Rapport

De L’application ‘inCarpool’

Réalisé par :

Mahdi Boughariou

2ème Ingénierie en génie logiciel

TD1 / TP1

Introduction :

L'application ‘inCarpool’ est conçue avec Spring Boot et se divise en deux parties : l'interface administrateur (admin) et l'interface utilisateur (user). L'admin a un contrôle complet sur la gestion des utilisateurs, des offres, et des réservations, avec la possibilité de valider ou refuser des demandes. L’utilisateur peut créer un compte, rechercher des covoiturages, confirmer des réservations, et gérer son profil. Il peut également soumettre une demande pour devenir un offreur de covoiturage, qui doit être approuvée par l'admin avant de pouvoir proposer des trajets.

Fonctionnalités de l'admin :

**Tableau de bord :**

-Visualiser le nombre total de clients, d'offres et de réservations.

-Voir les 10 derniers clients et les 6 dernières offres.

**Gestion des clients :**

-Voir la liste des clients classés par statut : approved, not approved, pending.

**Gestion des demandes de publication d'offres :**

-Voir la liste des utilisateurs dont la demande est en pending.

-Accepter ou rejeter la demande pour permettre à l'utilisateur de publier des offres.

Fonctionnalités de l'utilisateur :

**Création et gestion de compte :**

-Créer un compte.

-Se connecter à son compte.

-Modifier ses informations personnelles.

**Recherche et réservation de covoiturage :**

-Rechercher des covoiturages disponibles.

-Confirmer une réservation pour un covoiturage.

-Consulter les covoiturages confirmés.

-Annuler une confirmation de covoiturage.

**Demande de publication d'offres :**

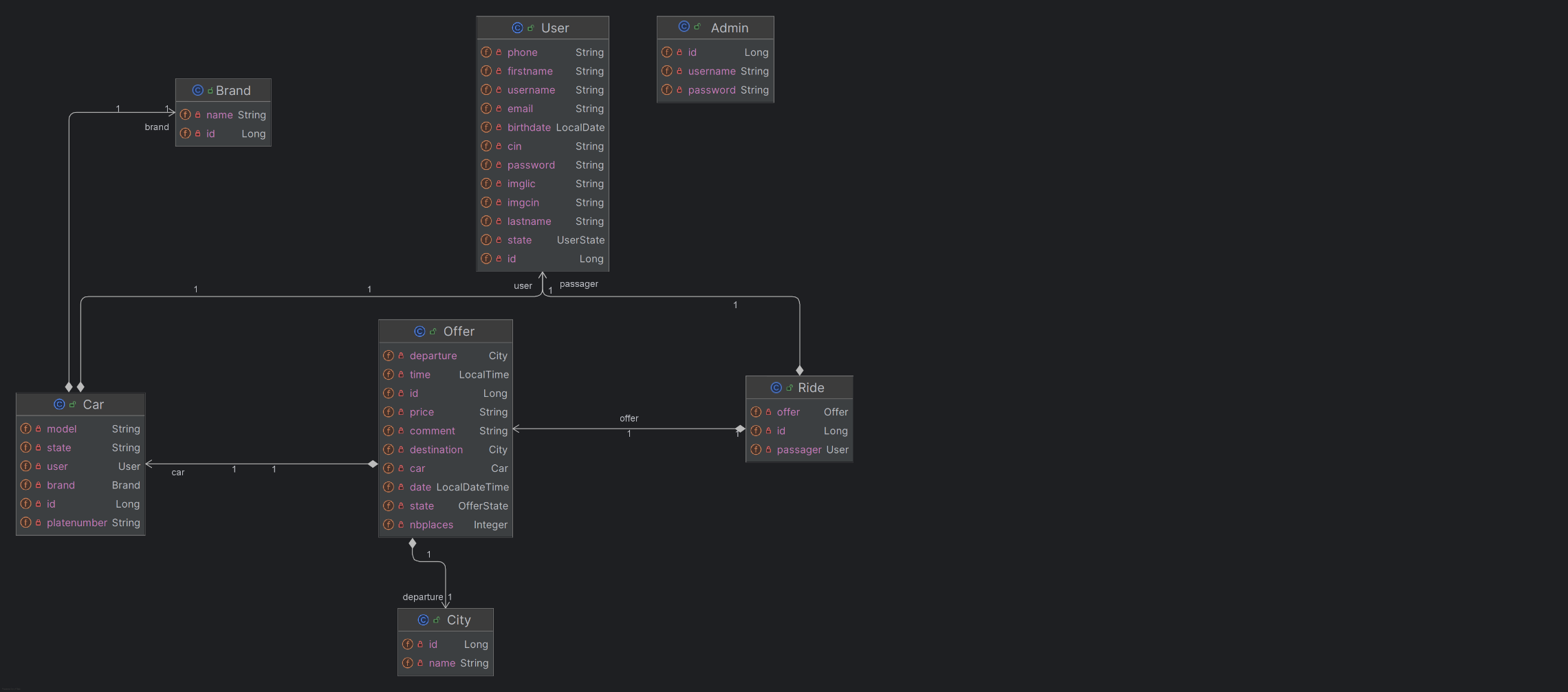
-Créer une demande pour devenir un offreur de covoiturage en fournissant son CIN et son permis de conduire.

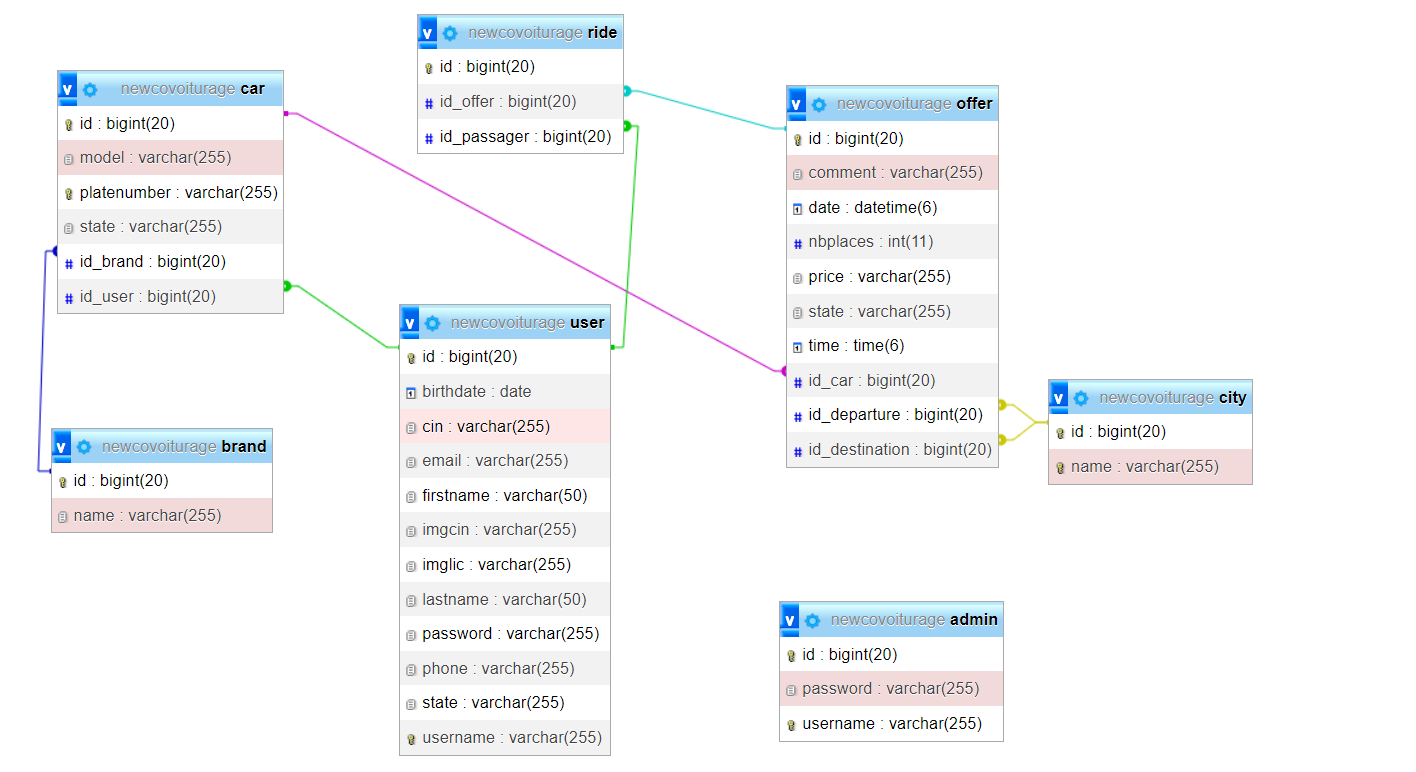
**Publication d'offres (après approbation de l'admin) :**

-Ajouter des véhicules avec lesquels il peut voyager.

-Publier des offres de covoiturage.

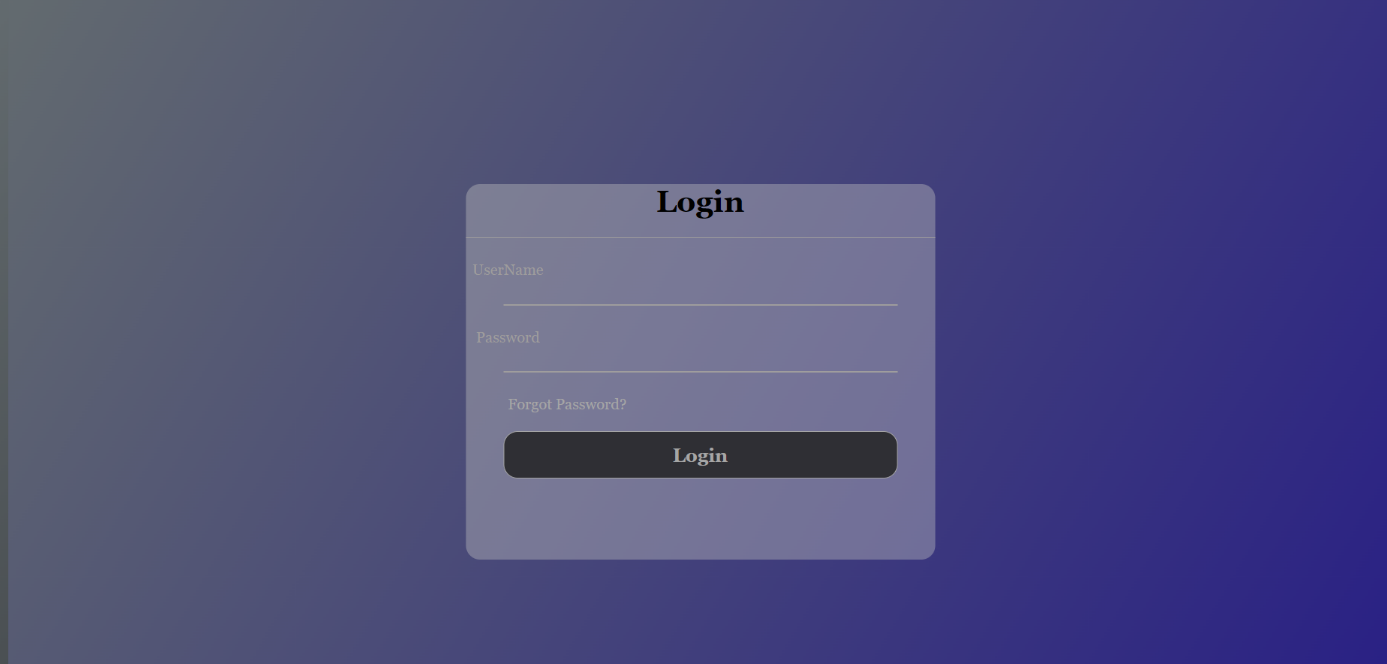
Diagramme de classe :

