

DAKAR INSTITUTE OF TECHNOLOGY

Habilitation n°3386-21 FEV.2024

Domaine : Sciences et Technologie

Mention : Informatique

Spécialité : BIG DATA

Niveau : Licence 2

RAPPORT DE PROJET

**INSEREZ VOTRE SUJET DE PROJET ICI.**

Présenté par :

AMINATA BA

Soutenu le .../.../... devant le jury composé de :

……………………………………………………............

…………………………………………………................

……………………………………………………............

…………………………………………………................



**Année académique 2024 -2025**

# RESUME

Ce projet porte sur la création d’une plateforme web de covoiturage destinée exclusivement aux professionnels, avec pour objectif de favoriser le partage de trajets domicile-travail à moindre coût. Conçue pour répondre aux besoins de mobilité dans le cadre professionnel, l’application permet aux salariés de proposer ou de réserver des trajets en toute simplicité, dans un environnement sécurisé. Le développement repose sur une architecture web alliant HTML et CSS pour la partie visuelle, Flask pour la logique serveur et SQLAlchemy pour la gestion de la base de données. Chaque utilisateur peut s’inscrire en tant que conducteur ou passager, accéder à un tableau de bord adapté à son rôle, publier des trajets ou effectuer des réservations. L’interface a été pensée pour être intuitive et fonctionnelle, avec un design moderne assurant une bonne expérience utilisateur. En ciblant spécifiquement les professionnels, la plateforme cherche à instaurer un climat de confiance et à encourager une mobilité plus durable et solidaire. Ce projet pose ainsi les bases d’un service utile, éthique et évolutif, avec la perspective d’intégrer à l’avenir des fonctionnalités plus avancées telles que la vérification de l’identité professionnelle ou le développement d’une application mobile.

# REMERCIEMENTS

Je tiens à exprimer ma sincère gratitude à toutes les personnes qui ont, de près ou de loin, contribué à la réalisation de ce projet. Mes remerciements vont particulièrement à mes enseignants pour leur accompagnement constant, leurs conseils éclairés et leur bienveillance tout au long de cette aventure. Leur expertise a été une source d’inspiration précieuse dans l’élaboration de ce travail.  
Je remercie également mes camarades pour leur solidarité, leurs encouragements et les échanges enrichissants qui m’ont permis d’avancer avec motivation et confiance.  
Une pensée reconnaissante va également à mes aînés, les étudiants des promotions précédentes, pour leurs orientations et leur disponibilité, qui m’ont été d’un grand secours à plusieurs étapes du projet.  
Enfin, je remercie chaleureusement ma famille pour son soutien moral et matériel indéfectible. Leur présence constante à mes côtés a été essentielle pour surmonter les difficultés et mener à bien ce projet avec engagement et détermination.

# DEDICACE

À ma famille, pilier indéfectible de mon parcours. Votre amour, vos sacrifices silencieux et votre soutien constant ont été ma plus grande source de force. Vous avez cru en moi même dans les moments de doute, et c’est à vous que je dois l’aboutissement de ce projet.

À mes enseignants, qui m’ont guidée avec rigueur et bienveillance. Vos conseils ont nourri ma réflexion et m’ont permis d’aller au bout de cette réalisation avec confiance.

À mes camarades et amis, pour les échanges, les encouragements et les sourires partagés durant ce parcours.

À toutes les personnes qui, de près ou de loin, ont semé dans mon chemin des graines de savoir, de motivation ou d’inspiration.

Et enfin, à moi-même pour avoir persévéré, appris, chuté parfois, mais surtout pour avoir tenu bon.  
Cette dédicace est le reflet d’un parcours fait de volonté, de soutien et de gratitude.

# LISTE DES FIGURES

**Figure 1: page d'accueil du site.** .............................................................................................. 9

**Figure 2: Formulaire de soumission** ..................................................................................... 10

**Figure 3: Page de connexion administrateur** ....................................................................... 11

**Figure 4: Page administrateur** .............................................................................................. 12

**Figure 5: Page Admin mise à jour** ........................................................................................ 13

# SOMMAIRE

# INTRODUCTION ................................................................................................................................. 1

# OBJECTIFS ET CONCEPT DU PROJET ......................................................................................... 2

**TECHNOLOGIES EMPLOYEES** ...................................................................................................... 4

**ÉTAPES DE DEVELOPPEMENT** ...................................................................................................... 6

**RESULTATS ET FONCTIONNALITES MISES EN PLACE** ......................................................... 9

**LIMITATIONS ET AMELIORATIONS FUTURES** ....................................................................... 16

**CONCLUSION** .................................................................................................................................... 21

#### INTRODUCTION

Dans un contexte marqué par l’urbanisation croissante, les embouteillages récurrents et la montée des préoccupations environnementales, les solutions de mobilité partagée s’imposent progressivement comme des alternatives crédibles et responsables. Le covoiturage, en particulier, répond à un double enjeu : économique, en réduisant les coûts liés aux trajets quotidiens, et écologique, en limitant le nombre de véhicules en circulation. Cependant, malgré ses avantages évidents, il reste sous-exploité dans le milieu professionnel, notamment dans les grandes villes où les salariés effectuent souvent des trajets similaires à des horaires fixes.

La problématique principale réside dans l’absence de plateformes fiables et adaptées spécifiquement aux professionnels, où la sécurité, la confiance et l’organisation sont essentielles. En effet, les solutions existantes sont souvent généralistes, ce qui rend difficile la mise en relation rapide et sécurisée entre collègues ou salariés d’une même zone géographique ou d’un même secteur d’activité. De plus, la crainte de partager un véhicule avec un inconnu constitue un frein pour de nombreux usagers potentiels.

C’est dans cette optique que ce projet a vu le jour : concevoir une **plateforme de covoiturage exclusivement destinée aux professionnels**, visant à faciliter la mise en relation entre salariés souhaitant partager leurs trajets domicile-travail. Cette plateforme permet non seulement de réduire les frais de transport, mais aussi de favoriser la ponctualité, la collaboration et une meilleure gestion du temps dans les entreprises.

Le développement de cette application web repose sur des technologies modernes telles que **HTML**, **CSS** pour la structure et le design, **Flask** pour le backend, et **SQLAlchemy** pour la gestion de la base de données. Le tout a été pensé de manière à garantir une **interface intuitive**, une **expérience utilisateur fluide**, et une **sécurité renforcée** autour des informations personnelles des usagers.

Ce rapport retrace l’ensemble du processus de conception et de réalisation de ce projet, de l’analyse des besoins à la mise en œuvre technique, tout en mettant en lumière les choix effectués, les défis rencontrés et les perspectives d’évolution envisageables pour cette solution innovante.

#### ETUDE DU BESOIN ET OBJECTIS DU PROJETS

Dans le contexte professionnel actuel, les trajets domicile-travail représentent une contrainte quotidienne pour de nombreux salariés. Ces déplacements engendrent à la fois une fatigue considérable et des coûts importants, liés notamment au carburant, à l'entretien du véhicule et aux frais de stationnement. Le stress lié aux embouteillages et à la ponctualité affecte également la qualité de vie des travailleurs.

Le covoiturage apparaît dès lors comme une solution pertinente, à la fois économique, écologique et solidaire. Il permet non seulement de partager les frais, mais aussi de créer un lien social entre collègues ou professionnels d’un même secteur géographique. Dans le cadre de ce projet, l’objectif est de proposer une plateforme exclusivement dédiée aux salariés, garantissant ainsi un environnement sécurisé et fiable.

Les utilisateurs ciblés sont donc exclusivement des professionnels désireux de partager leurs trajets de manière sécurisée. Pour cela, la plateforme doit répondre à plusieurs besoins fondamentaux : simplicité d’utilisation, fiabilité des données, accès restreint aux salariés authentifiés, et gestion fluide des rôles utilisateurs.

L'application permettra ainsi aux **conducteurs** de publier des trajets réguliers ou ponctuels, en renseignant des informations essentielles telles que le lieu de départ, la destination, l’heure, le nombre de places disponibles, ainsi que les détails du véhicule (marque, plaque, permis). De leur côté, les **passagers** pourront consulter la liste des trajets disponibles, effectuer une recherche ciblée et réserver une place selon leurs besoins.

Le système prévoit également une gestion rigoureuse des rôles via un processus d’inscription différencié. Chaque utilisateur, en se connectant, accède à une interface adaptée à son profil (conducteur ou passager), avec des fonctionnalités personnalisées. Cette segmentation permet de garantir une meilleure expérience utilisateur et une sécurité renforcée dans la gestion des trajets.

# CAHIER DE CHARGES

Le cahier des charges définit les objectifs fonctionnels et techniques à atteindre pour garantir la réussite du projet, en répondant de manière claire aux besoins identifiés dans l’étude précédente.

