparées et de la **faille de sécurité XSS** avec des filtrages des entrées avant l'entrée dans la base de données. (Voir **Annexe 4.4**)
- ✓ Mise en place **des vérifications lors de l'inscription** : Email et pseudo **unique**, les 2 mots de passe identiques + **Hachage du mot de passe** en utilisant l'algorithme par défaut de PHP.
- ✓ Mise en place **des vérifications lors de la connexion** : Vérifie l'existence du mail, hachage du mot de passe et vérification avec celle dans la base de données. (Voir **Annexe 4.5**)

2. Précisez les moyens utilisés :

- ▶ Gestion de Projet : Trello
- ▶ Modélisation de base de données : Looping
- ▶ Hébergement & Contrôle de Version : Github (Git)
- ▶ Editeur de code : VSCode
- ▶ Environnement de développement : Laragon
- ▶ Langages : PHP
- ▶ Architecture: Modèle MVC "Elan" (Model-View-Controller)
- ▶ Gestion de bases de données : MySQL

DOSSIER PROFESSIONNEL (DP)

3. Avec qui avez-vous travaillé ?

Exercice réalisé dans le cadre de la formation sous la supervision des Formateurs.

4. Contexte

Nom de l'entreprise, organisme ou association ► **ELAN Formation**

Chantier, atelier, service ► Elan Formation DWWM

Période d'exercice ► Du 16/05/2024 au 29/05/2024

5. Informations complémentaires (facultatif)

Texte

Activité-type 2

Développer la partie back-end d'une application web ou web mobile sécurisée

Exemple n°2 ► Projet de Stage – Garage LCM

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

Dans le cadre de mon Stage, j'ai eu l'opportunité de réaliser un site Vitrine pour le Garage LCM. Pour ce qui est des tâches effectuées, j'ai :

- ✓ Conception du **MCD** (Modèle conceptuel de données) et du **MLD** (Modèle logique de données) en suivant la méthode Merise et en utilisant l'outil Looping.
- ✓ Mise en place de l'environnement **Symfony** avec le **design Pattern MVC** (Model – View-Controller)
- ✓ Création et Gestion de la base de données (BDD) avec **Doctrine** : Conception de la BDD en **MySQL**, en créant les entités et en définissant les **relations entre elles (Cardinalité)** selon le MLD. Utilisation de **Doctrine ORM** pour la **persistance** des données avec des **migrations** pour versionner les schémas de base de données.
- ✓ Utilisation de **requête DQL personnalisées** pour communiquer avec la BDD.
- ✓ Mise en place d'un système d'Inscription et d'authentification pour l'administrateur du site.
- ✓ Mise en place d'un espace d'administration avec le bundle **EasyAdmin** pour gérer le contenu du Site en effectuant des opérations **CRUD** (Create, Read, Update, Delete) et gérer les formulaires reçus. (Voir **Annexe 5.2**)
- ✓ Mise en place de plusieurs formulaires d'entrée avec la mise en place de Validation côté Serveur en appliquant des contraintes « Email, String, etc... ». (Voir **Annexe 5.1**)
- ✓ Gestion des **rôles** (admin, user) pour bloquer certains accès tel que le Panel Administration
- ✓ Mise en place de la **protection des injections SQL** avec les requêtes préparées et de la **faille de sécurité XSS** avec des filtres des entrées avant l'entrée dans la base de données. (Voir **Annexe 5.1**)
- ✓ Mise en place de la protection contre la **faille CSRF** via Symfony par le biais de formType.
- ✓ Mise en place de la protection de la **faille par Force brute** avec le composant Rate Limiter.

2. Précisez les moyens utilisés :

- Gestion de Projet : Trello
- Modélisation de base de données : Looping
- Hébergement & Contrôle de Version : Github (Git)
- Editeur de code : VSCode
- Environnement de développement : Laragon
- Langages : PHP
- Framework : **Symfony**
- Architecture: Modèle MVC (Model-View-Controller)
- Moteur de template : Twig
- Gestion de bases de données : MySQL

DOSSIER PROFESSIONNEL (DP)

3. Avec qui avez-vous travaillé ?

Ce projet de stage a été effectué en autonomie.