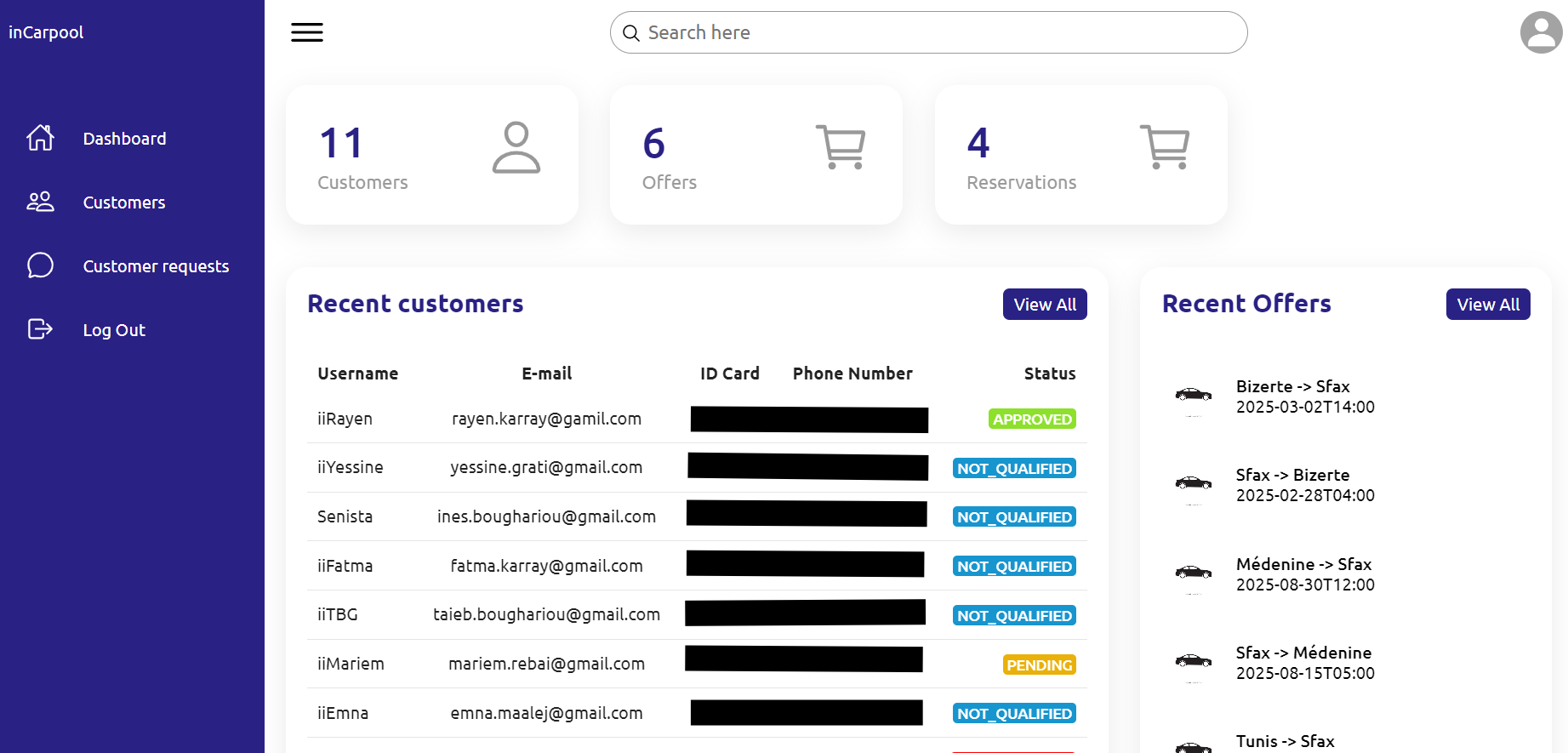


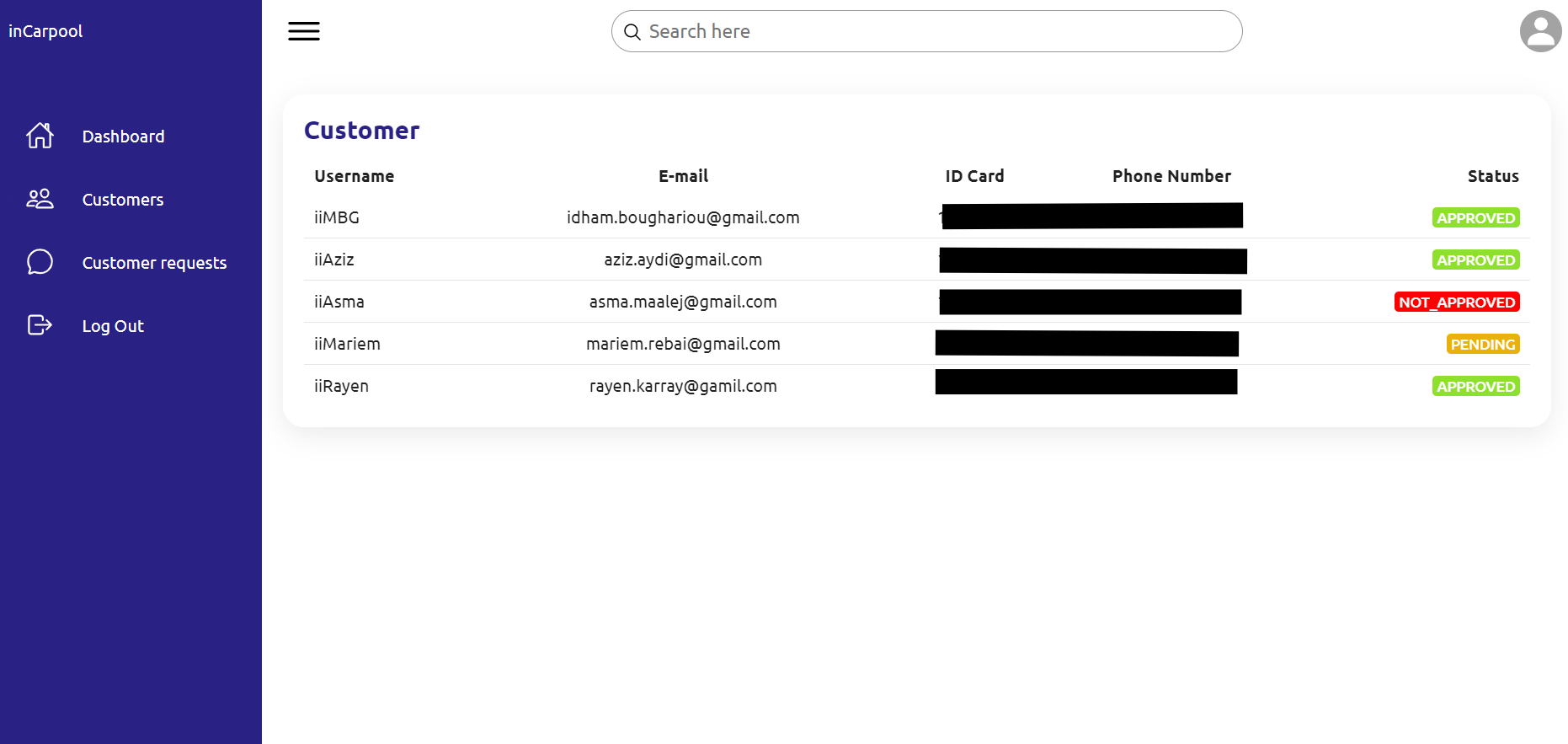


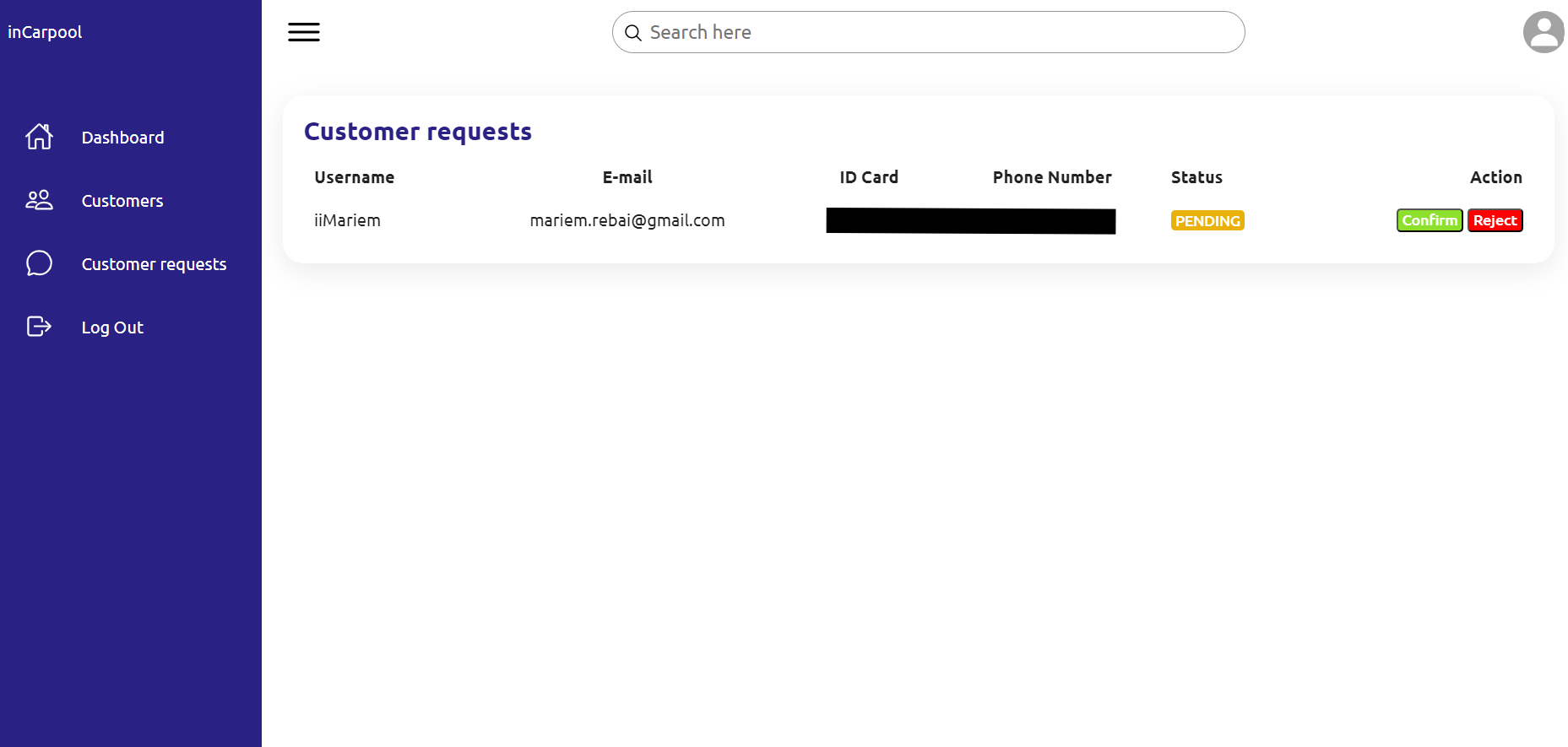
Interfaces :