## Objectifs fonctionnels

L’application a pour but de faciliter le partage de trajets entre salariés en mettant à leur disposition une interface intuitive et sécurisée. Pour cela, plusieurs fonctionnalités majeures ont été identifiées :

* **Système d’inscription et de connexion** : chaque utilisateur peut créer un compte avec un rôle spécifique (conducteur ou passager) et se connecter à la plateforme.
* **Rôles utilisateurs distincts** : selon le rôle attribué, l’utilisateur accède à des interfaces et fonctionnalités différentes. Le conducteur peut proposer un trajet, tandis que le passager peut en réserver.
* **Ajout de trajet** : le conducteur a la possibilité de publier un trajet en précisant le lieu de départ, la destination, l’heure, le nombre de places disponibles, le coût, ainsi que des informations sur le véhicule (marque, plaque d’immatriculation, permis).
* **Recherche et réservation de trajets** : le passager peut parcourir la liste des trajets disponibles et réserver une place selon ses besoins.
* **Tableau de bord personnalisé** : une fois connecté, l’utilisateur accède à un tableau de bord adapté à son rôle, lui permettant de gérer ses trajets (proposés ou réservés).

## Objectifs techniques

Pour assurer la performance, la sécurité et la compatibilité de la plateforme, plusieurs choix techniques ont été faits :

* **Application web responsive** : l’interface s’adapte à différents types d’écrans (ordinateurs, tablettes, smartphones) pour garantir une accessibilité optimale.
* **Base de données relationnelle** : l'utilisation de SQLAlchemy permet une gestion robuste des données, avec des relations bien définies entre utilisateurs, trajets et réservations.
* **Navigation fluide et interface claire** : l’architecture des pages et l’organisation des routes Flask sont conçues pour garantir une navigation simple, rapide et cohérente.
* **Authentification sécurisée avec Flask** : la gestion des sessions permet de sécuriser l’accès aux pages et fonctionnalités selon le rôle de l’utilisateur.

## Contraintes

Le projet a été réalisé dans un contexte académique avec plusieurs contraintes à prendre en compte :

* **Temps limité** : la réalisation complète du projet s’est faite sur quelques semaines seulement.
* **Travail individuel** : toutes les étapes ont été menées seule, depuis l’analyse jusqu’à la mise en œuvre.
* **Technologies restreintes** : aucune utilisation de JavaScript, ce qui impose une gestion côté serveur et des interfaces entièrement en HTML/CSS.

# METHODOLOGIE

La réalisation de ce projet a suivi une approche incrémentale et structurée, en partant de la définition des besoins jusqu’au développement, aux tests et à l'amélioration progressive des fonctionnalités. Cette méthode a permis de maintenir une cohérence tout au long du processus et d'intégrer les technologies de manière pertinente.

## Technologies utilisés :

### Frontend

* **HTML (HyperText Markup Language) :** Le langage HTML a servi à construire la **structure des pages** de l’application. Il a permis de créer tous les éléments de l’interface utilisateur : formulaires de connexion, tableaux de bord, formulaires d’ajout de trajet, etc. Chaque page de l’application repose sur un template HTML, rendu via Flask, assurant une navigation fluide entre les vues.
* **CSS (Cascading Style Sheets) :** le CSS a servi à styliser toutes les pages de l’application, afin de rendre l’interface plus agréable, moderne et intuitive. Les feuilles de style ont permis de définir des couleurs harmonieuses, d’adapter l’affichage sur différents écrans (notamment mobiles) et de créer des éléments interactifs comme les boutons, les formulaires et les barres de navigation. Tout le design de l'application repose sur des classes CSS personnalisées écrites manuellement, sans recourir à un framework externe.

### Backend

* **Flask** **:** C’est le micro-frame work Python qui a servi de base au développement du backend de l’application. Sa légèreté et sa flexibilité ont permis de structurer l’ensemble de la logique serveur de manière claire et modulaire. Grâce à Flask, les routes ont été définies pour gérer les différentes actions utilisateur, telles que l’inscription, la connexion, l’ajout de trajet ou la réservation. Il a également permis de gérer les sessions utilisateurs, en distinguant les droits d’accès selon les rôles (conducteur ou passager). L’organisation du code à travers des Blueprints a facilité la séparation des fonctionnalités et la lisibilité du projet. Enfin, Flask a intégré facilement le moteur de template Jinja2, ce qui a permis de générer dynamiquement les pages HTML à partir des données récupérées côté serveur.
* **SqlAlchemy :** a été utilisé pour gérer les interactions entre l’application Flask et la base de données. Grâce à cet outil, il a été possible de modéliser les données sous forme de classes Python représentant les entités principales de l’application (utilisateurs, trajets, réservations), tout en se connectant à une base relationnelle. SQLAlchemy a facilité la création des tables, la définition des relations entre elles, ainsi que les opérations de lecture, d’insertion, de mise à jour ou de suppression. Son utilisation a permis d’écrire un code plus concis, maintenable et sécurisé, tout en bénéficiant des avantages de la programmation orientée objet appliquée aux données.Bas du formulaire

## Structure du projet :

Le développement de l’application a été structuré de manière claire et modulaire afin de garantir la maintenabilité, la lisibilité et l’évolutivité du code. L’arborescence du projet repose principalement sur une séparation des responsabilités entre les **templates HTML**, les **fichiers Python** pour la logique métier et les **fichiers CSS** pour la mise en forme.

### Les templates :

Ce dossier contient l’ensemble des fichiers HTML qui définissent l’interface utilisateur. Chaque fichier a été conçu pour répondre à une fonctionnalité précise de l’application, facilitant la navigation, l’interaction et la clarté de l’interface.

Voici une description des principaux fichiers présents dans ce dossier :

* **base.html** : il s’agit du template de base commun à toutes les pages. Il contient la structure HTML principale (en-tête, barre de navigation, importation des styles, zones de contenu dynamiques). Les autres templates héritent de ce fichier pour assurer une cohérence d’affichage sur tout le site.
* **accueil.html** : cette page constitue la page d’accueil du site. Elle présente une vue d’introduction au service de covoiturage professionnel, avec des liens vers l’inscription, la connexion et éventuellement une description du service ou des avantages du covoiturage.
* **login.html** : ce template contient le formulaire de connexion permettant aux utilisateurs (conducteurs ou passagers) d’accéder à leur espace personnel selon leur rôle.
* **register.html** : cette page permet l’inscription d’un nouvel utilisateur. Elle recueille les informations essentielles comme l’email, le mot de passe, le rôle (conducteur ou passager).
* **conducteur.html** : une fois connecté, l’utilisateur ayant le rôle de conducteur accède à cette page. Elle affiche son tableau de bord, listant ses trajets proposés et donnant accès à l’ajout de nouveaux trajets ou à d’autres fonctionnalités spécifiques au conducteur.
* **ajouter\_trajet.html** : cette page permet aux conducteurs de publier un nouveau trajet. Le formulaire recueille toutes les informations nécessaires : point de départ, destination, horaire, nombre de places, itinéraire, ainsi que le coût.
* **passager.html** : ce template constitue le tableau de bord des passagers. Il permet d'accéder à l'outil de recherche de trajets, de consulter les réservations passées et de voir les informations du conducteur.
* **recherche.html** : cette page permet aux passagers de rechercher un trajet disponible en fonction de critères tels que la destination, ou le départ. Les résultats affichent les trajets compatibles, avec la possibilité de réserver une place.
* **confirmation.html** : ce fichier affiche un message de confirmation suite à une réservation réussie de la part du passager. Il permet à ce dernier de voir les détails de son trajet.

### Le dossier static/css :

Ce dossier contient les feuilles de style de l’application. Le fichier principal est :

* **style.css** : ce fichier regroupe l’ensemble des règles CSS personnalisées permettant d’uniformiser l’apparence du site. Il gère les couleurs, les polices, les boutons, les formulaires et la mise en page globale. Il est importé dans base.html.

## Les fichiers python :

Les fichiers Python du projet assurent la logique backend et l'interaction avec la base de données.

* **app.py** : point d’entrée principal de l’application. Il initialise l’instance Flask, configure la base de données, et lance le serveur.
* **routes.py** : regroupe toutes les routes de l’application (connexion, inscription, ajout de trajet, réservation, etc). Il gère aussi les messages flash, les redirections et la vérification des rôles utilisateur.
* **models.py** : contient les modèles de la base de données définis avec SQLAlchemy. On y retrouve les classes Utilisateur, Trajet et Réservation, avec leurs colonnes et relations.