4. Contexte

Nom de l'entreprise, organisme ou association ► **Garage LCM**

Chantier, atelier, service ► **Garage LCM**

Période d'exercice ► Du **03/06/2024** au **19/07/2024**

5. Informations complémentaires (facultatif)

Texte

Activité-type 2

Développer la partie back-end d'une application web ou web mobile sécurisée

Exemple n°3 ► Projet de Formation ELAN -- PROSER

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

Mon Projet « ProSer » est une application de multiservice permettant de mettre en contact des clients et des Entrepreneurs. Pour ce qui est des tâches effectuées, j'ai :

- ✓ Gestion de projet : Utilisation de la méthode Scrum et de la méthode MoSCoW
- ✓ Conception du **MCD** (Modèle conceptuel de données) et du **MLD** (Modèle logique de données) en suivant la méthode Merise et en utilisant l'outil Looping.
- ✓ Création et Gestion de la base de données (BDD) avec **Doctrine** : Conception de la BDD en **MySQL**, en créant les entités et en définissant les **relations entre elles (Cardinalité)** selon le MLD. Utilisation de **Doctrine ORM** pour la **persistance** des données avec des **migrations** pour versionner les schémas de base de données.
- ✓ Utilisation de **requête DQL personnalisées** pour communiquer avec la BDD. (Voir **Annexe 6.1**)
- ✓ Mise en place d'un système d'Inscription et d'authentification.
- ✓ Gestion des **rôles** (« ADMIN », « ROLE_PRO », « ROLE_USER ») pour bloquer certains accès et ajout d'un formulaire pour obtenir le « ROLE_PRO » correspondant à l'Entrepreneur.
- ✓ Mise en place de la **protection des injections SQL** avec les requêtes préparées et de la **faille de sécurité XSS** avec des filtres des entrées avant l'entrée dans la base de données.
- ✓ Mise en place de la protection contre la **faille CSRF** via un contrôleur Symfony et une requête AJAX dans le composant React contenant le formulaire.
- ✓ Mise en place de la protection de la **faille par Force brute** avec le composant **Rate Limiter** avec l'utilisation d'**Event Listener** pour une protection personnalisée.
- ✓ Mise en place **des vérifications lors de l'inscription** : Email **unique**, les 2 mots de passe identiques + **Hachage du mot de passe** en utilisant l'algorithme proposé par Symfony 7.1.
- ✓ Mise en place **des vérifications lors de la connexion** : Vérifie l'existence du mail, hachage du mot de passe et vérification avec celle dans la base de données.
- ✓ Mise en place d'une **Messagerie** entre le client et l'entrepreneur permettant aussi à l'entrepreneur d'envoyer un rendez-vous pour qu'il soit complété par le client.
- ✓ Mise en place d'une gestion **de Rendez-vous** pour l'affichage d'un rendez-vous et de sa suppression si voulue par l'entrepreneur.
- ✓ Intégration d'une **plateforme de paiement « Stripe »** directement dans l'application. (Voir **Annexe 6.5**)
- ✓ Mise en place d'un **Dashboard** via React pour le client et l'entrepreneur permettant de **gérer leur annonce (CRUD)** et d'afficher les rendez-vous et messages.
- ✓ Mise en place d'un **Espace d'administration** avec le bundle **EasyAdmin** pour gérer le contenu du Site en effectuant des opérations **CRUD** (Create, Read, Update, Delete) et gérer les formulaires reçus.
- ✓ Respect du **RGPD** : **minimisation des données** et la possibilité à l'utilisateur : de récupérer ses données personnelles, de les modifier, de supprimer son compte temporairement (30 jours) ou définitivement avec une **anonymisation**. (Voir **Annexe 6.2 et 6.3**)
- ✓ **Envoi de mail** pour des confirmations tel que la suppression de compte, la modification de mot de passe ou encore l'activation de compte lors de l'inscription. (Voir **Annexe 6.4**)
- ✓ Ajout **des alertes** avec affichage des messages de Succès, Warning et d'Erreur.

DOSSIER PROFESSIONNEL (DP)

2. Précisez les moyens utilisés :

- ▶ Gestion de Projet : Trello
- ▶ Modélisation de base de données : Looping
- ▶ Hébergement & Contrôle de Version : Github (Git)
- ▶ Editeur de code : VSCode
- ▶ Environnement de développement : Laragon
- ▶ Langages : PHP
- ▶ Framework : **Symfony**
- ▶ Architecture: Modèle MVC (Model-View-Controller)
- ▶ Moteur de template : Twig
- ▶ Gestion de bases de données : MySQL

3. Avec qui avez-vous travaillé ?

Ce projet de fin de formation a été réalisé en autonomie avec le soutien des formateurs du centre de formation.