* **Admin**

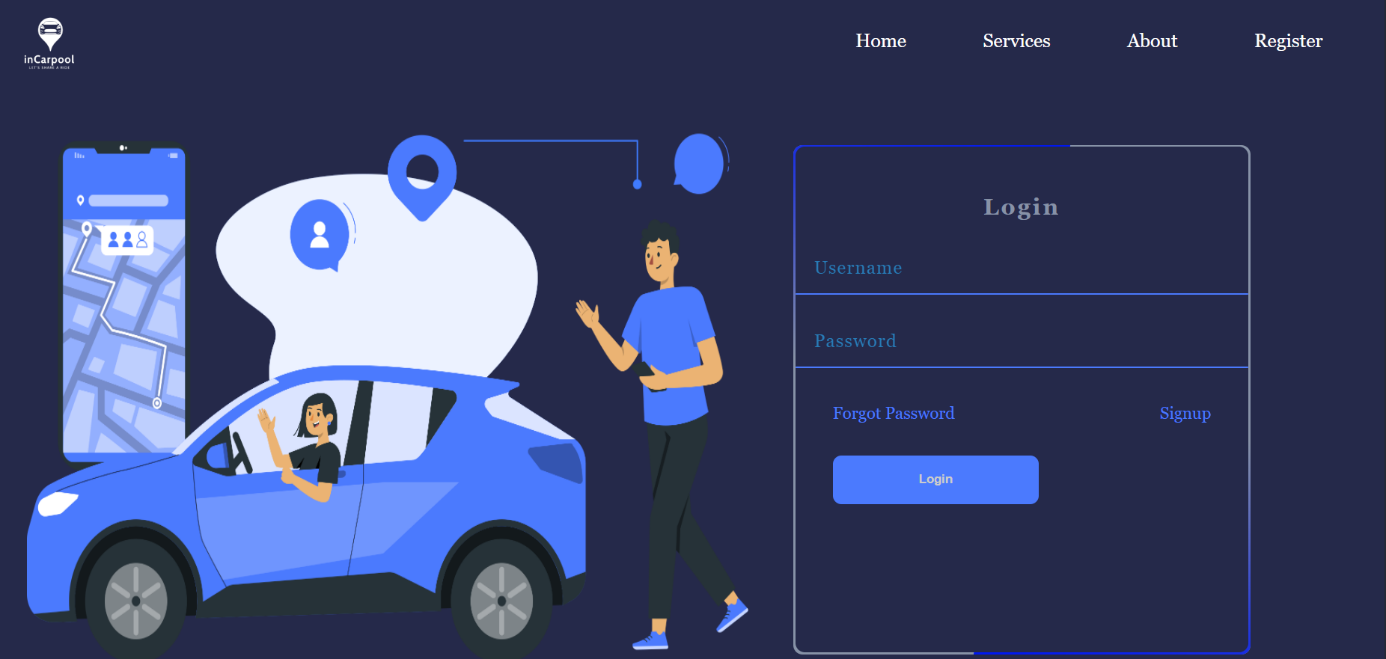
****

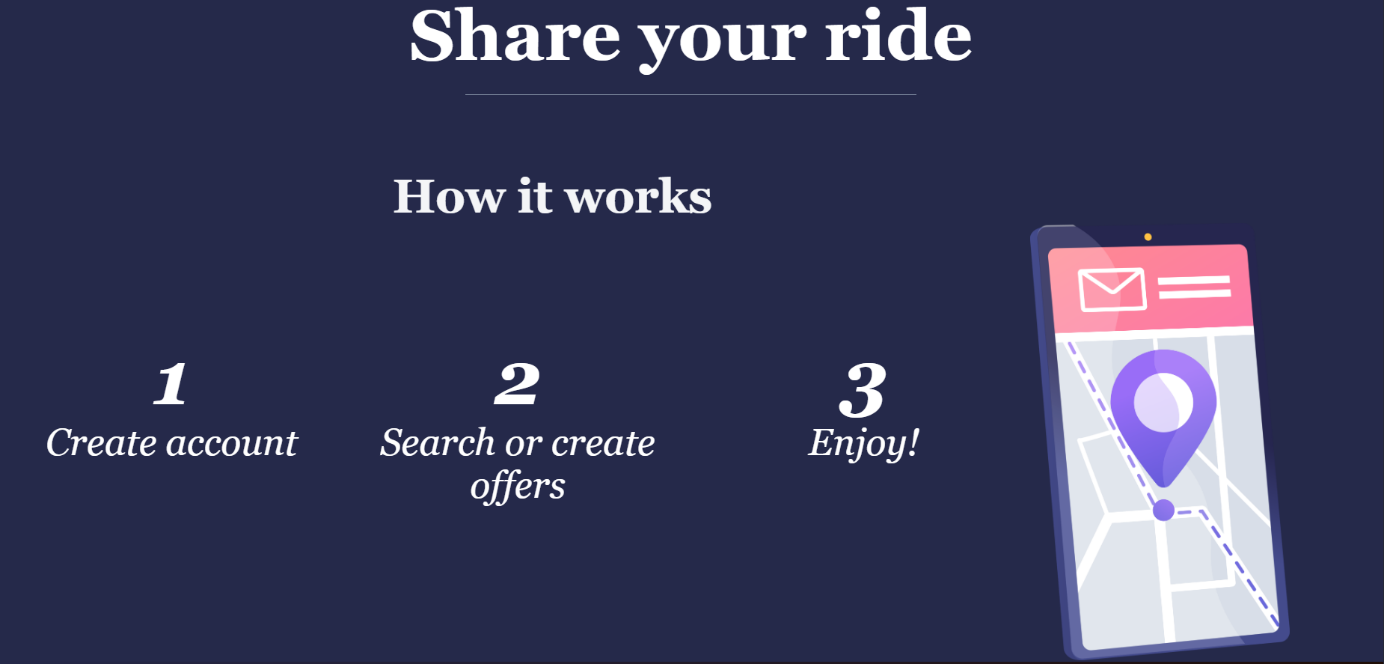




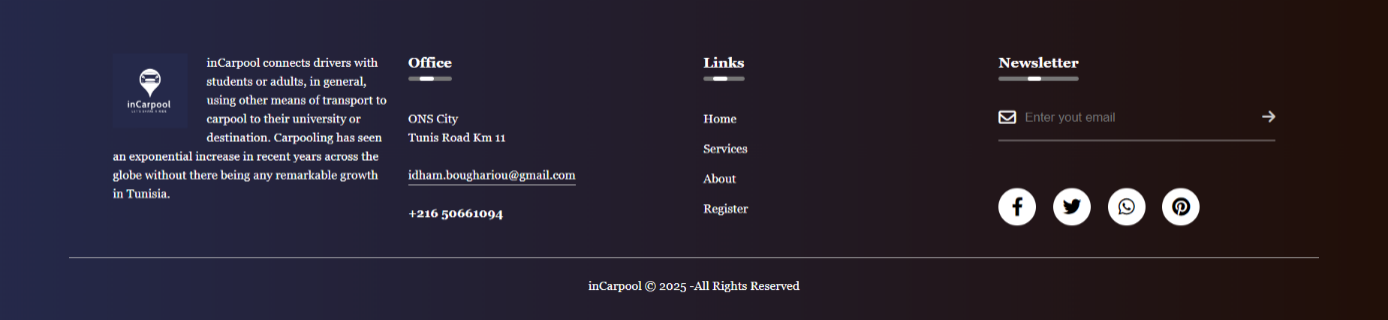


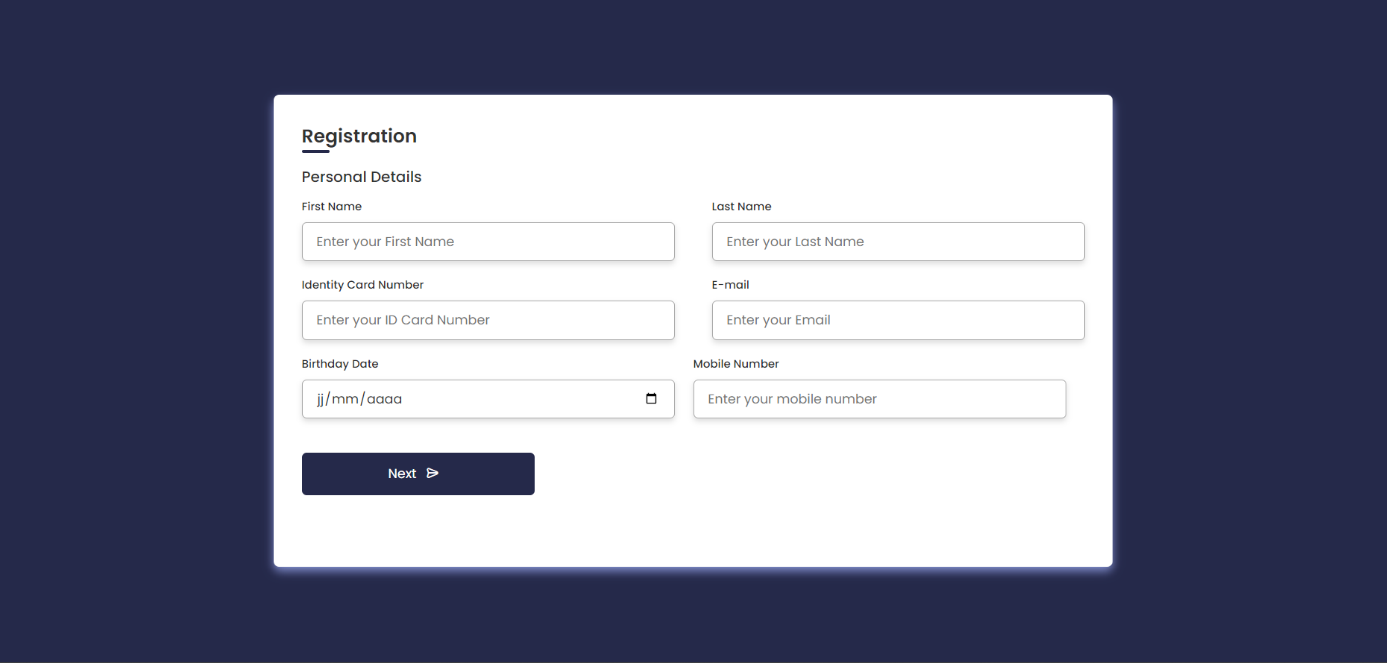
