

Ecole IRIS Paris

BTS SIO SLAM Année 2023 – 2024

Documentation PPE

Client lourd Mojo

Sommaire

Introduction

Qu'est-ce qu'un PPE ?

- Définition et objectifs
- Importance dans le cursus scolaire ou professionnel

Mise en situation professionnelle

- Contexte
- professionnel Besoin
 de l'entreprises

Présentation du client léger

- Cahier des charges
 - Objectifs et exigences du projet
 - Fonctionnalités attendues
- Étude de cas utilisateur
 - Scénarios d'utilisation
 - Diagrammes des cas d'utilisation

Modèle Conceptuel de Données (MCD)

- Description du modèle
- Entités, attributs et relations
- Illustration graphique du MCD

Problèmes rencontrés et solutions

- Difficultés techniques
- Contraintes de développement

Conclusion

Bilan du projet
Enseignements tirés

Introduction

Dans le cadre de notre cursus en BTS Services Informatiques aux Organisations (SIO) option Solutions Logicielles et Applications Métiers (SLAM), nous avons eu l'opportunité de réaliser un Projet Personnel Encadré (PPE) d'envergure. Ce projet constitue une étape cruciale de notre formation, nous permettant d'appliquer les connaissances théoriques acquises en classe à des situations pratiques et concrètes.

Le présent document vise à présenter en détail notre PPE, qui représente une occasion unique de mettre en pratique nos compétences en développement logiciel et en gestion de projet. Nous examinerons les différentes étapes de conception, de développement et de déploiement de notre application, ainsi que les défis rencontrés et les solutions adoptées tout au long du processus.

Au fil de ce rapport, nous mettrons en lumière l'importance de la méthodologie de travail en équipe, de la communication efficace et de la gestion des délais, compétences essentielles dans le domaine de l'informatique. Nous détaillerons également les technologies utilisées, les choix architecturaux et les fonctionnalités implémentées, offrant ainsi un aperçu complet de notre travail.

Définition du PPE

Le Projet Personnel Encadré (PPE) est une étape incontournable dans le parcours de nombreux étudiants en formation professionnelle, notamment dans le domaine de l'informatique et du développement logiciel. Il constitue un projet de grande envergure réalisé en autonomie ou en équipe, sous la supervision d'un encadrant pédagogique ou professionnel.

1 Qu'est-ce qu'un PPE ?

Le PPE est une initiative visant à permettre aux étudiants de mettre en pratique les connaissances acquises tout au long de leur formation. Il s'agit d'un projet concret et significatif, souvent lié à leur domaine d'étude, qui leur offre l'opportunité de développer et de démontrer leurs compétences techniques, leur créativité et leur capacité à travailler en équipe.

2 Objectifs du PPE

Le principal objectif du PPE est de permettre aux étudiants de consolider et d'approfondir leurs connaissances théoriques en les appliquant à des situations réelles. En travaillant sur un projet concret, les étudiants sont confrontés à des défis réels du monde professionnel, ce qui les prépare efficacement à leur future carrière.

3 Importance du PPE

Le PPE revêt une importance capitale dans la formation des étudiants, car il leur permet de :

- Mettre en pratique les concepts théoriques appris en cours.
- Acquérir de l'expérience professionnelle en travaillant sur un projet réel.
- Développer des compétences techniques, méthodologiques et relationnelles essentielles pour leur future carrière.
- Démontrer leur capacité à mener à bien un projet de grande envergure, de la conception à la réalisation.
- Se familiariser avec les contraintes et les exigences du monde professionnel.

Mise en Situation Professionnelle

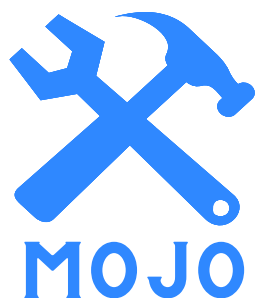
Contexte Professionnel

Mojo est une entreprise innovante évoluant dans le secteur des services en ligne. Forte de son expertise technique et de sa volonté d'offrir des solutions numériques de qualité, **Mojo** s'est positionnée comme un acteur majeur dans le domaine de la réservation de prestataires en ligne. Inspirée par le succès rencontré par des plateformes similaires, **Mojo** s'efforce de proposer une expérience utilisateur exceptionnelle à travers son propre site web de réservation de prestataires.