4. Contexte

Nom de l'entreprise, organisme ou association ▶ **ELAN Formation**

Chantier, atelier, service ▶ **ELAN Formation DWWM**

Période d'exercice ▶ Du **08/06/2024** au **18/10/2024**

DOSSIER PROFESSIONNEL (DP)

5. Informations complémentaires *(facultatif)*

Texte

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.
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 *MARZAK Ali*,
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 *Strasbourg* le *18/10/2024*

pour faire valoir ce que de droit.

Signature :



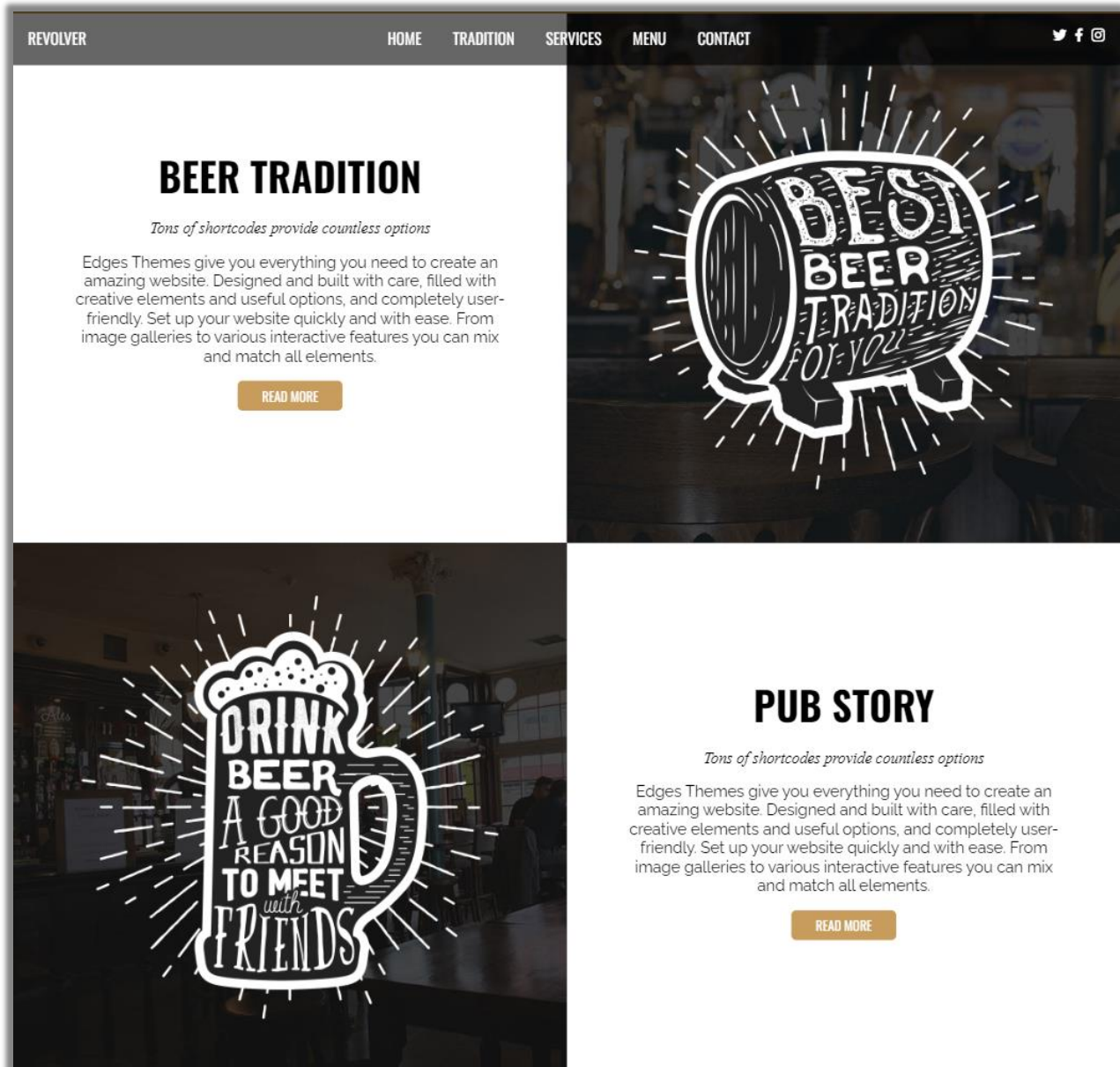
DOSSIER PROFESSIONNEL (DP)

Documents illustrant la pratique professionnelle

(facultatif)

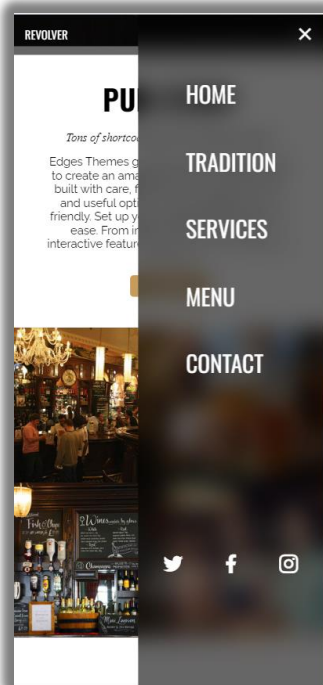
Intitulé
Cliquez ici pour taper du texte.