* **User**

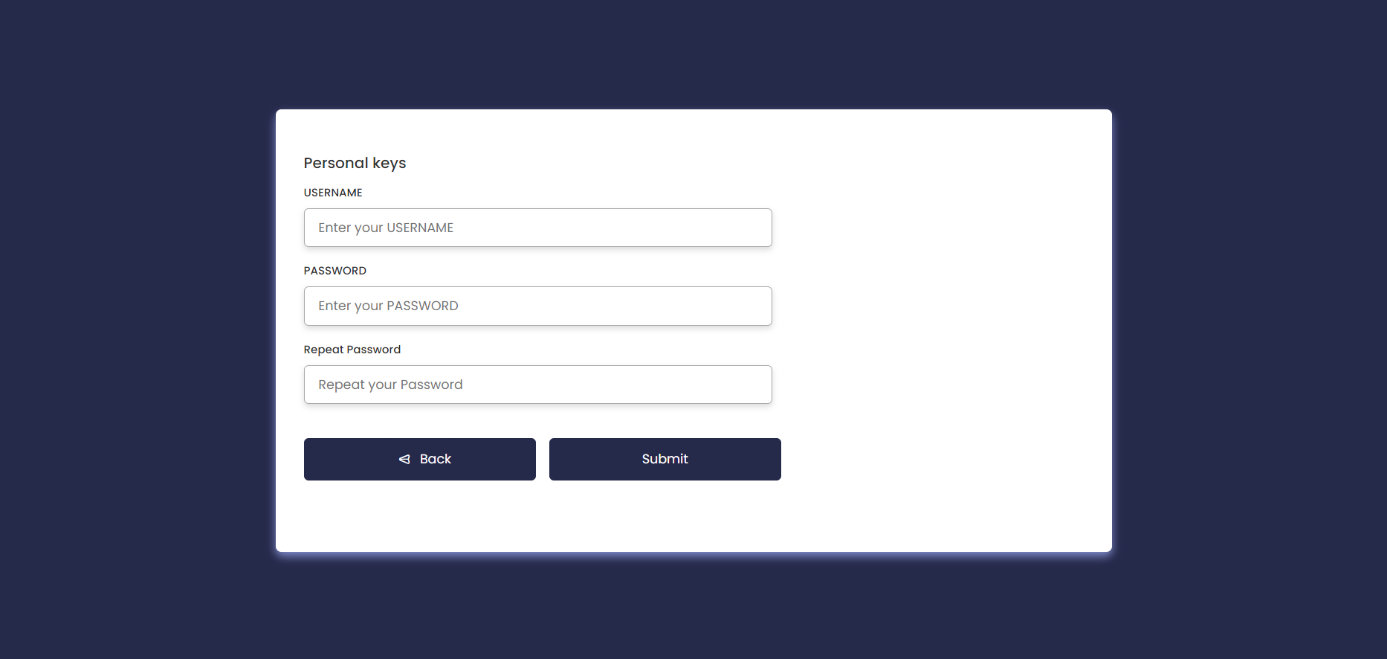
****

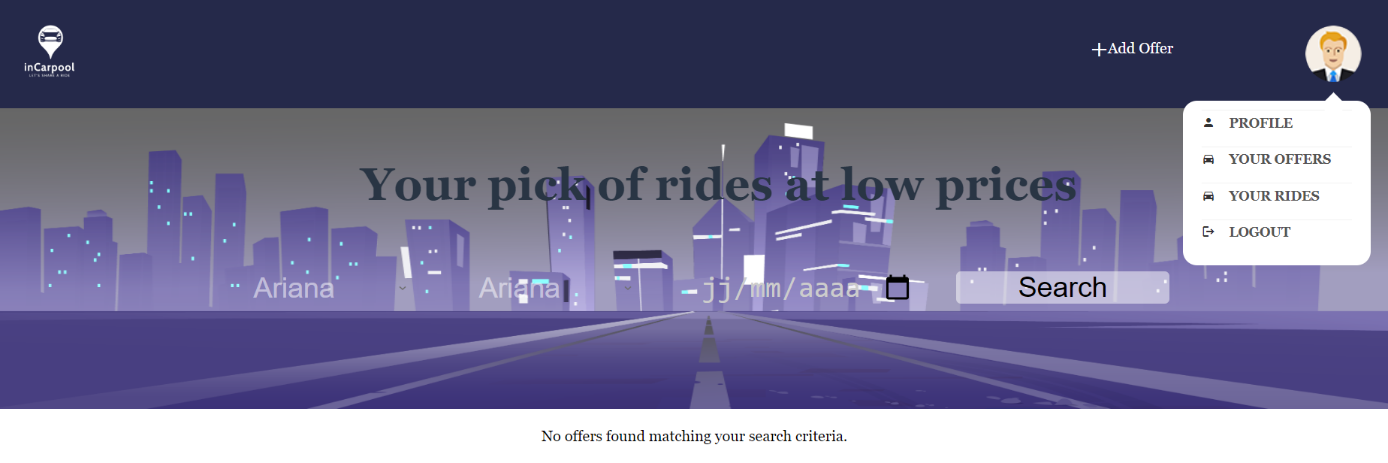


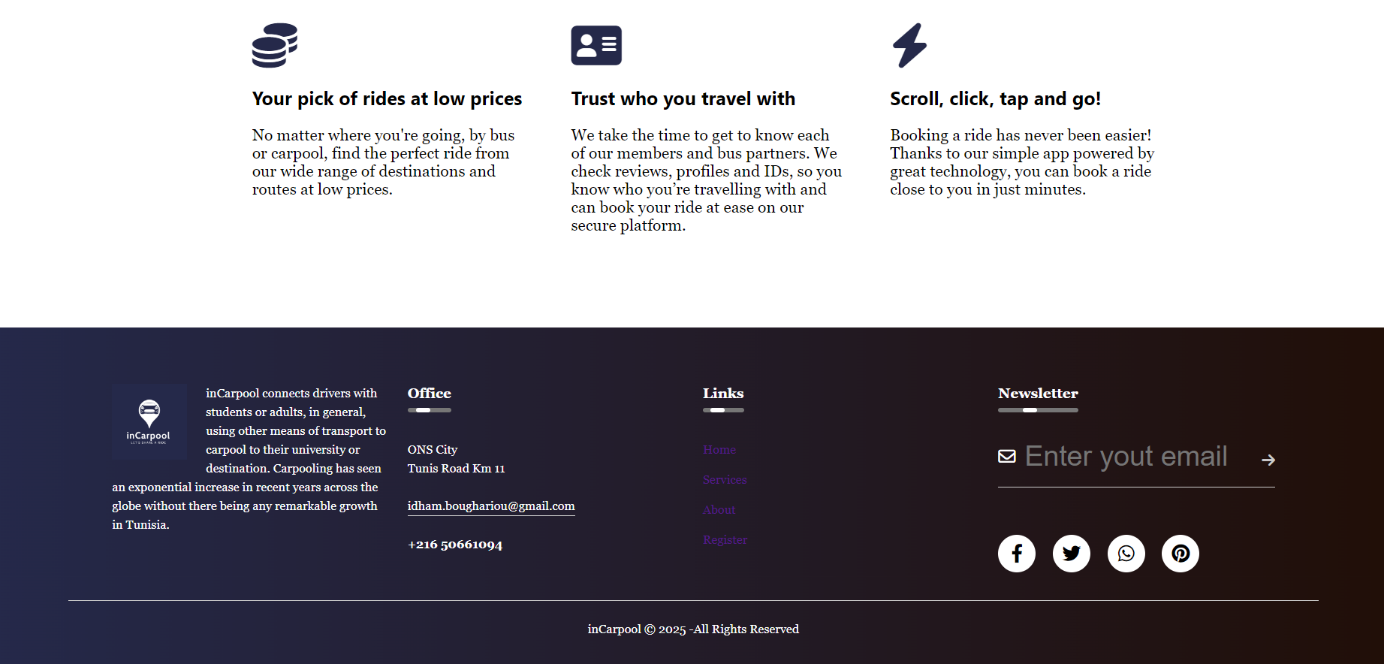