Besoin de l'Entreprise

Dans un contexte où les services en ligne connaissent une montée en puissance et où les utilisateurs recherchent des solutions pratiques et efficaces, votre application Java identifie un besoin crucial sur le marché : la nécessité d'une plateforme de gestion des prestataires de services. Cette application vise à offrir aux administrateurs la possibilité de gérer efficacement les données

Objectifs de l'application Mojo



1. **Interface Utilisateur Simple** : Créer une interface utilisateur intuitive et conviviale permettant aux administrateurs d'ajouter et de supprimer des données facilement, sans confusion ni complications inutiles.
2. **Fonctionnalité d'Ajout de Données** : Développer une fonctionnalité permettant aux administrateurs d'ajouter de nouvelles données à la base de données de l'application, en suivant un processus clair et efficace.
3. **Fonctionnalité de Suppression de Données** : Mettre en place une fonctionnalité de suppression pour permettre aux administrateurs de retirer des données de la base de données lorsque cela est nécessaire, en veillant à ce que cette action soit réversible si possible.

Cahier des Charges - Plateforme de Réservation de Prestataires à Domicile

1. Introduction

Le présent document constitue le cahier des charges pour le développement d'une application Java d'administration des prestataires à domicile, nommée "Mojo". Cette application vise à fournir une interface conviviale et efficace permettant aux administrateurs de gérer les données

2. Objectifs du Projet

- L'objectif principal du projet est de développer une application Java robuste et conviviale qui permettra aux administrateurs de gérer efficacement les données des prestataires et des clients, y compris leurs informations de profil, leurs services proposés, leurs tarifs, leurs disponibilités, etc.

3. Fonctionnalités Attendues

La plateforme Mojo devra inclure les fonctionnalités suivantes :

3.1 Gestion des Clients et des Prestataires

Système d'inscription pour les utilisateurs et les prestataires, avec vérification par e-mail.

3.2. Recherche et Réservation

- Interface de gestion permettant aux administrateurs d'ajouter, de modifier et de supprimer les informations des clients et des prestataires.
- Fonctionnalité d'ajout de nouveaux clients et prestataires, permettant de saisir leurs informations personnelles et de contact.
- Possibilité de modifier les informations existantes des clients et des prestataires, telles que les coordonnées, les services proposés, les tarifs, etc.
- Fonction de suppression des clients et des prestataires du système, avec confirmation pour éviter les suppressions accidentelles.

3.3. Gestion des Services

Interface de gestion permettant aux administrateurs d'ajouter, de modifier et de supprimer les services proposés par les prestataires

Étude de Cas Utilisateur Scénarios d'Utilisation:

L'administrateur se connecte à l'application d'administration de la plateforme *Mojo*.

Une fois connecté, il accède à la section de gestion des services, des prestataires ou des clients, selon ses besoins.

1. Pour ajouter un nouveau service :

- L'administrateur sélectionne l'option d'ajout de service.
- Il saisit les informations du nouveau service telles que le nom, la description et le tarif.
- Après vérification, il confirme l'ajout du service dans le système.

2. Pour supprimer un service existant :

- L'administrateur accède à la liste des services existants.
- Il sélectionne le service à supprimer.
- Après confirmation, le service est retiré du système.

3. Pour ajouter un nouveau prestataire :

- L'administrateur sélectionne l'option d'ajout de prestataire.
- Il saisit les informations du nouveau prestataire telles que le nom, les coordonnées et les services proposés.
- Après vérification, il confirme l'ajout du prestataire dans le système.

4. Pour supprimer un prestataire existant :

- L'administrateur accède à la liste des prestataires existants.
- Il sélectionne le prestataire à supprimer.
- Après confirmation, le prestataire est retiré du système.

5. Pour ajouter un nouveau client :

- L'administrateur sélectionne l'option d'ajout de client.
- Il saisit les informations du nouveau client telles que le nom, les coordonnées, etc.
- Après vérification, il confirme l'ajout du client dans le système.

6. Pour supprimer un client existant :

- L'administrateur accède à la liste des clients existants.
- Il sélectionne le client à supprimer.
- Après confirmation, le client est retiré du système.