ANNEXES

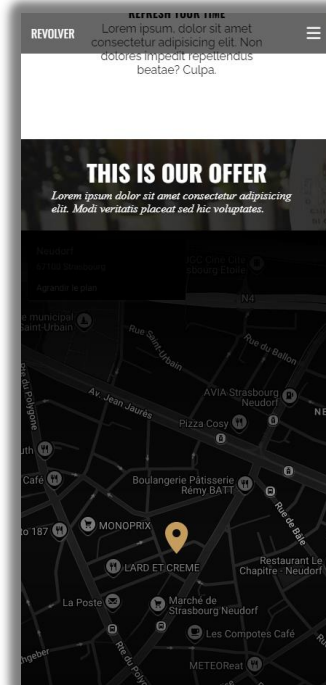


Annexe 1.1 Affichage Desktop Beergarden

DOSSIER PROFESSIONNEL (DP)



Annexe 1.2 Menu Burger NavBar



Annexe 1.3 Effet Parallaxe & Map

```
354 @media screen and (max-width:1024px) {
355
356     /*-----TRADITION SECTION-----*/
357     .tradition_section{
358         height: auto;
359     }
360
361     .container_tradition{
362         width: 100%;
363         display: grid;
364         grid-template-columns: 100vw;
365         grid-template-rows: repeat(4, 1fr);
366     }
367
368     /*-----GALLERY SECTION-----*/
369     .container_gallery{
370         display: grid;
371         grid-template-rows: 1fr 1fr;
372         grid-template-columns: 1fr 1fr;
373     }
374
375     /*-----SERVICE SECTION-----*/
376     .container_service{
377         flex-direction: column;
378         padding: 20%;
379         gap: 100px;
380     }
381 }
```

Annexe 1.4 Responsive Design (Max : 1024px)

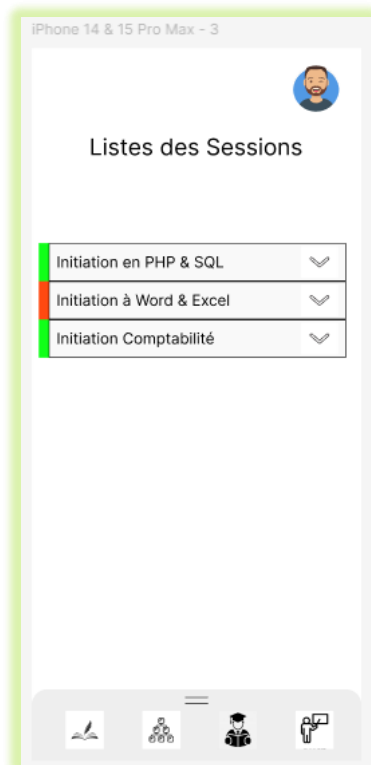


Figure 2.1 Wireframe Mobile



Figure 2.2 Wireframe Tablette / Desktop

```

729  /* ----- Media screen ----- */
730  /* ----- */
731
732  @media screen and (min-width:1025px){
733      .wrapper{
734          display: flex;
735          justify-content: space-between;
736      }
737
738      header{
739          position: unset;
740          margin: 100px 40px;
741          width: unset;
742          border-radius: 30px;
743          height: 500px;
744      }
745
746      .nav_hover_desktop{
747          display: none;
748      }
749
750      .desktop{
751          display: none;
752      }
753
754      .nav_container{
755          flex-direction: column;
756          height: 400px;
757      }
758
759      main{
760          width: 80vw;
761          padding: 10vh 0;
762      }
763  }
    
```

Figure 2.3 Responsive Design (Max : 1025px)