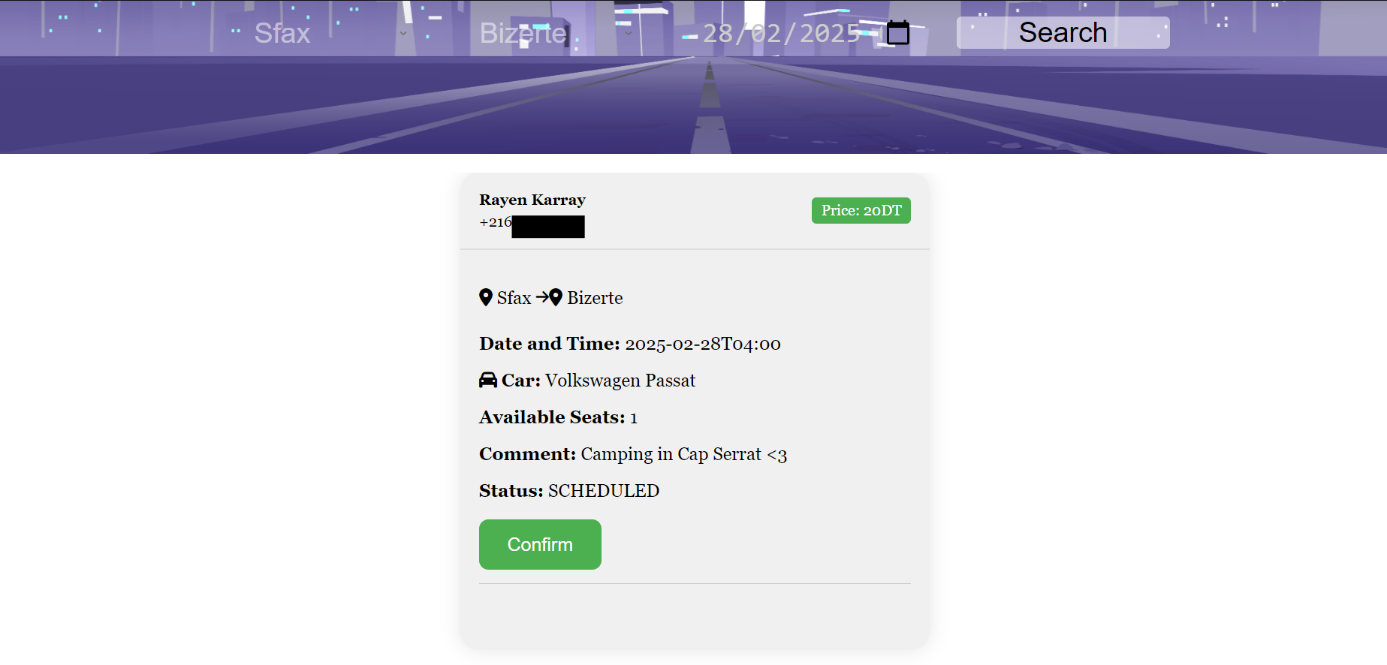


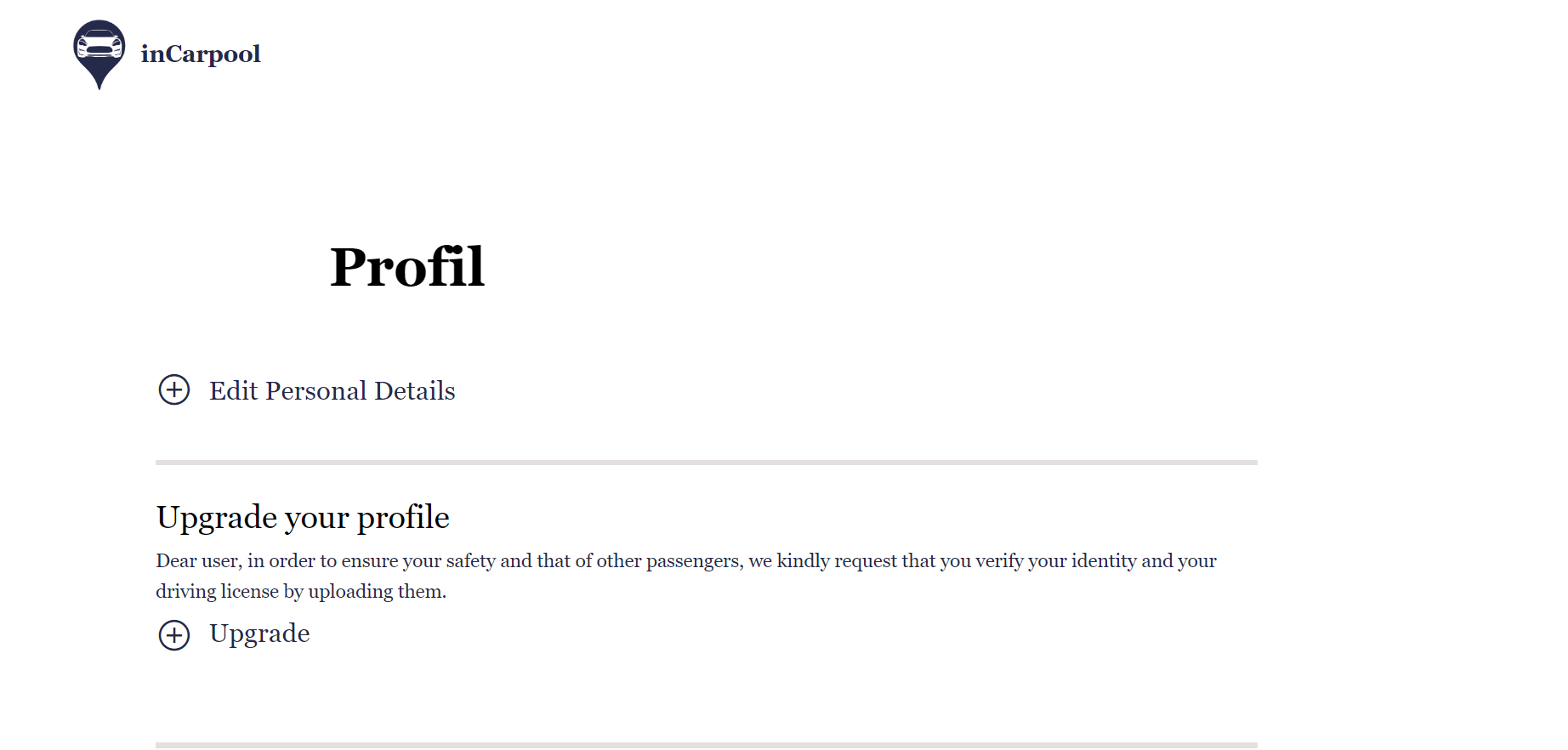




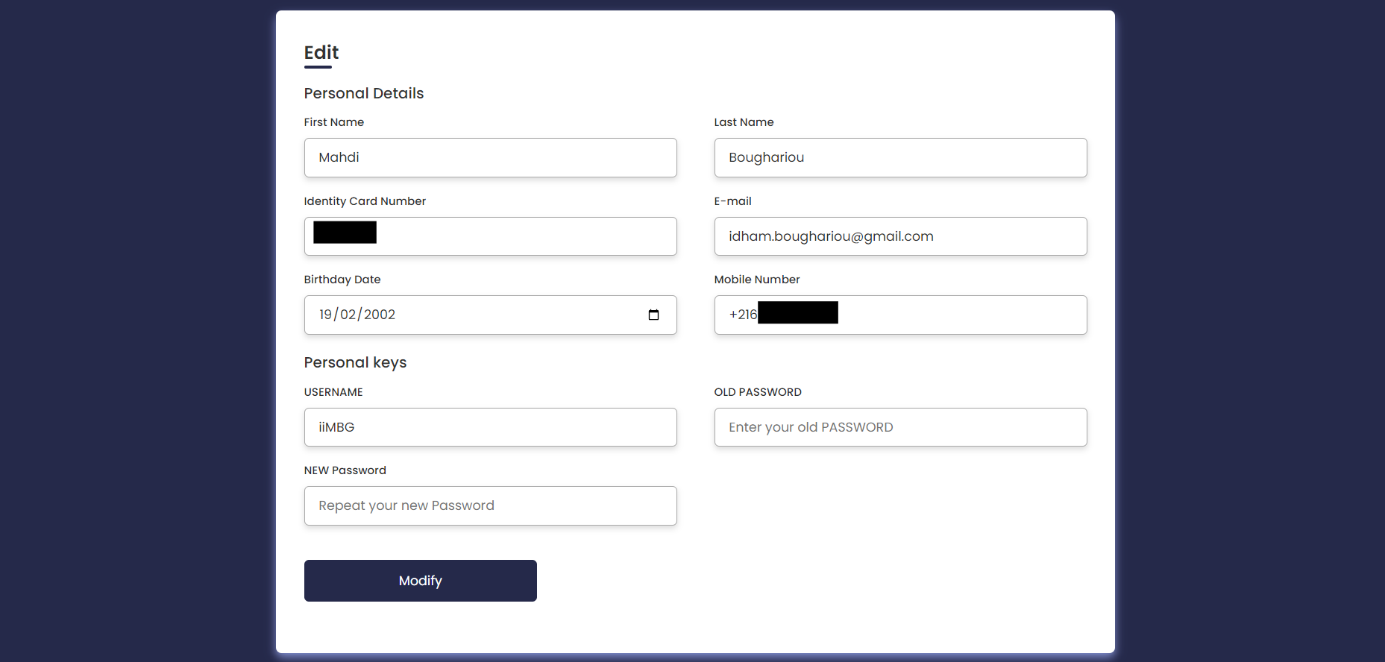


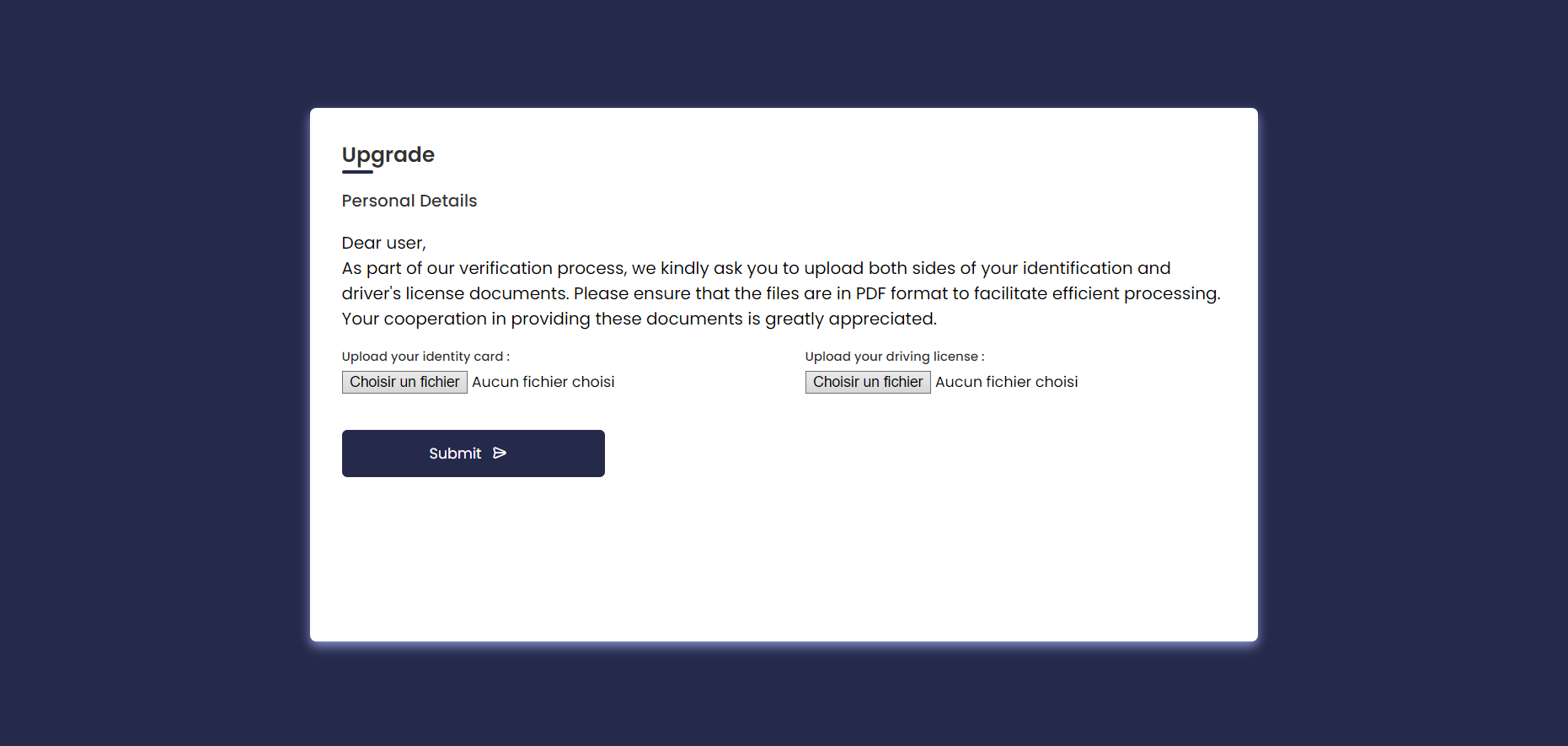