7. Une fois toutes les modifications effectuées, l'administrateur peut vérifier les mises à jour dans le système et confirmer que les services, prestataires et clients ont été correctement ajoutés, modifiés ou supprimés.

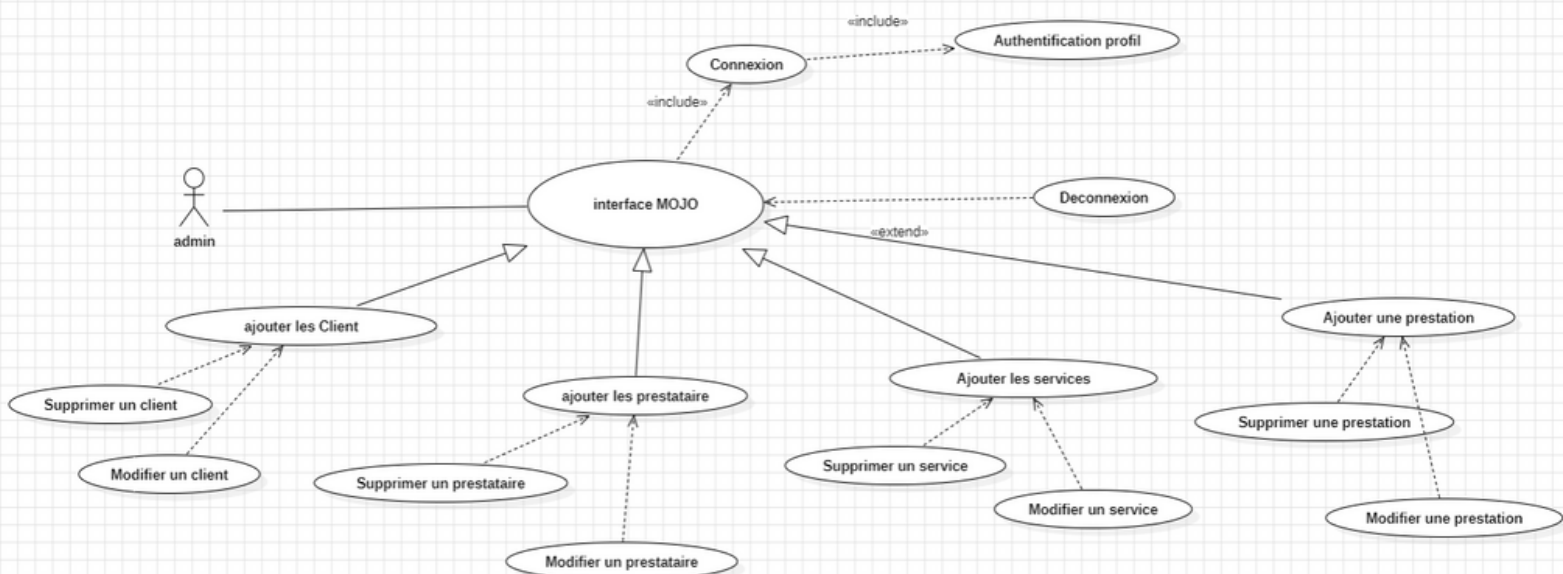
Diagrammes des Cas d'Utilisation:

Ce diagramme présente les différentes actions que peuvent effectuer les admins l'application **Mojo**.

Les principaux cas d'utilisation incluent :

- **Se Connecter** : L'administrateur se connecte à l'application d'administration de la plateforme **Mojo**.
- **Gérer Services, Prestataires ou Clients** : Une fois connecté, l'administrateur accède à la section de gestion des services, des prestataires ou des clients en fonction de ses besoins.
- **Ajouter Service, Prestataire ou Client** : L'administrateur peut choisir d'ajouter un nouveau service, un nouveau prestataire ou un nouveau client en sélectionnant l'option correspondante et en saisissant les informations nécessaires.
- **Supprimer Service, Prestataire ou Client** : De même, l'administrateur peut supprimer un service existant, un prestataire existant ou un client existant en sélectionnant l'option correspondante et en choisissant l'élément à supprimer.