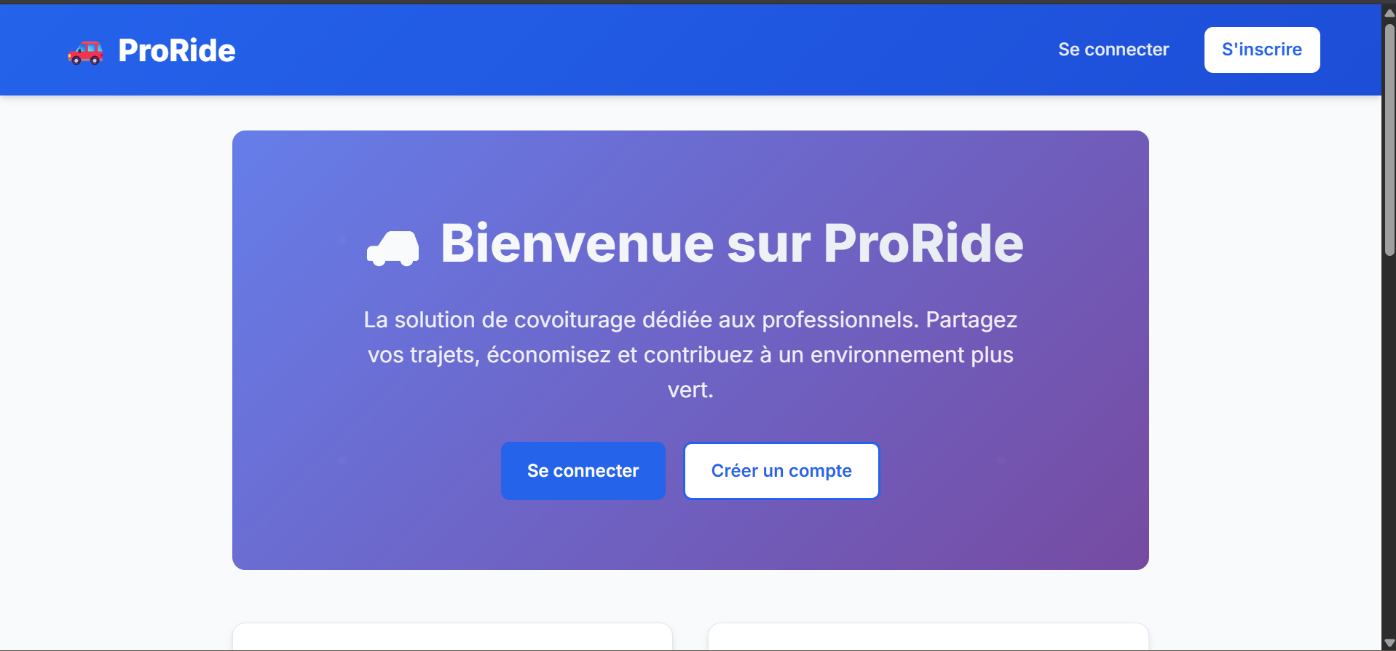
Cette organisation permet une bonne séparation des responsabilités : initialisation, logique métier et gestion des données sont clairement réparties.

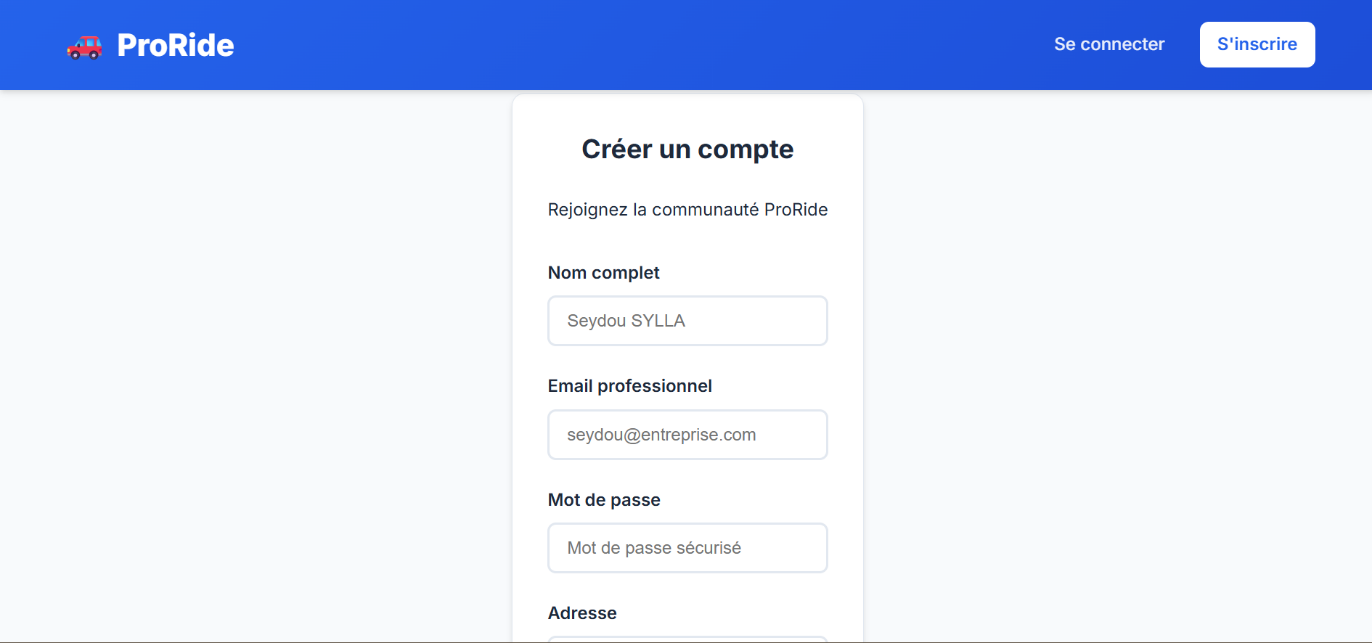
# FONCTIONNALITES REUSSIES ET RESULTATS OBTENUS

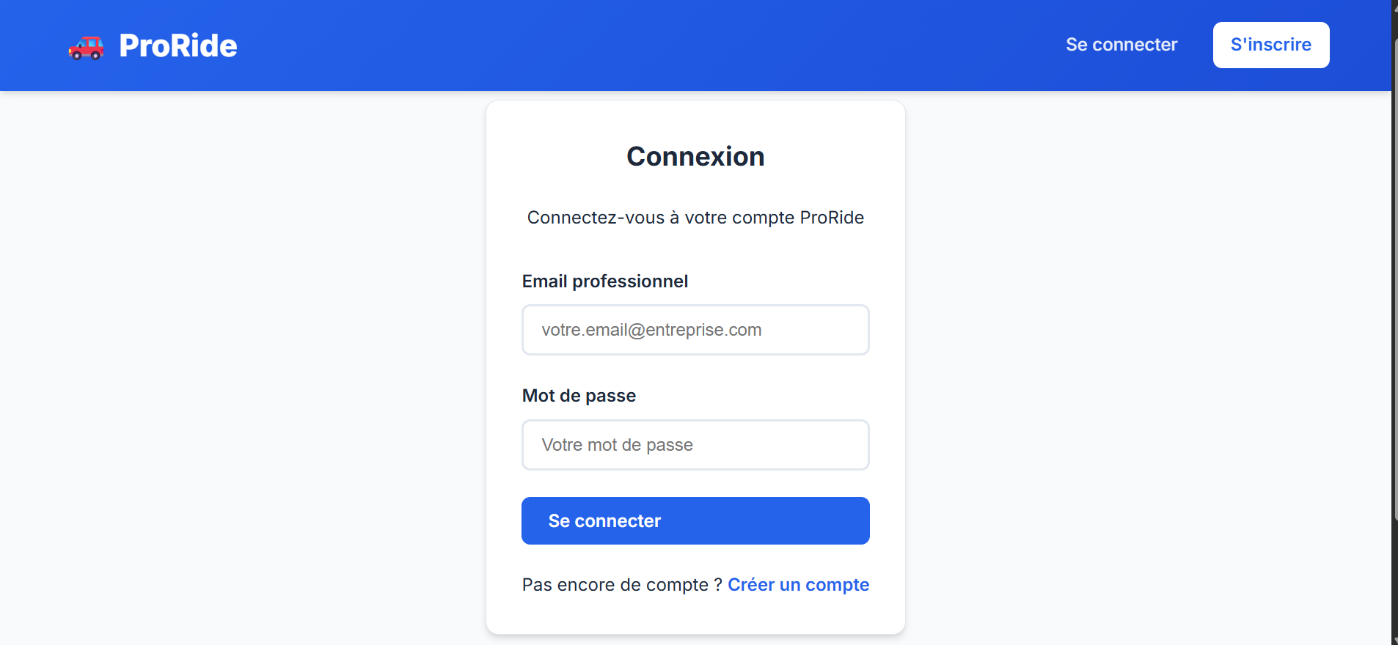
Le développement de la plateforme a permis de mettre en œuvre un ensemble cohérent de fonctionnalités essentielles, alignées avec les objectifs fonctionnels définis au préalable. L’application distingue efficacement les rôles utilisateur, garantit une navigation fluide, et offre un parcours simple et intuitif.

## Inscription et connexion des utilisateurs :

L’utilisateur peut désormais s’inscrire en choisissant un rôle : **conducteur** ou **passager**. Les informations requises sont différentes selon le rôle Une fois inscrit, l’utilisateur peut se connecter à l’aide de son email et mot de passe. Le système reconnaît automatiquement le rôle de l’utilisateur et le redirige vers **le tableau de bord adapté**.