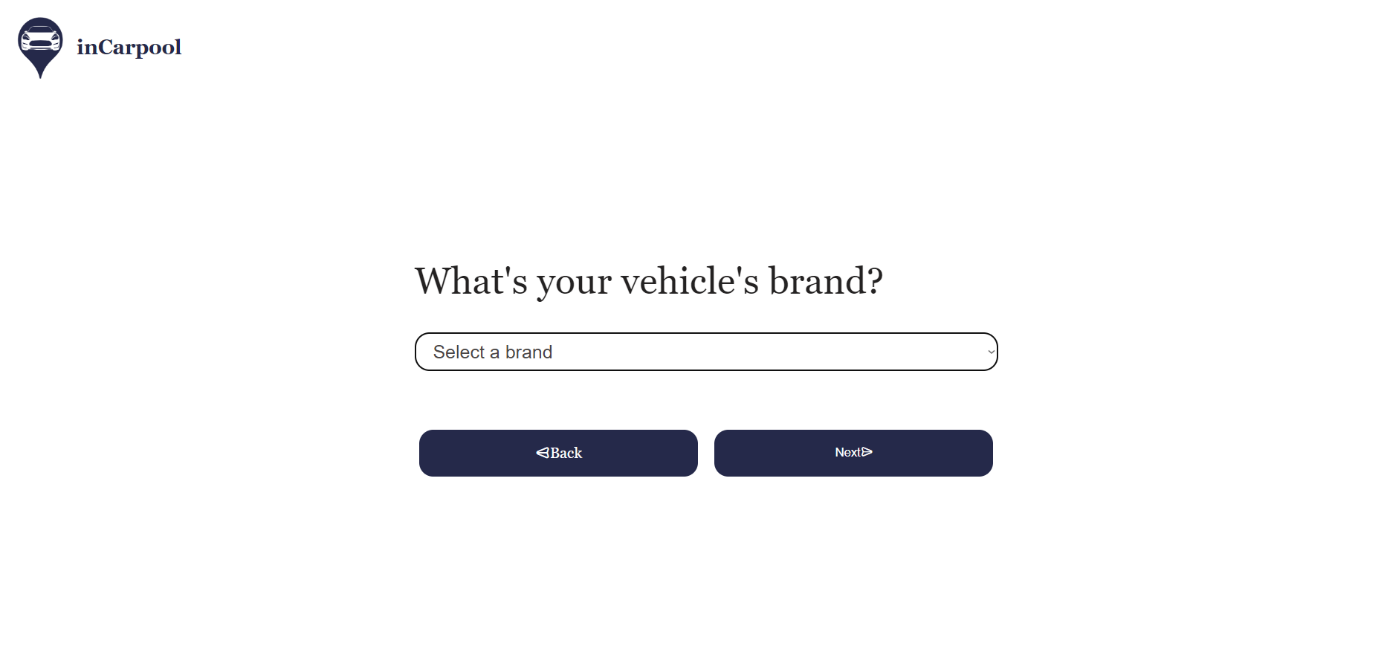


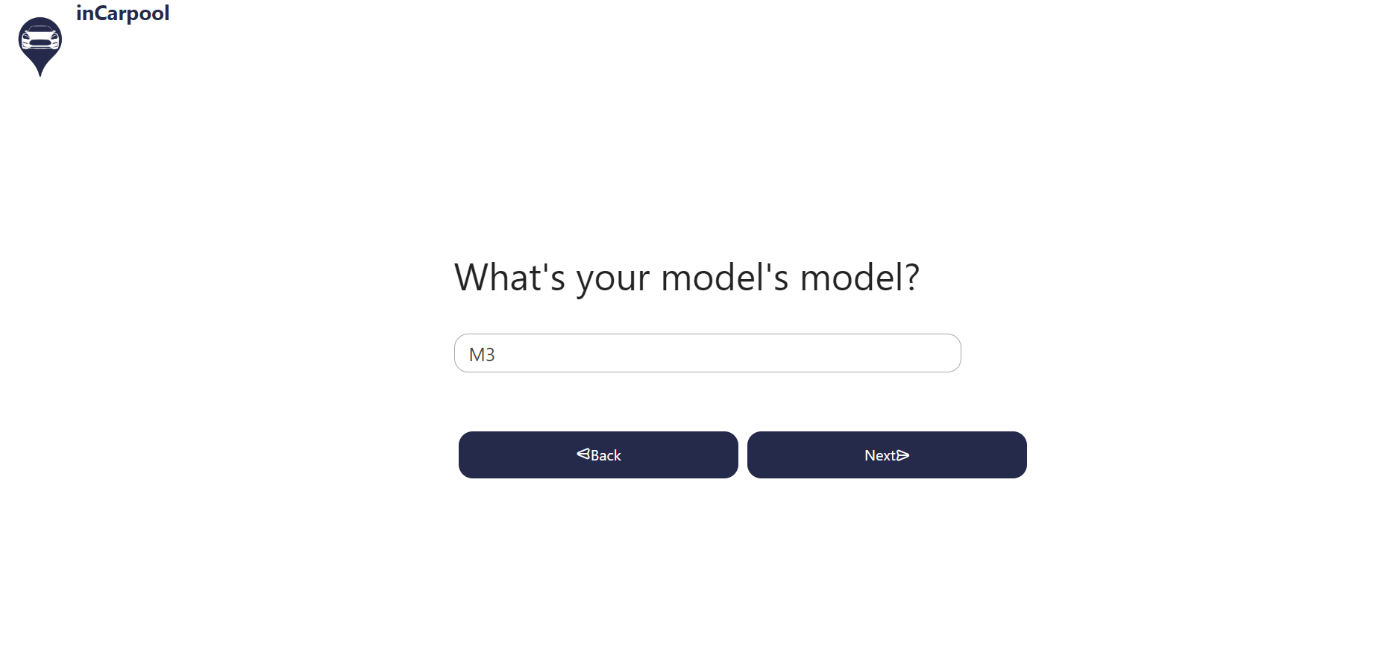


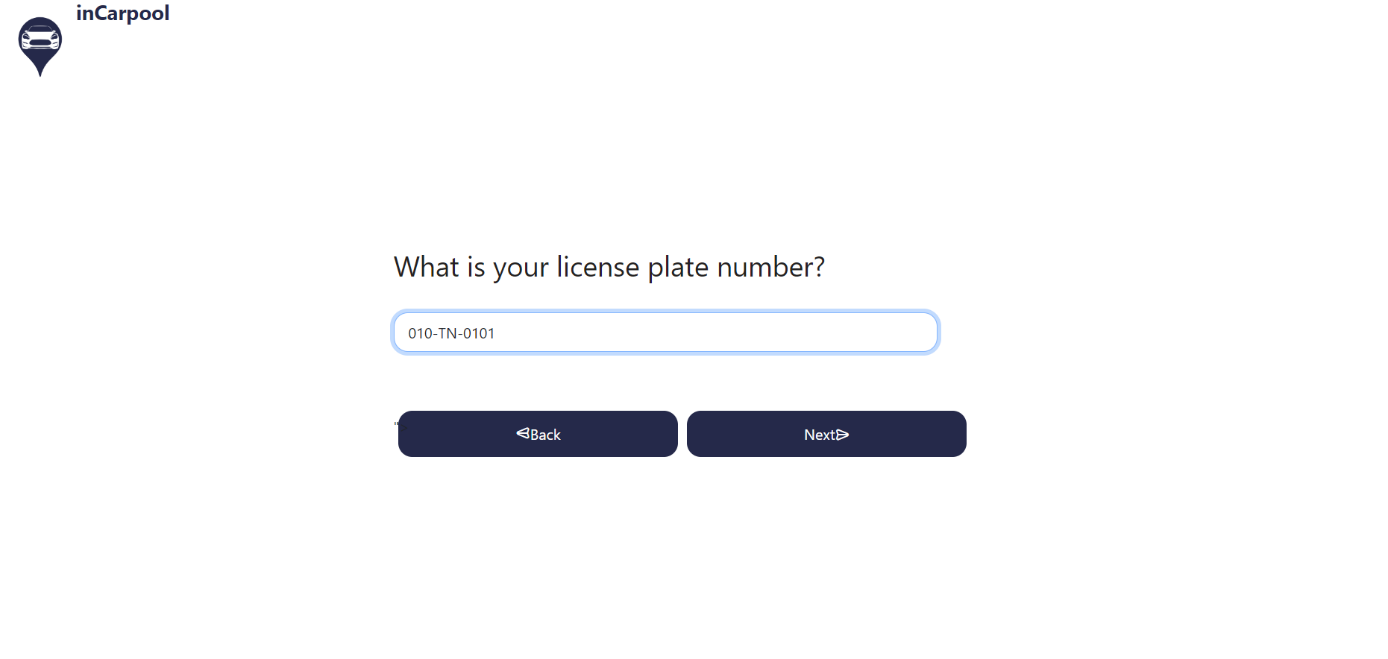


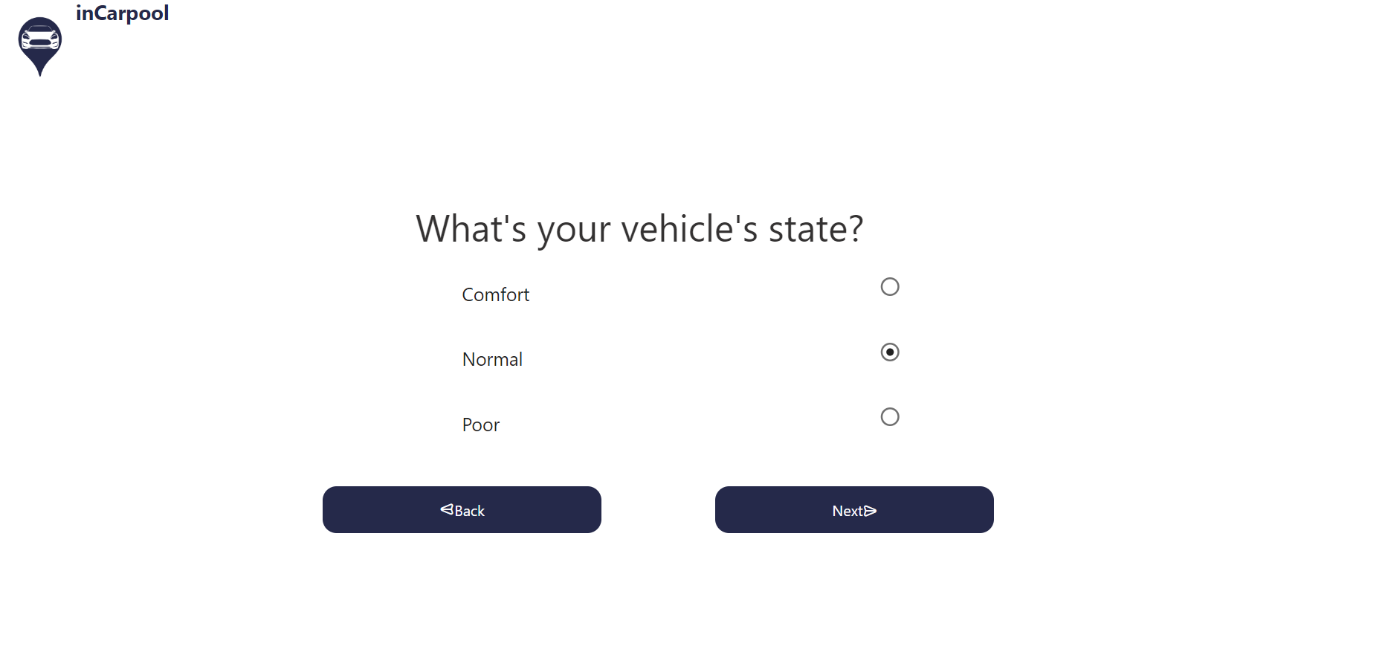


