



DEVOIR DE STARTUP IT : Présentation détaillé du projet

Présenté par :

- **E4 CCSN : AMINA Amir Minihadji**
- **E4 CCSN : ALONGO Clara Divine Emmanuella**
- **E4 CCSN : DONFACK YMELE Errol Arthur**
- **E4 MBA DATA : HARZI Maram**
- **E4 CCSN : MOUJAHED Mohamed Amine**

«Le succès n'est pas le fruit du hasard, mais de l'ardeur au travail », Auguste Rodin

Promo 2024-2026

SOMMAIRE

INTRODUCTION	1
I. LISTE DES FONCTIONNALITES	3
A. CAS DU MVP	3
1. Inscription et authentification utilisateur	3
2. Gestion des profils utilisateurs	3
3. Création et réservation de trajets	4
4. Système d'évaluation et de notation	4
5. Filtrage et recherche de trajets	4
6. Moyen de communication entre les passagers et le conducteur	5
7. Système De Paiement	5
B. CAS DE LA V1	6
1. Notifications intelligentes	6
2. Partage de localisation en temps réel	6
3. Map interactive pour les trajets	6
4. Support en ligne intégré	6
5. Statistiques personnelles	7
C. CAS DE LA V_x	7
1. Covoiturage pour événements étudiants	7
2. Programme de fidélité	7
3. Covoiturage longue distance	8
4. Offres exclusives pour étudiants	8
5. Algorithme de correspondance intelligent	8
6. Mode hors ligne	8
7. Amélioration de la localisation en temps réel des conducteurs et passagers	9
8. Map sociale : points d'intérêt et communauté	9
II. PARCOURS UTILISATEUR	10
A. L'AUTHENTIFICATION	14
1. Écran d'accueil	15
2. Inscription	15
3. Connexion	17

4. Vérification du Profil	17
B. NAVIGATION.....	17
1. Tableau de bord.....	17
2. Menu principal	18
C. FONCTIONNALITES AU NIVEAU DU PASSAGER	18
D. FONCTIONNALITES AU NIVEAU DU CONDUCTEUR	21
E. INTERACTION ENTRE LES UTILISATEURS	22
F. PARAMETRES GENERAUX ET IMPORTANTS	22
III. MAQUETTE VISUELLE DE LA V1.....	23
SPLASH	24
ONBOARDING	25
AUTHENTIFICATION.....	26
ACCUEIL	29
SELECTION DES DEPART ET DES ARRIVEES.....	30
CONDUCTEUR	31
LES RECHERCHES PASSAGERS	32
CONTACT CONDUCTEUR – PASSAGER	33
AVIS CONDUCTEUR.....	35

TABLE DES ILLUSTRATIONS

Figure 1: parcours de l'utilisateur	11
Figure 2: cas d'utilisation global de l'application	12
Figure 3: Diagramme de cas d'utilisation "Espace personnel"	13
Figure 4: Architecture de l'application.	14
Figure 5: Diagramme d'activité "Inscription"	16
Figure 6: cas d'utilisation de la recherche de trajet	18
Figure 7: diagramme d'activité reservation passager	20
Figure 8: diagramme de Séquence "créer une annonce"	21

INTRODUCTION



Le projet que nous proposons vise à développer une application de covoiturage dédiée exclusivement aux étudiants en France. En réponse à l'augmentation constante des coûts de la vie, en particulier des frais de transport, ce service ambitionne de répondre à un besoin croissant d'une

solution de mobilité accessible, économique et durable, spécifiquement adaptée aux contraintes des étudiants.

Le problème majeur auquel ce projet répond est la précarité financière de nombreux étudiants, dont une part importante peine à couvrir les frais de transport pour se rendre à l'université ou sur leurs lieux de stage. Selon une étude menée par l'application Karos, le covoiturage étudiant a augmenté de 66 % en 2023, illustrant l'intérêt croissant pour cette solution. Cependant, malgré cet intérêt, l'accès au covoiturage reste souvent fragmenté, avec des plateformes généralistes qui ne répondent pas aux besoins spécifiques des étudiants (horaires flexibles, sécurité, etc.). Les étudiants, souvent très dépendants des transports en commun et confrontés à un manque d'alternatives abordables, se trouvent dans une situation où leurs déplacements sont sources de stress financier et logistique.