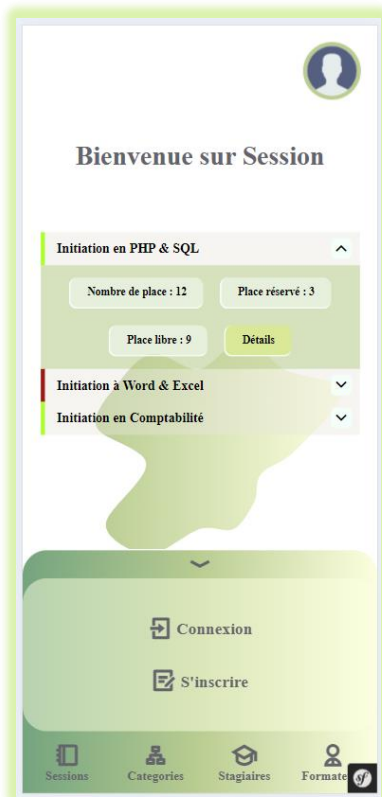


Figure 2.4 Vue Session Mobile



Figure 2.5 Formulaire d'Inscription

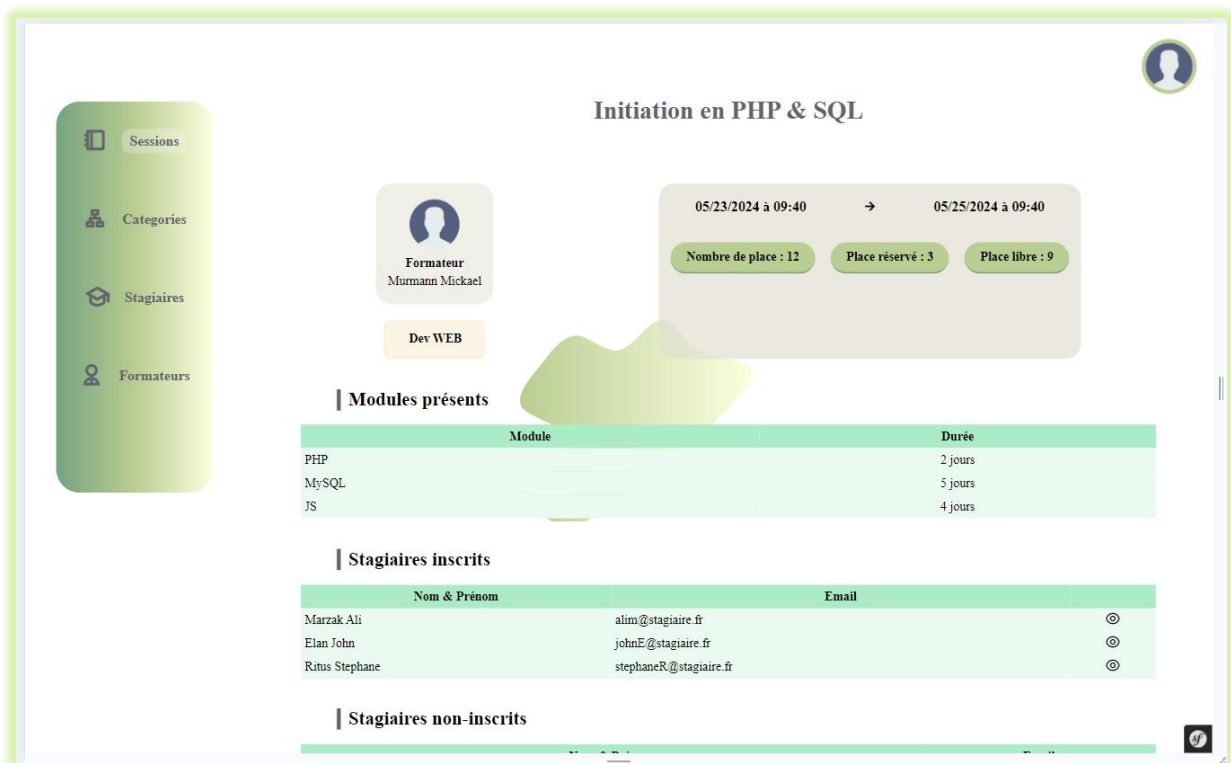


Figure 1.6 Vue de la Session Desktop

DOSSIER PROFESSIONNEL (DP)