Diagrammes des Cas d'Utilisation:



Modèle Conceptuel de Données (MCD)

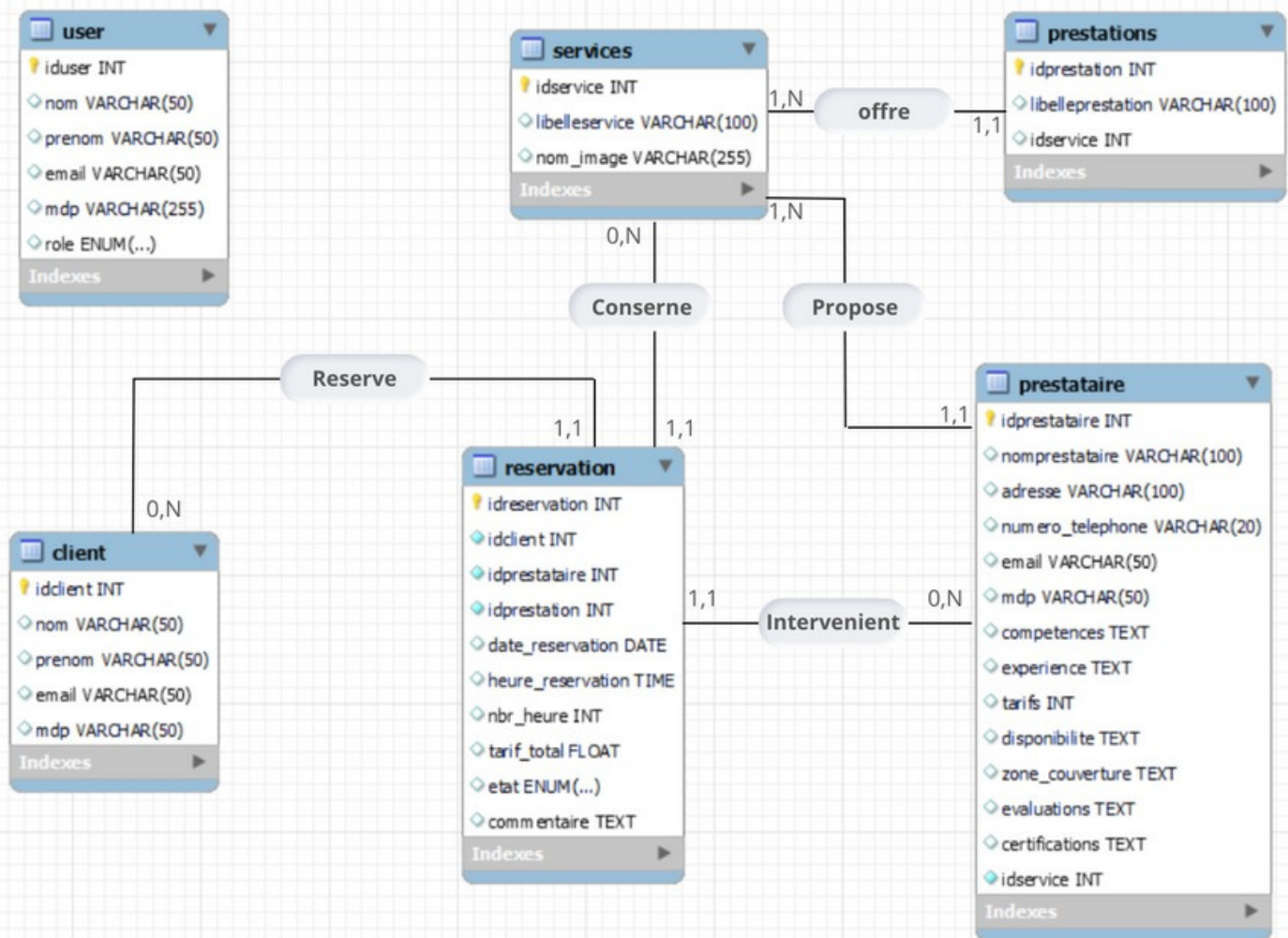
Le modèle conceptuel de données (MCD) décrit la structure des données de la plateforme Mojo. Il identifie les entités, leurs attributs et les relations entre elles

Description du Modèle:

Le modèle comprend les entités suivantes :