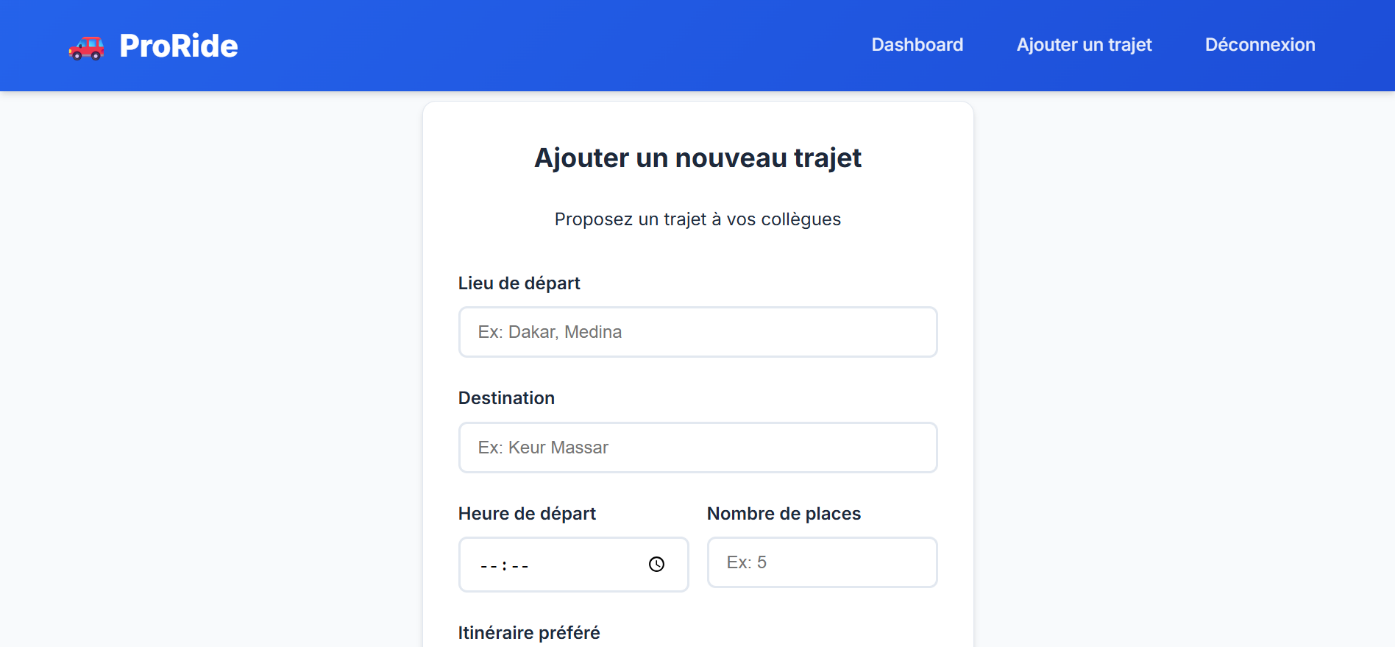
**





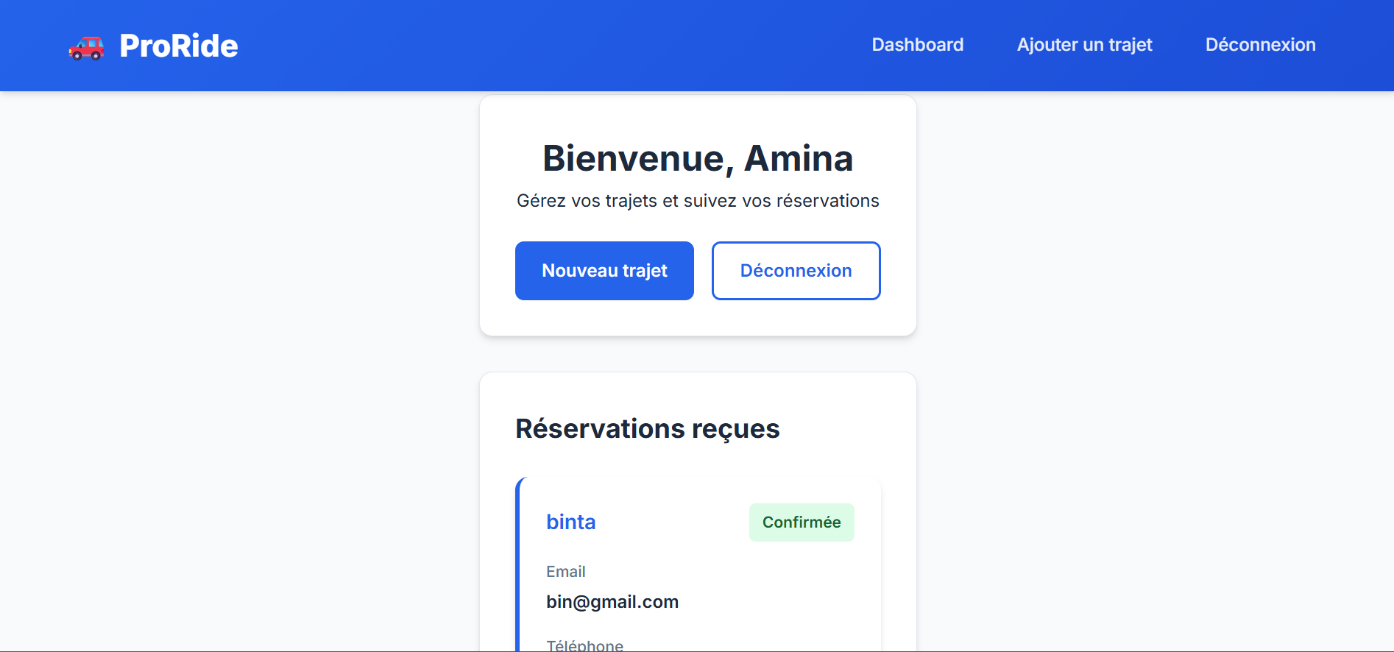
## Ajout de trajets par le conducteur :

Une fois connecté, le conducteur peut ajouter un nouveau trajet en remplissant un **formulaire détaillé** incluant : lieu de départ, destination, heure de départ, nombre de places, itinéraire, coût. Ces trajets sont ensuite listés sur son tableau de bord.



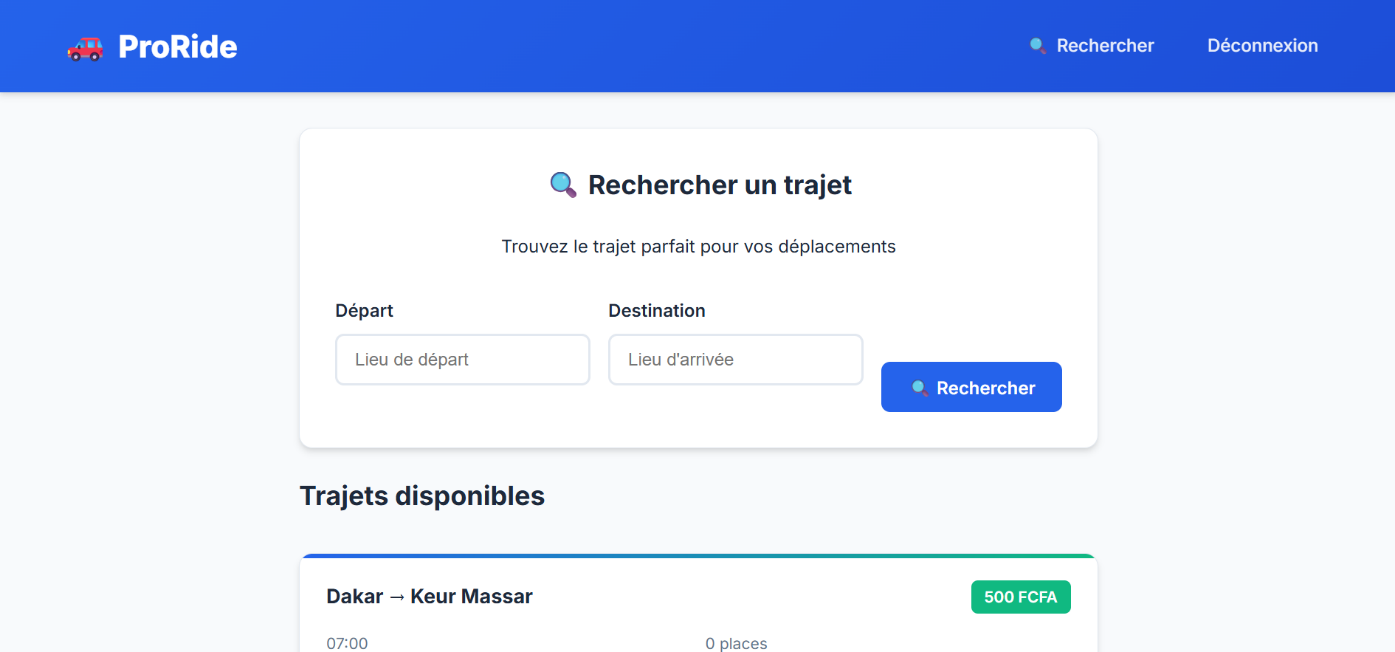
## ****Réception des réservations par le conducteur :****