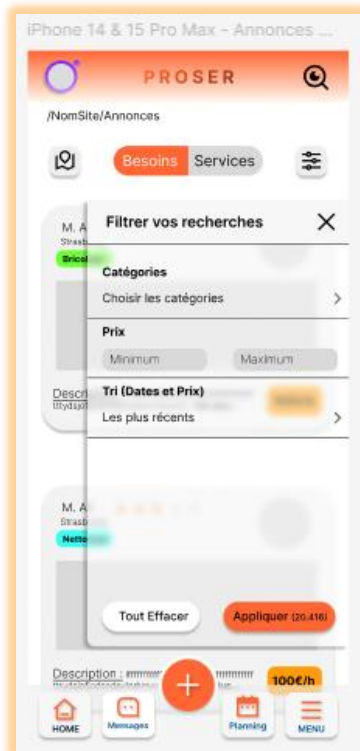


Figure 3.1 Wireframe Version Mobile

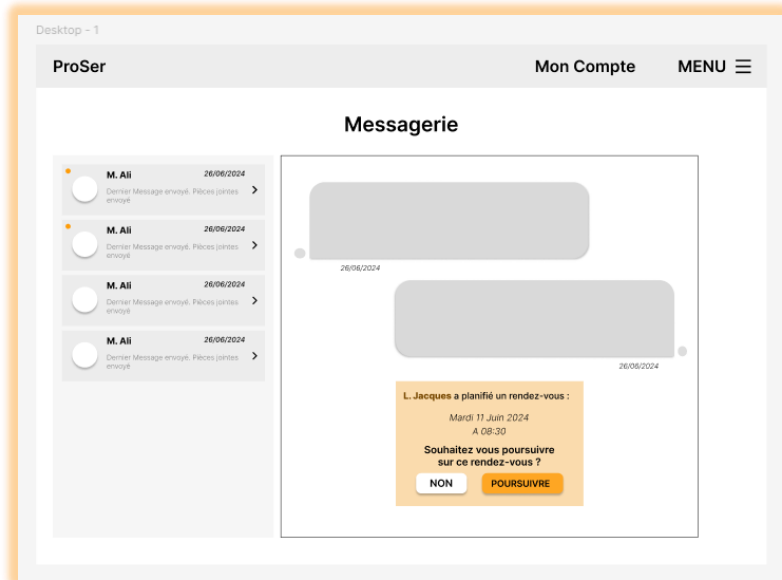


Figure 3.2 Wireframe Version Tablet / Desktop

```
1 import React from 'react'; 6.9k (gzipped: 2.7k)
2
3 export default function CardAdvert({ advert, classTailwind }) {
4   return (
5     <React.Fragment>
6       <a href={` /ficheAdvert/${advert.slug}`} className={`flex flex-col w-full items-center`>
7         <div className='w-full flex justify-between text-sm'>
8           <div className='flex flex-col gap-2'>
9             <div className='flex flex-col'>
10              <span className='text-lg font-semibold'>{advert.user.lastname}</span> {advert
11              <span>{advert.city}</span>
12            </div>

```

Figure 3.3 React.js avec Tailwind CSS

```
10 // Fonction Asynchrone récupérant les annonces actives
11 const fetchAdvertActive = async () => {
12   try {
13     // Reponse suite à une requête AJAX
14     const response = await fetch('/advertActive');
15     if (!response.ok) {
16       throw new Error('Network response was not ok');
17     }
18     const data = await response.json();
19     setAdvertsActive(data);
20   } catch (error) {
21     console.error('Erreur lors de la récupération des annonces actives', error);
22   }
23 };

```

Figure 3.4 Fonction contenant une requête AJAX via L'API FETCH

DOSSIER PROFESSIONNEL (DP)