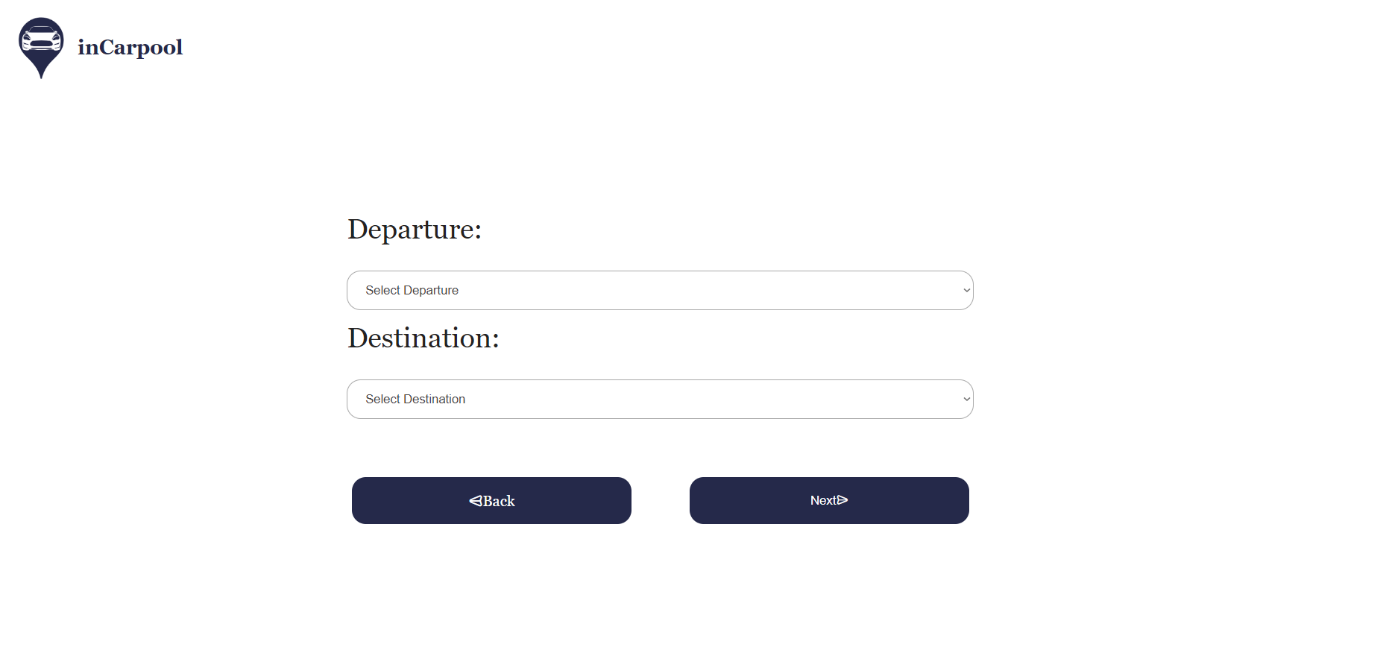


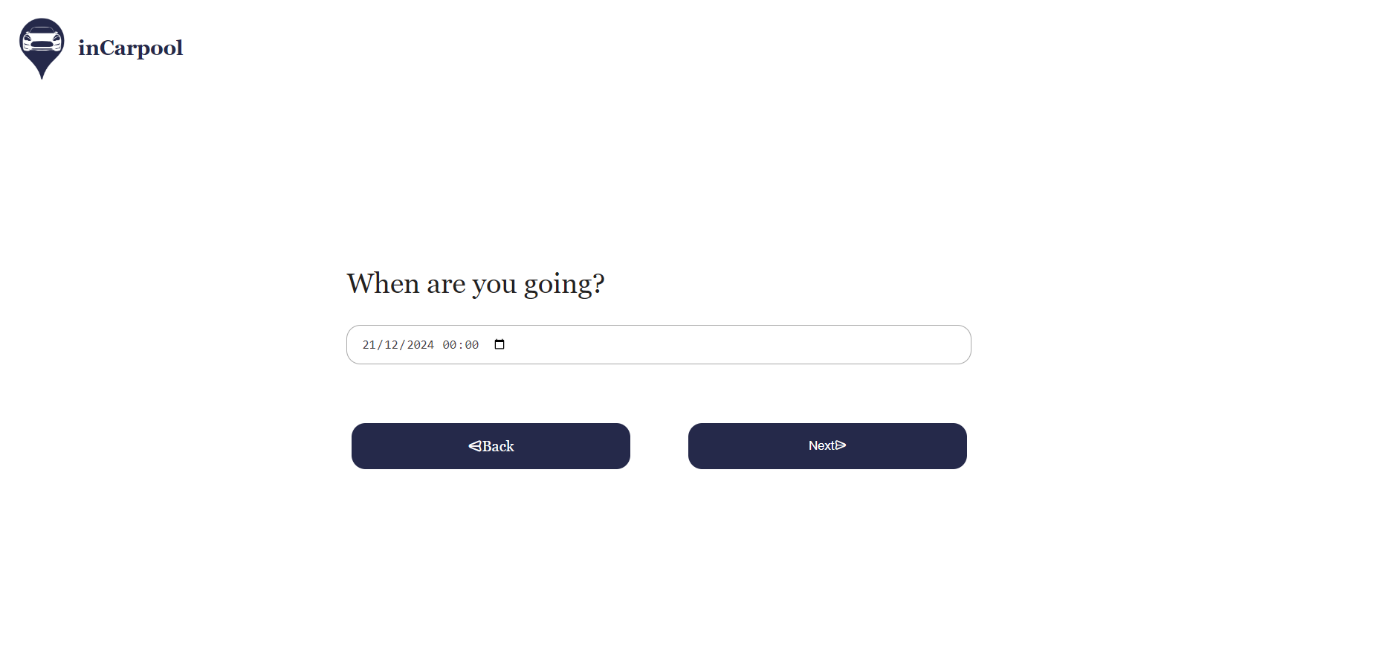




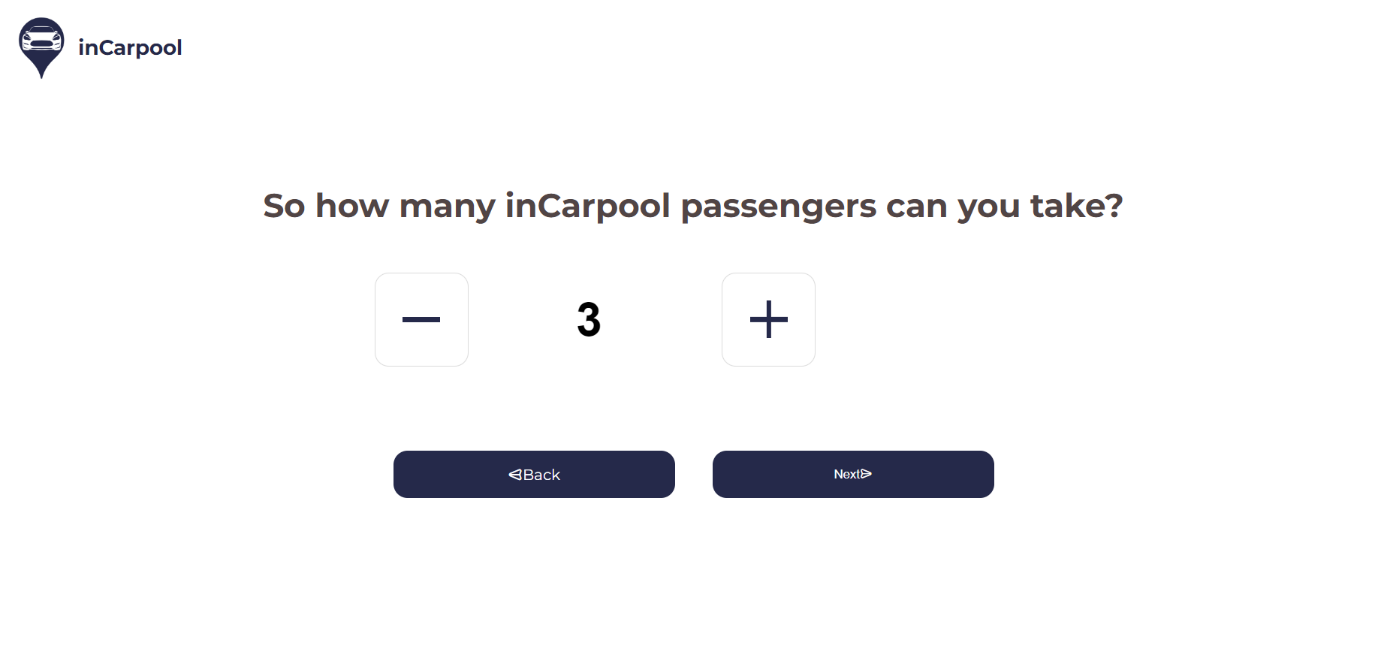


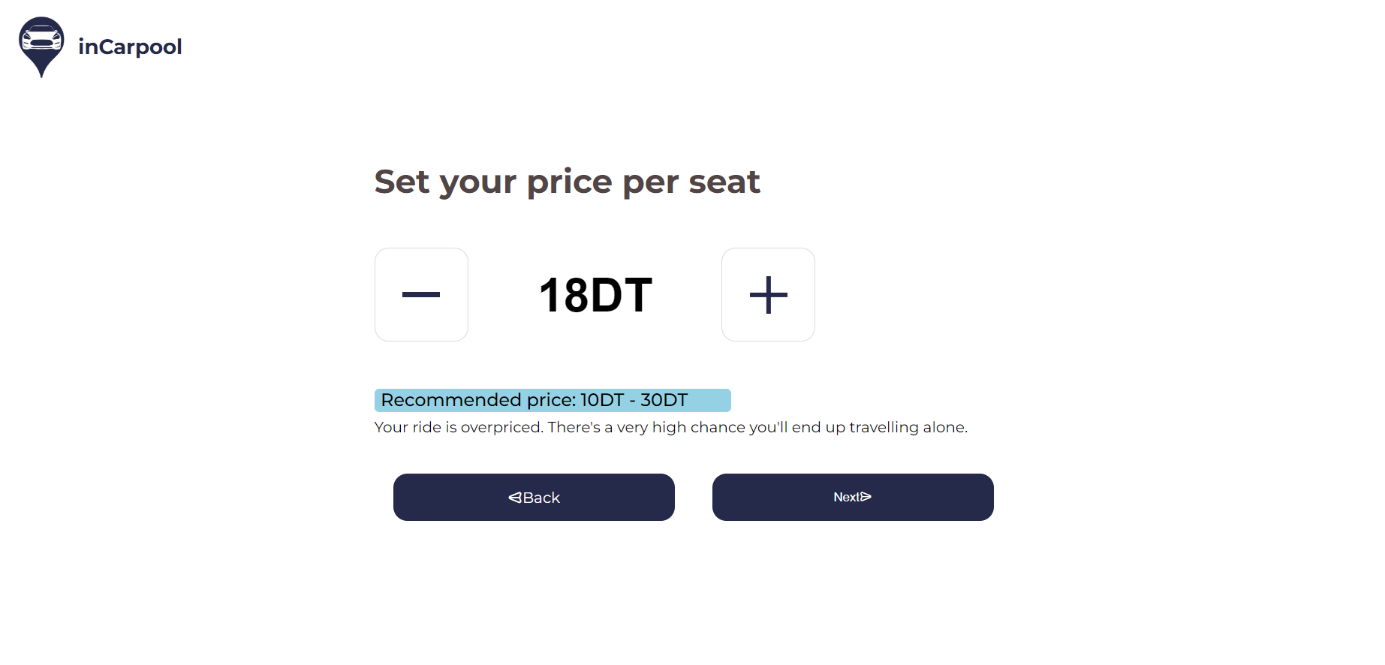