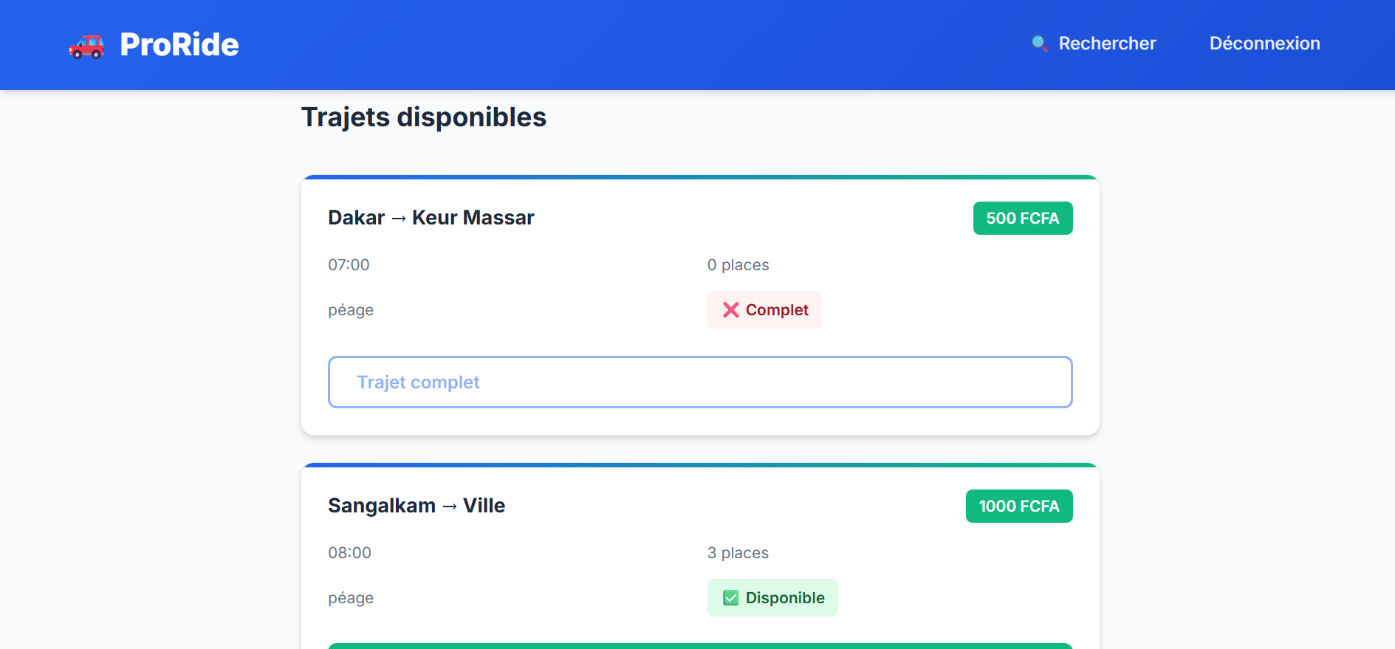
Les conducteurs peuvent **voir les réservations effectuées** par les passagers pour leurs trajets. Cela leur permet de suivre facilement le nombre de places restantes et les trajets en cours.



## Recherche de trajets pour le passager :

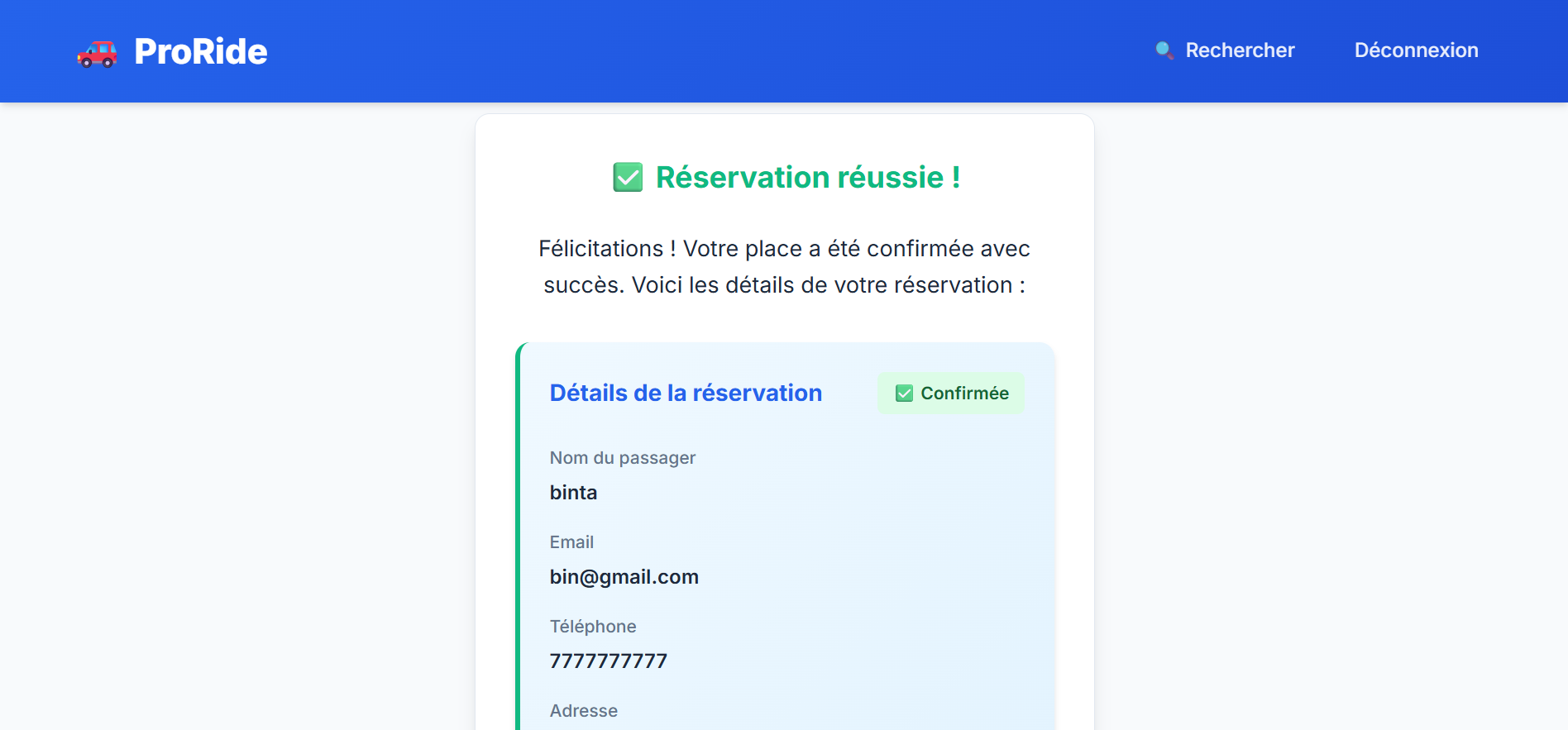
Les passagers ont accès à un moteur de recherche qui leur permet de trouver rapidement les trajets disponibles selon la destination, le lieu de départ ou même les deux. Les résultats sont affichés de façon claire et permettent une consultation rapide des détails du trajet.





## Réservation d’un trajet par le passager :

Après avoir trouvé un trajet correspondant à ses critères, le passager peut réserver une place directement en un clic tout en voyant le nombre de place disponible, s’il n’y a plus de place, il verra que c’est complet et il ne pourra pas réserver. Après ça, sa réservation est confirmé et il pourra voir les détails de son trajets.



# DIFFICULTES RENCONTREES

Au cours de la réalisation de ce projet de, plusieurs difficultés techniques et organisationnelles ont été rencontrées. Bien que la majorité des étapes aient été menées à bien, certaines parties ont demandé plus d’efforts et de recherches que prévu.

## Connexion à la base de données :

L'une des premières difficultés a concerné la gestion de la base de données. Initialement, j'avais choisi d'utiliser **SQLite** en pensant que cela suffirait pour gérer les données des utilisateurs et des trajets. Cependant, en avançant dans le développement, je me suis vite rendu compte que la gestion directe des fichiers SQLite devenait limitée et peu flexible, notamment lorsqu'il fallait effectuer des relations plus complexes entre les tables.

Pour pallier cela, j'ai décidé de me tourner vers **SQLAlchemy**, un ORM (Object Relational Mapper) très utilisé avec Flask. Au début, l'adaptation n'a pas été simple, notamment pour comprendre la logique des modèles, des sessions et des requêtes avec SQLAlchemy. Mais après plusieurs lectures de documentation, tutoriels et essais, j'ai fini par comprendre son fonctionnement et à bien l’intégrer dans mon projet. Ce changement m’a permis de mieux structurer mes tables et de faciliter les interactions avec la base.

## Difficultés liés avec le style et le design :

Une autre difficulté a été la partie stylisation de la plateforme. J’avais initialement prévu d’utiliser Tailwind **CSS**, un framework moderne qui permet de construire des interfaces rapidement avec des classes utilitaires. Cependant, la configuration de Tailwind dans mon environnement de travail avec Flask s’est révélée plus compliquée que prévu. Malgré plusieurs tentatives d’installation via npm et d’intégration dans mes templates, je n’ai pas réussi à faire fonctionner Tailwind correctement.