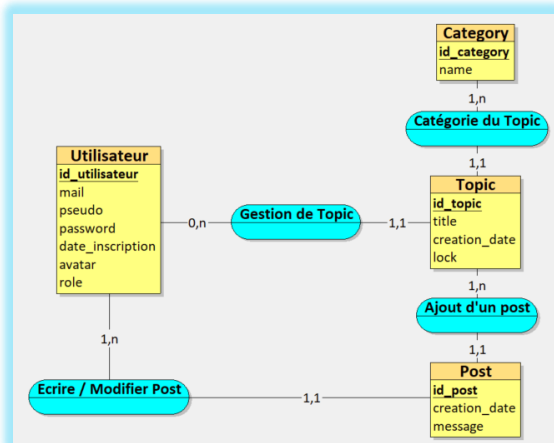


Figure 4.1 MCD

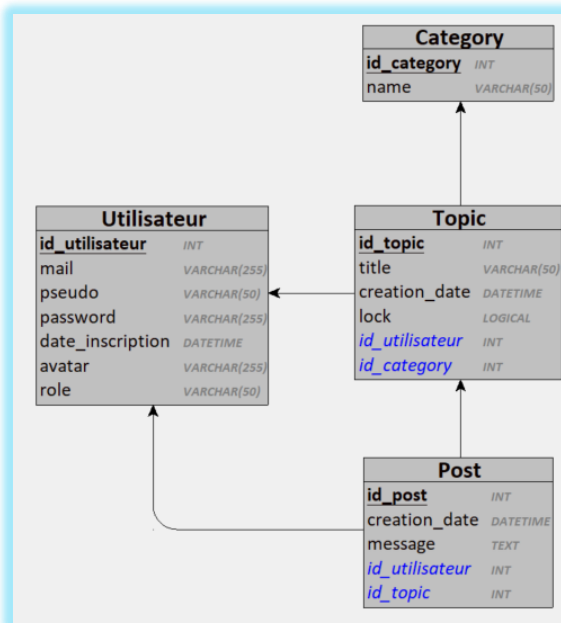


Figure 4.2 MLD

```

18 // récupérer tous les topics d'une catégorie spécifique (par son id)
19 public function findTopicsByCategory($id) {
20
21     $sql = "SELECT *
22           FROM ".$this->tableName." t
23           WHERE t.category_id = :id
24           ORDER BY t.creationDate DESC";
25     // la requête renvoie plusieurs enregistrements --> getMultipleResults
26     return $this->getMultipleResults(
27         DAO::select($sql, ['id' => $id]),
28         $this->className
29     );
30 }
  
```

Figure 4.3 Requête SQL dans le Topic Manager

```

19 // Filtrage de la saisie des champs du formulaire
20 $mail = filter_input(INPUT_POST, "mail", FILTER_SANITIZE_FULL_SPECIAL_CHARS, FILTER_VALIDATE_EMAIL);
21 $pseudo = filter_input(INPUT_POST, "pseudo", FILTER_SANITIZE_FULL_SPECIAL_CHARS);
22 $f_password = filter_input(INPUT_POST, "f_password", FILTER_SANITIZE_FULL_SPECIAL_CHARS);
23 $s_password = filter_input(INPUT_POST, "s_password", FILTER_SANITIZE_FULL_SPECIAL_CHARS);
  
```

Figure 4.4 Filtrage des Entrées : Protection contre la faille XSS

```

125 // Verifier si la saisie est correcte
126 if($mail && $password)
127 {
128     $user = $userManager->listUserByMail($mail);
129     // Verifie si le mail existe
130     if($user)
131     {
132         $hash = $user->getPassword();
133         // Verifie si le mot de passe est correcte
134         if(password_verify($password, $hash))
135         {
136             Session::setUser($user);
137         }
138     }
139 }
  
```

Figure 4.5 Vérification des saisies de Connexion

```
18 public function buildForm(FormBuilderInterface $builder, array $options): void
19 {
20     $builder
21         ->add('immatriculation', TextType::class)
22         ->add('nom', TextType::class)
23         ->add('prenom', TextType::class)
24         ->add('description', TextareaType::class)
25         ->add('email', EmailType::class)
26         ->add('telephone', TextType::class)
27         ->add('sectionCarrosseries', EntityType::class, [
28             'class' => SectionCarrosserie::class,
29             'choice_label' => 'nomSection',
30             'multiple' => true,
31             'expanded' => false,
32             'attr' => ['class' => 'select2'],
33             'placeholder' => 'Sélectionnez des sections de carrosserie',
34             'required' => false,
35         ])
36     ;
37 }
```

Figure 5.1 FormType avec Type d'entrée

```
59 public function configureDashboard(): Dashboard
60 {
61     return Dashboard::new()
62         ->setTitle('Dashboard Garage LCM');
63 }
64
65 public function configureMenuItems(): iterable
66 {
67     yield MenuItem::linkToDashboard('Dashboard', 'fa fa-home');
68     yield MenuItem::linkToCrud('Mecanique', 'fa-solid fa-wrench', Mecanique::class);
69     yield MenuItem::linkToCrud('Carrosserie', 'fa-solid fa-car-rear', Carrosserie::class);
70     yield MenuItem::linkToCrud('Autres', 'fa-solid fa-info', Other::class);
71     yield MenuItem::section('Prestation');
72     yield MenuItem::linkToCrud('Categorie Prestation', 'fa-solid fa-list', CategoriePrestation::class);
73     yield MenuItem::linkToCrud('Prestation', 'fa-solid fa-hand-holding-dollar', Prestation::class);
74 }
75 }
```