1. ***Prestataire:*** Représente les prestataires de services disponibles sur la plateforme Mojo.
2. ***Services:*** Représente les différents types de services proposés par les prestataires. Chaque service a un libellé et peut être associé à une image.
3. ***Prestations:*** Représente les différentes prestations offertes dans le cadre des services. Chaque prestation a un libellé et est associée à un service.
4. ***Client:*** Représente les utilisateurs clients de la plateforme. Il contient des informations telles que le nom, le prénom, l'email et le mot de passe du client.
5. ***Utilisateur:*** Représente les utilisateurs de la plateforme, qui sont les admins
6. ***Réservation:*** Représente les réservations effectuées par les clients pour les prestations des prestataires. Chaque réservation est associée à un client, un prestataire, une prestation, une date, une heure, un nombre d'heures, un tarif total, un état (en attente, confirmée ou annulée) et un commentaire.

MCD



Thecnologies utilisées

Java

Utilisation : Langage de programmation principal utilisé pour le développement de l'application Mojo, tant pour la partie backend que frontend.

MariaDB

Utilisation : Système de gestion de base de données relationnelle utilisé pour stocker les données des utilisateurs, des prestataires, des services et autres informations pertinentes.

Eclipse

Utilisation : Environnement de développement intégré (IDE) utilisé pour coder, déboguer et déployer l'application Java.

Gantt

Utilisation : Outil de planification de projet utilisé pour créer et suivre le diagramme de Gantt, permettant la gestion des tâches et des délais du projet Mojo.

GitHub

Utilisation : Plateforme de développement collaboratif utilisée pour héberger le code source du projet, gérer les versions, faciliter la collaboration et le suivi des problèmes.

Architecture utilisée

Architecture MVC : Utilisée pour structurer l'application en séparant les données (Modèle), la logique métier (Contrôleur) et l'interface utilisateur (Vue). Cela facilite la maintenabilité et l'évolutivité de l'application.

Notre App

Mojo Admin



Email :

MDP :

Annuler

Se Connecter

Mojo Admin

Profil

Services

Prestations

Prestataires

Client

Quitter

Gestion des Services

Nom du service :

Nom de l'image

Annuler

Enregistrer

Filtrer les Services :

Filtrer

Nombre de Services : 1

Gestion des Prestataire

Image du prestataire :

Nom du prestataire :

Adresse :

Numéro de téléphone :

Email de contact :

Mot de passe :

Compétences :

Expérience :

Tarifs :

Disponibilité :

Zone de couverture :

Évaluations :

Certifications :

ID du service :

Annuler

Enregistrer

Filtrer les Services :

Filtrer

iD ...	ima...	no...	adr...	nu...	email	mdp	co...	exp...	tarifs	dis...	zon...	eva...	cert...	ids...

Nombre de Prestataires : 0

Enseignements tirés

Architecture MVC

- **Compréhension de l'architecture MVC:** Ce projet nous a permis de maîtriser le concept et l'implémentation de l'architecture Modèle-Vue-Contrôleur (MVC). Nous avons appris à organiser notre code de manière modulaire, en séparant clairement les données (Modèle), la logique métier (Contrôleur) et l'interface utilisateur (Vue). Cela a rendu notre code plus facile à maintenir et à comprendre.
- **Avantages de l'architecture MVC:** En utilisant l'architecture MVC, nous avons constaté une meilleure réutilisabilité du code, une plus grande extensibilité et une séparation claire des préoccupations. Cela nous a permis de travailler plus efficacement en équipe, en permettant à différents développeurs de travailler sur des parties spécifiques de l'application sans interférer les uns avec les autres.

Travail en groupe

- Collaboration efficace: Ce projet nous a enseigné l'importance de la collaboration efficace au sein d'une équipe. Nous avons appris à bien communiquer, à partager les tâches de manière équitable et à résoudre les conflits de manière constructive. Cela a favorisé un environnement de travail harmonieux et productif.
- Répartition des tâches: Nous avons appris à répartir les tâches de manière équilibrée en fonction des compétences et des disponibilités de chaque membre de l'équipe. Cela nous a permis de maximiser notre efficacité et de progresser plus rapidement vers nos objectifs.

Gestion d'un projet en respectant les délais

- Planification efficace: Nous avons compris l'importance d'une planification efficace pour respecter les délais du projet. En établissant un calendrier réaliste, en définissant des jalons clés et en surveillant régulièrement notre progression, nous avons pu éviter les retards et maintenir le projet sur la bonne voie.