Ainsi, j’ai finalement décidé de me rabattre sur du CSS classique. Bien que cela ait limité les possibilités de design avancé, cela m’a permis de garder le contrôle sur l’apparence du site et d’obtenir un rendu relativement satisfaisant. Cela a toutefois demandé beaucoup de travail manuel, notamment pour rendre le site un minimum responsive et agréable à utiliser.

# ****PERSPECTIVES D’EVOLUTION****

Afin d’améliorer l’efficacité, la sécurité et l’ergonomie de la plateforme, plusieurs pistes d’évolution sont envisagées pour les futures versions de l’application :

* **Ajout de l’annulation de réservation** : permettre aux passagers de se désister en cas d’imprévu offrirait plus de flexibilité et permettrait une meilleure gestion des trajets par les conducteurs.
* **Renforcement des informations sur les conducteurs** : exiger des données supplémentaires comme la marque, le modèle, l’année du véhicule, une photo de la voiture et une preuve du permis renforcera la transparence et la confiance entre utilisateurs.
* **Vérification des emails professionnels** : pour limiter l’accès à la plateforme aux seuls salariés, l’inscription pourrait être restreinte aux adresses professionnelles en vérifiant le nom de domaine au moment de l’enregistrement.
* **Développement d’une application mobile** : une version mobile de la plateforme faciliterait l’utilisation depuis un smartphone, avec une interface adaptée et des fonctions comme la réservation rapide ou les notifications en temps réel.
* **Affichage des trajets sur une carte interactive** : l’intégration d’une carte permettrait aux passagers de visualiser les trajets disponibles géographiquement, facilitant ainsi la sélection.
* **Système de notation et d’avis** : offrir la possibilité aux utilisateurs de noter et commenter leurs expériences améliorerait la fiabilité de la plateforme et encouragerait un comportement responsable.
* **Création de trajets récurrents** : les conducteurs pourraient enregistrer des trajets réguliers, évitant ainsi une saisie répétitive quotidienne ou hebdomadaire.
* **Messagerie interne sécurisée** : une fonctionnalité de messagerie directe entre conducteurs et passagers permettrait d’échanger des informations sans partager de contacts personnels.
* **Envoi de notifications aux utilisateurs** : mettre en place des alertes automatiques par mail ou SMS (avant un trajet ou après une réservation) renforcerait l’engagement et améliorerait l’organisation des déplacements.
* **Amélioration du tableau de bord** : enrichir l’espace utilisateur avec des filtres, des statistiques ou une meilleure organisation visuelle améliorerait l’expérience globale et la gestion personnelle des trajets et réservations.

# CONCLUSION

La réalisation de ce projet de plateforme de covoiturage professionnel a été une expérience riche, à la fois sur le plan technique et humain. Dans un contexte où les problématiques de mobilité, de coûts de transport et de respect de l’environnement sont de plus en plus pressantes, cette solution propose une alternative pertinente pour les salariés désireux d’optimiser leurs trajets quotidiens. Elle favorise l'entraide entre professionnels tout en contribuant à la réduction du trafic et des émissions polluantes.

Au fil du développement, plusieurs technologies ont été mobilisées de manière cohérente pour construire une application web fonctionnelle, intuitive et sécurisée. L’utilisation de **Flask** a permis de structurer proprement le backend et d’organiser les routes de manière claire. Grâce à **SQLAlchemy**, la gestion des données s’est faite efficacement via des modèles bien définis, assurant la cohérence entre utilisateurs, trajets et réservations. Enfin, **HTML et CSS** ont permis de concevoir une interface simple et accessible, facilitant la navigation pour tous les utilisateurs.

Les fonctionnalités principales ont été mises en place avec succès : inscription avec rôles distincts, connexion, ajout de trajets pour les conducteurs, recherche et réservation pour les passagers, redirection selon le rôle, tableau de bord personnalisé, etc. Ces avancées témoignent d'une compréhension solide des besoins du terrain et de leur traduction concrète en outils numériques.

Ce projet n’est qu’une première étape vers une solution plus complète et plus robuste. Les perspectives identifiées ouvrent la voie à de nombreuses améliorations, tant en termes de sécurité, d’expérience utilisateur que de fonctionnalités supplémentaires. Une application mobile, une messagerie interne, ou encore une vérification automatique des utilisateurs professionnels seraient autant de moyens d’augmenter l’impact et la portée de cette plateforme.

En somme, ce projet a permis de renforcer mes compétences en développement web tout en répondant à un besoin réel du quotidien. Il s’inscrit dans une démarche d’innovation sociale et de durabilité, avec l’ambition de proposer à terme une solution utile, fiable et pérenne.

# SOURCES

* **Documentation officielle de Flask**

Ce site propose une documentation complète sur le Framework Flask, qu’on a utilisé pour la création de mon site web de gestion de vente et location de voitures.

* + lien: <https://flask.palletsprojects.com/en/3.0.x/>o Date de consultation : 12 Septembre 2024
* **Tutoriel YouTube : Créer des applications web simples avec Flask et JavaScript** Cette vidéo YouTube nous a aidé à comprendre comment créer des applications web simples à l'aide de Flask et JavaScript.
  + Lien : <https://www.youtube.com/watch?v=AYmcV3b7lWQ>o Date de consultation : 19 Septembre 2024
* **Outil de recherche Perplexity et chatgpt**

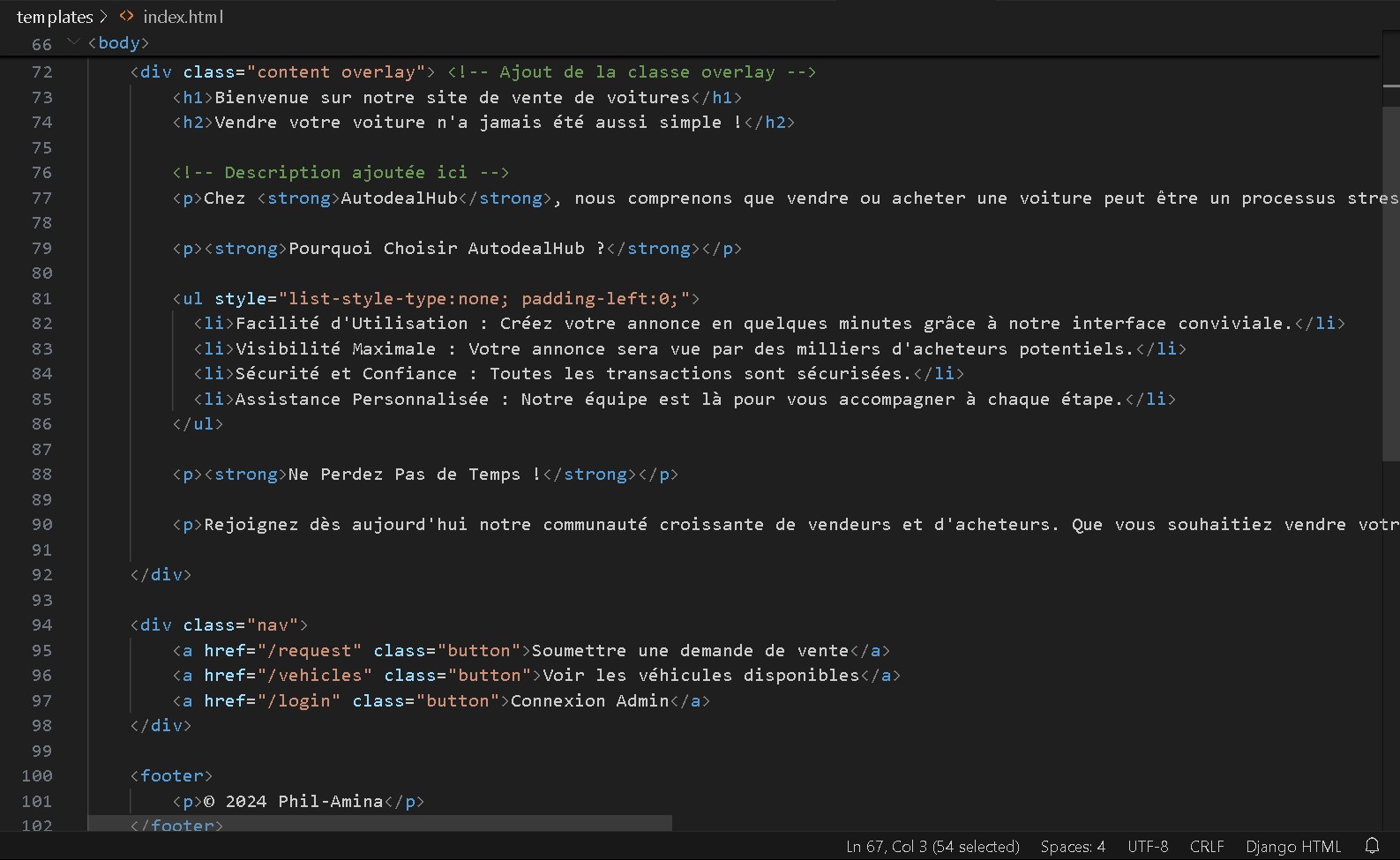
On a utilisé ces deux outils de recherche en ligne, pour trouver des réponses et des explications sur des questions techniques liées à mon projet. o Date de consultation : Du 19 au 25 Septembre 2024