Figure 5.2 Configuration du Menu du Dashboard

```

20 | // Fonction contenant une requête DQL pour les recherches d'annonces
21 | public function suggestionsSearch(string $searchInput): array
22 | {
23 |     $status = "active";
24 |     $query = $this->createQueryBuilder('a');
25 |     $query->where("a.status = :status");
26 |     $query->setParameter('status', $status);
27 |     if($searchInput){
28 |         $query->andWhere('MATCH_AGAINST(a.titleAd, a.description) AGAINST (:searchInput boolean)>0');
29 |         $query->setParameter('searchInput', '.*'. $searchInput .'*');
30 |     }
31 |     return $query->getQuery()->getResult();
32 | }

```

Figure 6.1 Requête DQL personnalisé

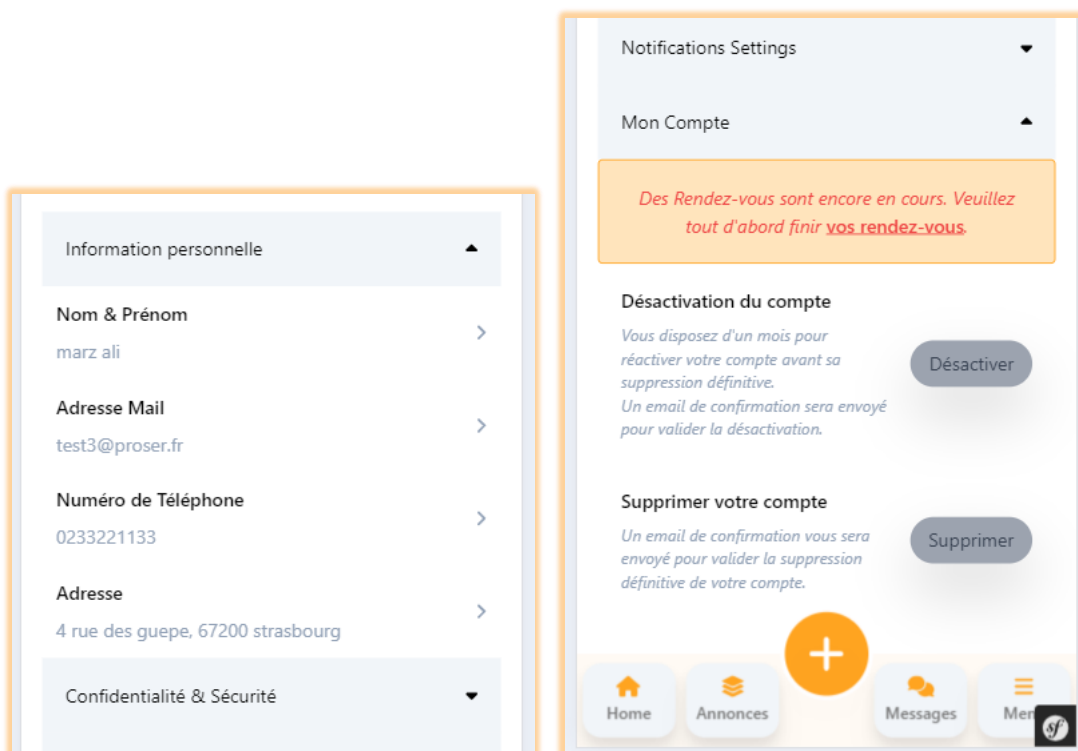


Figure 6.2 RGPD Droit de Rectification

Figure 6.3 RGPD Droit à l'oubli

```

68 | // Envoi mail de confirmation
69 | $this->emailVerifier->sendEmailConfirmation('app_verify_email', $user,
70 | (new TemplatedEmail())
71 |     ->from(new Address('support@proser.com', 'proser'))
72 |     ->to($user->getEmail())
73 |     ->subject('Confirmation pour la désactivation du compte')
74 |     ->htmlTemplate('account_management/deactivate_email.html.twig')
75 |     ->context([
76 |         'token' => $token,
77 |         'user' => $user,
78 |         'url' => $_ENV['APP_URL']
79 |     ])
80 | );

```

Figure 6.4 Envoi d'email de confirmation

```

44 | // Crée un nouvel Intent de Paiement via l'API Stripe
45 | public function createPaymentIntent(int $amount, string $email, string $id_client)
46 | {
47 |     return $this->stripe->paymentIntents->create([
48 |         'amount' => $amount * 100,
49 |         'currency' => 'eur',
50 |         'receipt_email' => $email,
51 |         'customer' => $id_client,
52 |         'description' => "Paiement d'une annonce PROSER",
53 |         'setup_future_usage' => 'off_session',
54 |         'automatic_payment_methods' => [
55 |             'enabled' => true,
56 |         ],
57 |     ]);
58 | }

```

Figure 6.5 Création d'une Instance de paiement