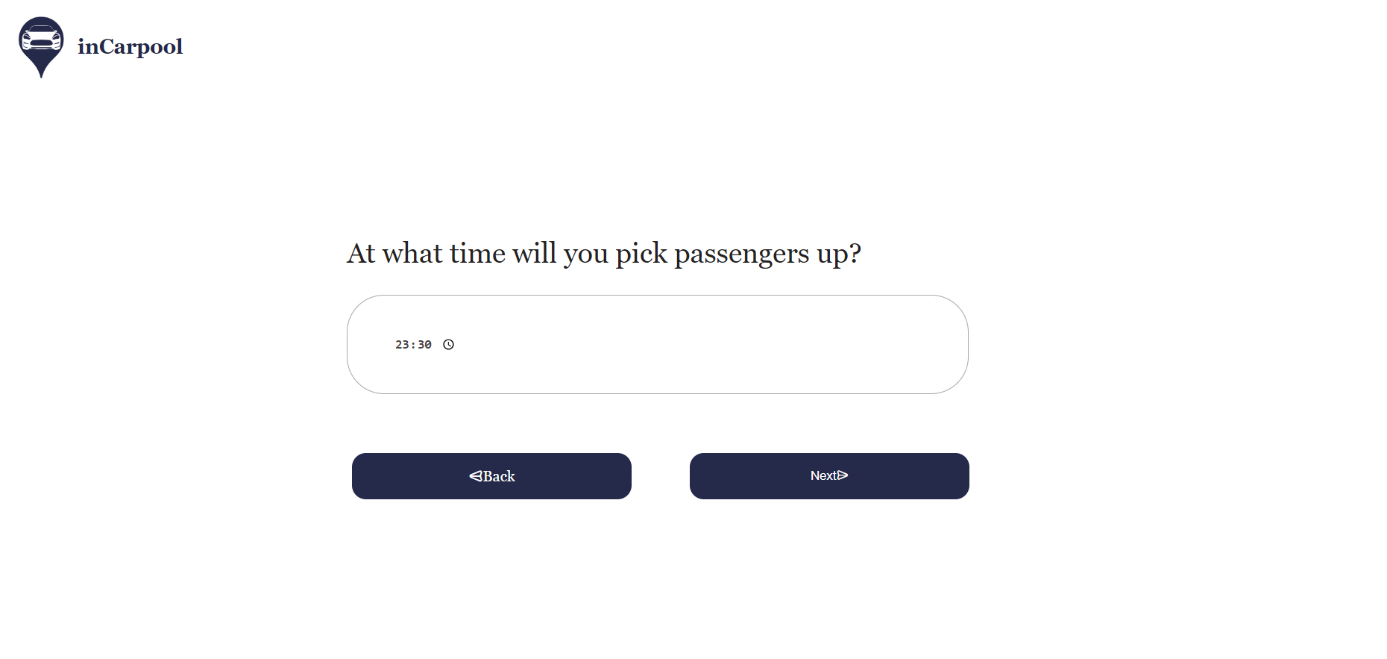


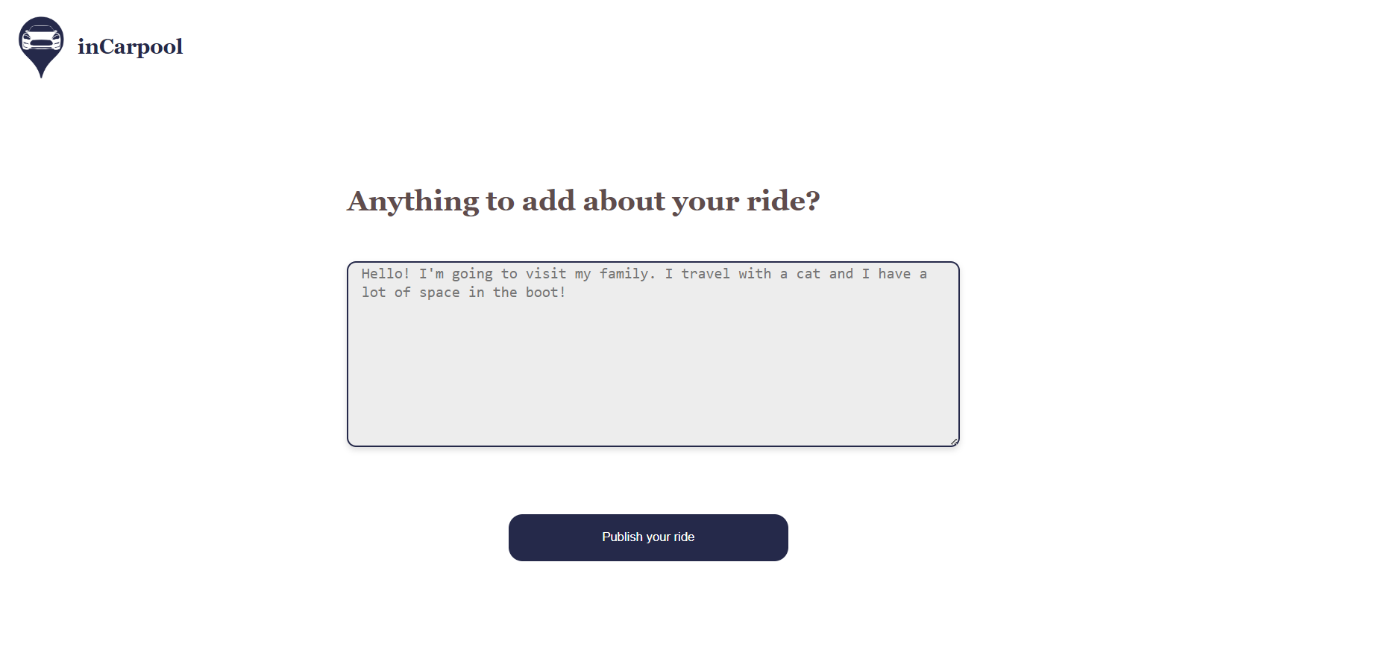


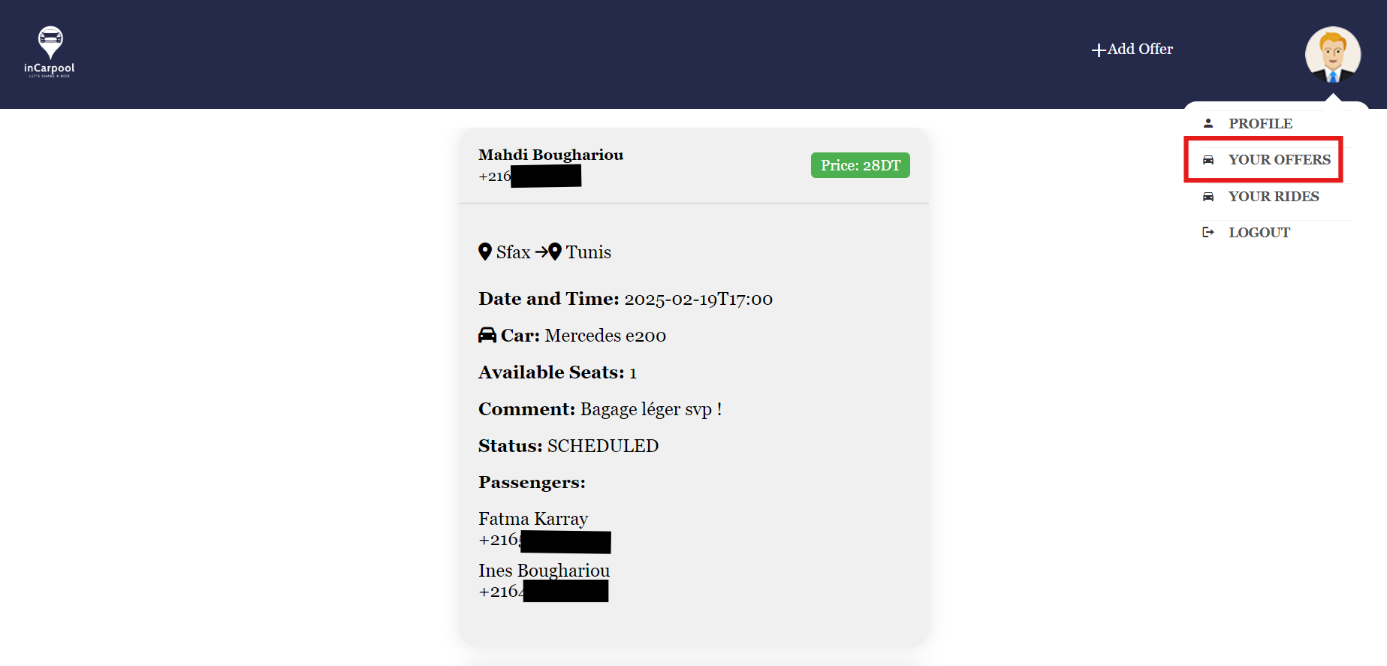


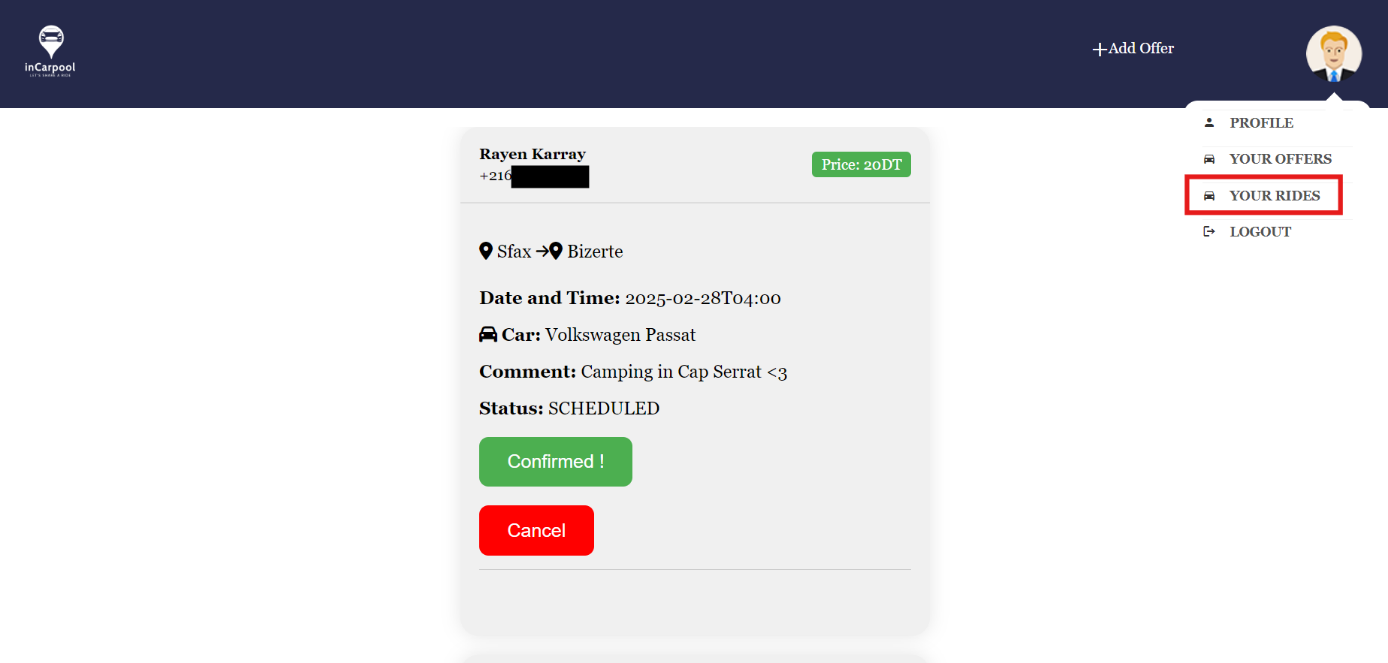












Docker :