* **Google**

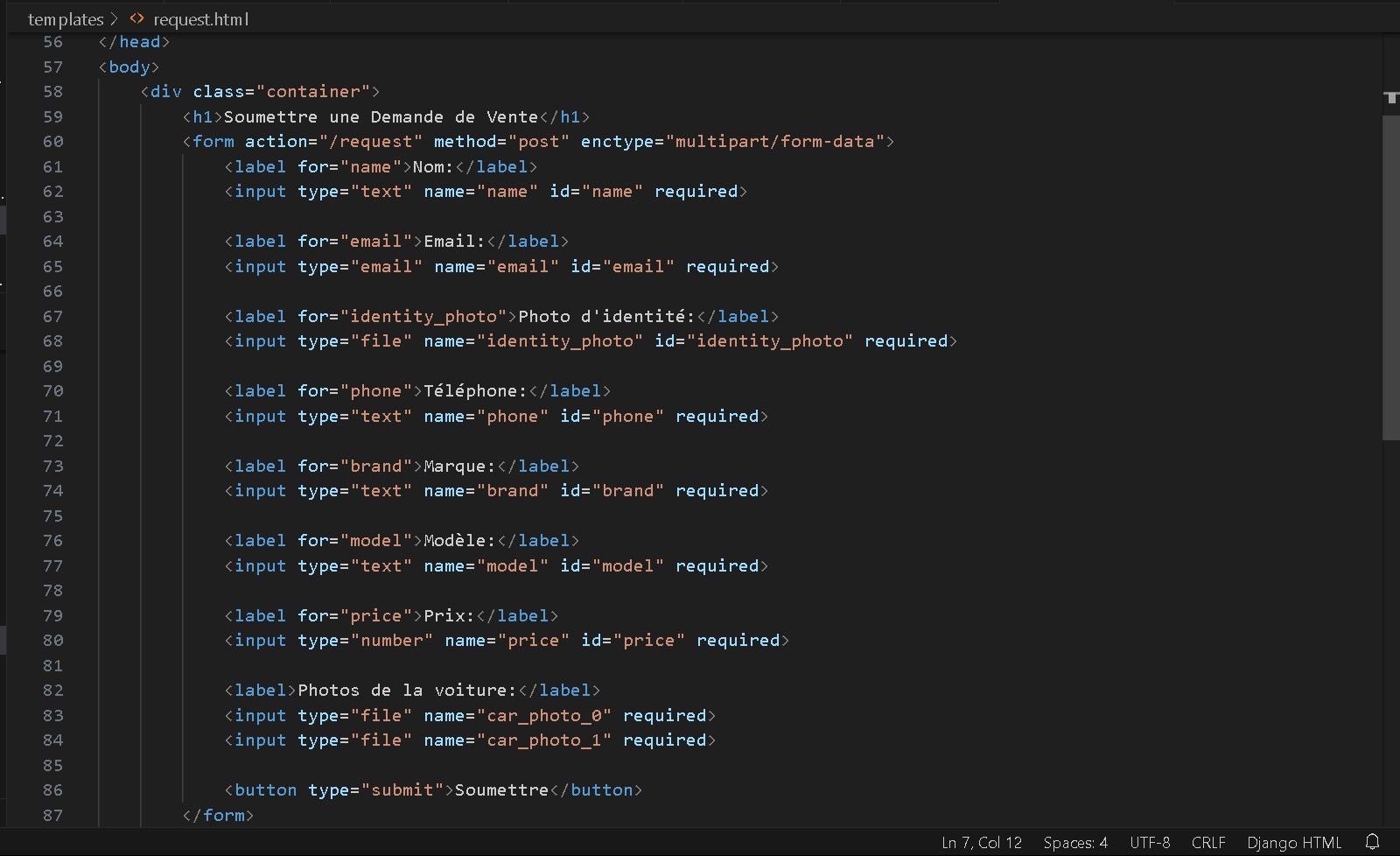
Google a été utilisé comme moteur de recherche pour accéder à diverses informations techniques et exemples pratiques pendant la réalisation du projet. o Date de consultation : Du 15 au 23 Septembre 2024

# ANNEXES

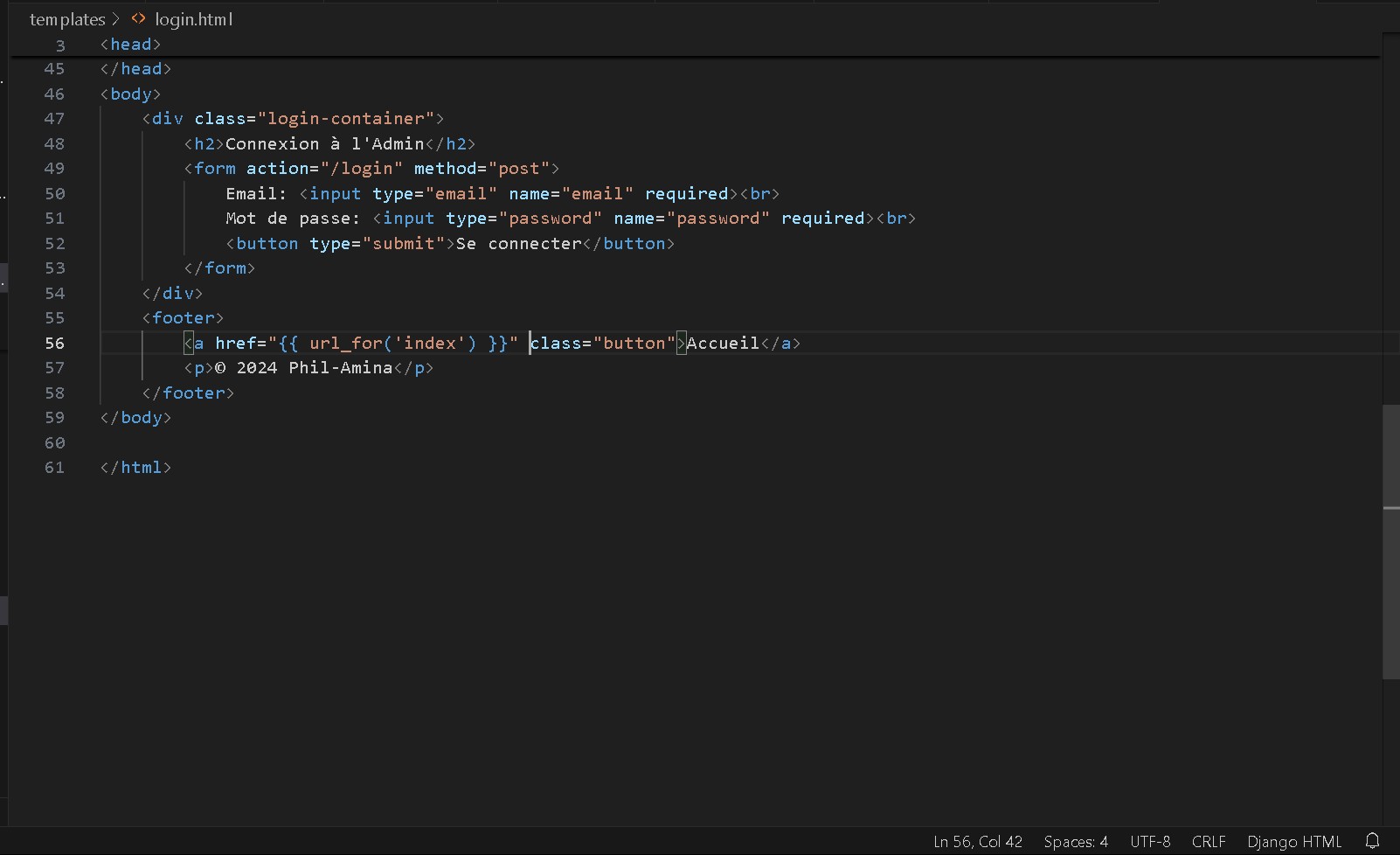
## 1. Extrait code HTML de la page d’accueil



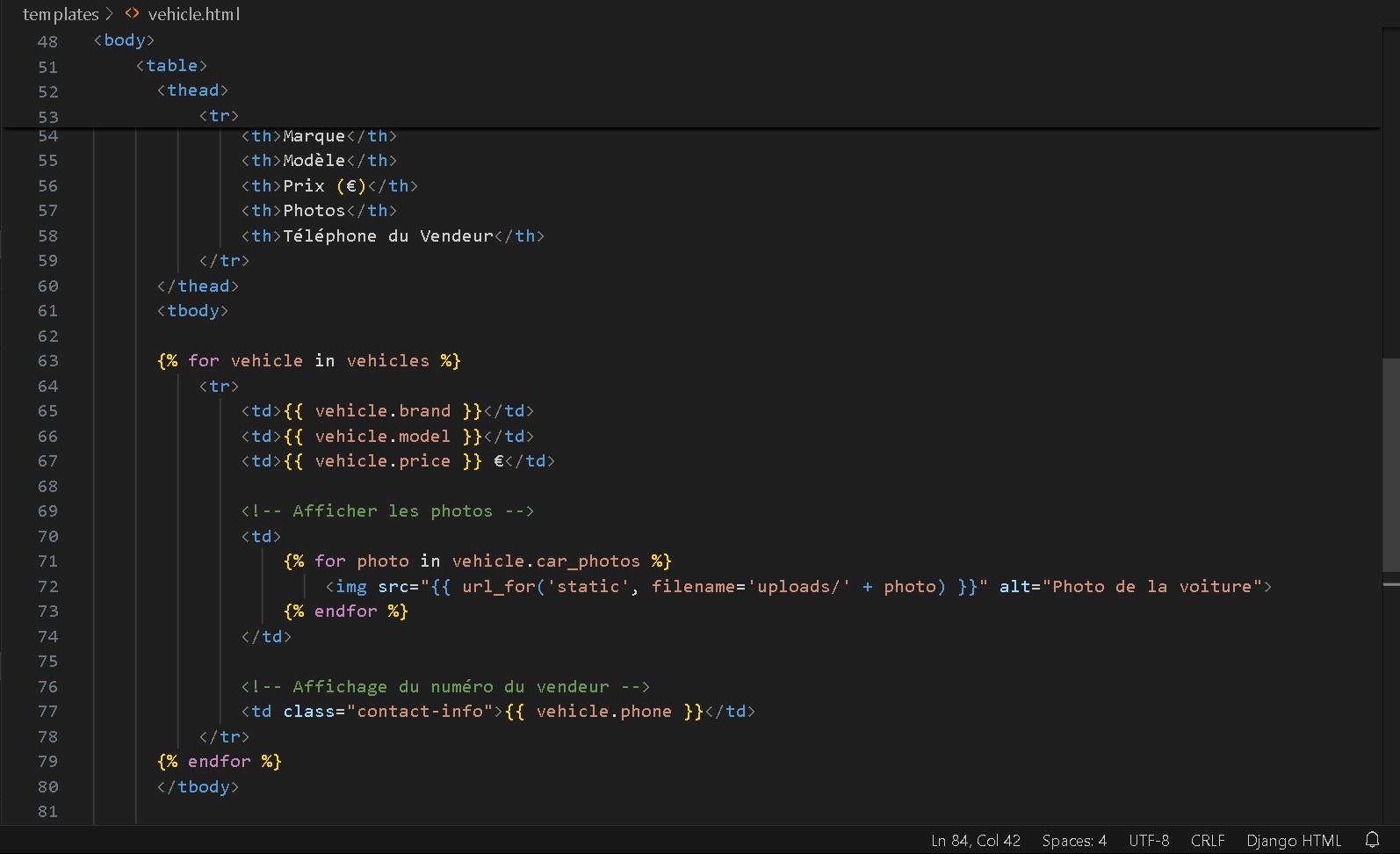
## 1. Extrait du code HTML du formulaire de soumission



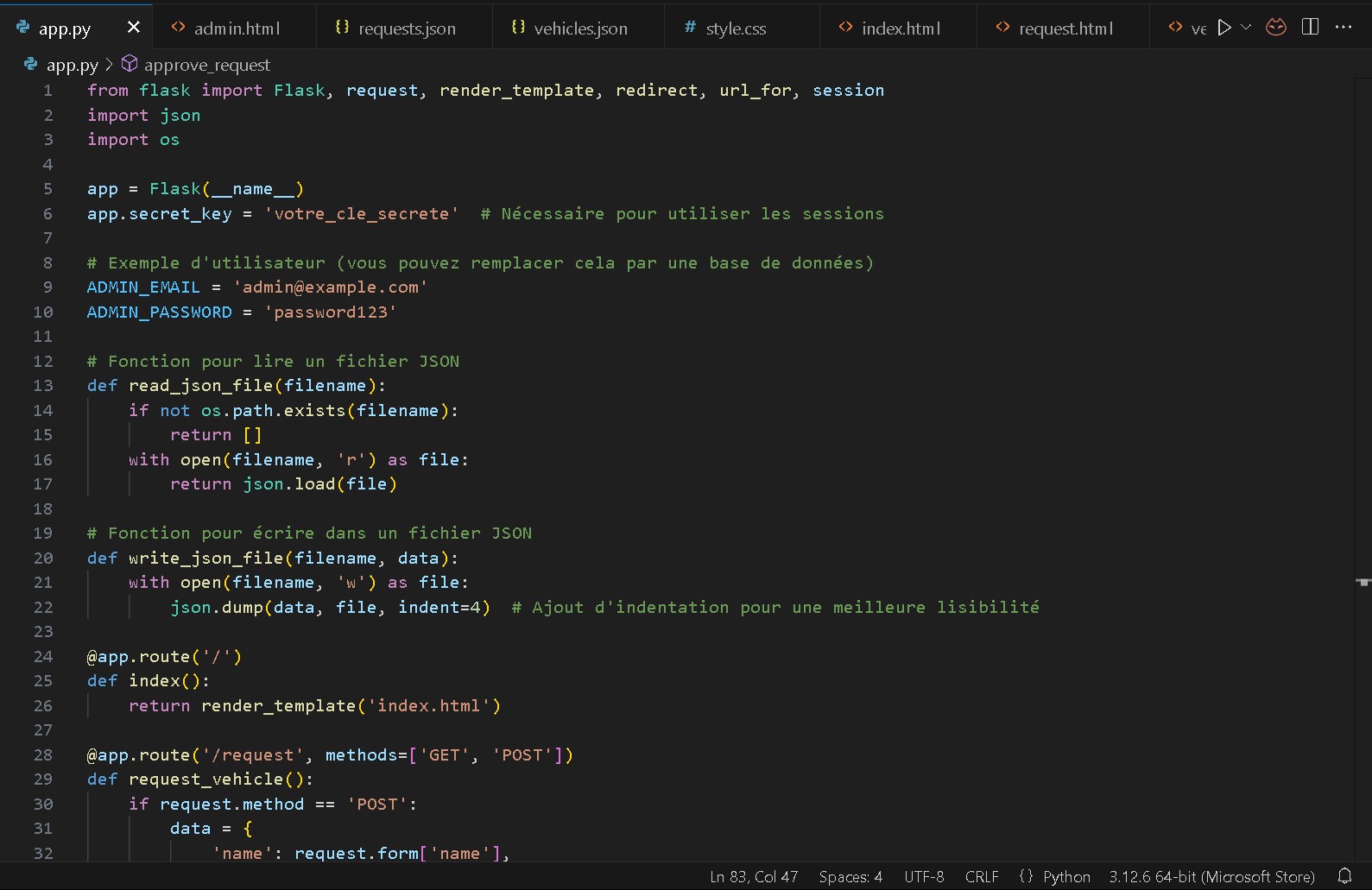
## 2. Extrait du code HTML de la page de connexion admin



## 3. Extrait du code HTML de la page d’annonces



## 4. Extrait de code de l'application Flask



##### TABLES DES MATIERES

[**RESUME** I](#_Toc22256)

[**REMERCIEMENTS** II](#_Toc22257)

[**DEDICACE** III](#_Toc22258)

[**LISTE DES FIGURES** IV](#_Toc22259)

[**SOMMAIRE** V](#_Toc22260)

[**INTRODUCTION** 1](#_Toc22261)

[**OBJECTIFS ET CONCEPT DU PROJET** 2](#_Toc22262)

[**1-** **Objectifs du projet :** 2](#_Toc22263)

[**2-** **Concept du projet :** 3](#_Toc22264)

[**TECHNOLOGIES EMPLOYEES** 4](#_Toc22265)

[**1.** **Frontend** 4](#_Toc22266)

[**a)** **HTML :** 4](#_Toc22267)

[**b)** **CSS :** 4](#_Toc22268)

[**2.** **Backend** 4](#_Toc22269)

[**Flask :** 4](#_Toc22270)

[**ÉTAPES DE DEVELOPPEMENT** 6](#_Toc22271)

[**1-** **Conception de l'architecture** 6](#_Toc22272)

[**a)** **Définition des pages principales :** 6](#_Toc22273)

[**b)** **Planification du flux de travail :** 7](#_Toc22274)

[**2-** **Intégration des fonctionnalités clés** 8](#_Toc22275)

[**RESULTATS ET FONCTIONNALITES MISES EN PLACE** 9](#_Toc22276)

[**1.** **Fonctionnalités réussies** 9](#_Toc22277)

[**1.** **Difficultés rencontrées** 15](#_Toc22278)

[**LIMITATIONS ET AMELIORATIONS FUTURES** 16](#_Toc22279)

[**CONCLUSION** 21](#_Toc22280)

[**SOURCES** A](#_Toc22281)

[**ANNEXES** B](#_Toc22282)

[**1.** **Extrait code HTML de la page d’accueil** B](#_Toc22283)

[**1.** **Extrait du code HTML du formulaire de soumission** B](#_Toc22284)

[**2.** **Extrait du code HTML de la page de connexion admin** C](#_Toc22285)

[**3.** **Extrait du code HTML de la page d’annonces** C](#_Toc22286)

[**4.** **Extrait de code de l'application Flask** D](#_Toc22287)