La solution que nous proposons consiste à créer une application dédiée qui facilite le covoiturage entre étudiants, leur permettant de partager des trajets de manière économique, sûre et efficace. Cette application permettra aux étudiants conducteurs de percevoir des rémunérations pour leurs trajets, tout en offrant aux passagers une solution de transport plus abordable que les alternatives classiques. En outre, elle intégrera des fonctionnalités de sécurité renforcée, comme la vérification des profils, et offrira des options de covoiturage en fonction des horaires et des lieux de résidence des étudiants. Le soutien de l'État à travers des subventions comme la prime de 100 euros pour les conducteurs débutants et le forfait mobilités durables renforcera l'attractivité de cette solution pour les étudiants (source : economie.gouv.fr).

Avec ce projet, nous visons à transformer les déplacements étudiants en une expérience plus écologique, plus sociale et surtout plus accessible. En optimisant le coût des trajets et en créant une communauté solidaire autour du covoiturage, cette solution permettra de lever un frein important à l'insertion professionnelle des jeunes, tout en réduisant l'empreinte carbone des transports étudiants (source : Le Monde).

Dans le cadre de cette étude détaillée, nous allons examiner les éléments clés nécessaires au développement de l'application de covoiturage pour étudiants. Nous définirons d'abord les fonctionnalités essentielles pour le MVP, puis pour les versions futures, tout en analysant le parcours utilisateur pour garantir une expérience fluide. Une maquette visuelle de la version 1 sera réalisée, et les choix technologiques seront justifiés pour assurer la performance et la compatibilité du projet. Enfin, nous proposerons l'architecture de la base de données, afin de structurer efficacement les informations des utilisateurs et des trajets.

I. LISTE DES FONCTIONNALITES

A. CAS DU MVP

1. Inscription et authentification utilisateur

Les utilisateurs doivent s'inscrire pour accéder à l'application. Pour ce faire il faudra :

- Utilisation des mails pour la connexion avec mot de passe. Utilisation de la possibilité de connexion via messagerie google
- Utilisation de numéros de téléphone pour l'authentification (via OTP, c'est-à-dire un code envoyé par SMS).
- Possibilité de déconnexion de l'application à tout moment.

2. Gestion des profils utilisateurs

Chaque utilisateur possède un profil avec des informations de base et son historique.

- Collecte des informations personnelles de base :
 - Nom et prénom.
 - Université ou école (non obligatoire mais encouragée).
- Ajout d'une photo de profil.
- Validation du statut étudiant :
 1. Les utilisateurs téléchargent une photo de leur carte étudiante ou tout autre justificatif (certificat de scolarité, inscription universitaire).
 2. Validation manuelle ou semi-automatique des documents par un système interne ou via un service tiers spécialisé en vérification d'identité.

3. Création et réservation de trajets

C'est la fonctionnalité clé pour connecter les conducteurs et passagers, rendant l'application immédiatement utile. Les conducteurs peuvent publier des trajets, et les passagers peuvent les réserver via l'application.

- ❖ Les conducteurs remplissent un formulaire simple pour détailler :
 - Les infos du véhicule
 - Point de départ et d'arrivée (adresse ou ville).
 - Date et heure du trajet.
 - Nombre de places disponibles.
 - Contribution financière demandée par passager.
- ❖ Les passagers sélectionnent un trajet et réservent une place.
- ❖ Une confirmation est envoyée à la fois au conducteur et au passager.

4. Système d'évaluation et de notation

Les conducteurs et passagers peuvent laisser des avis après un trajet, pour renforcer la confiance et améliorer l'expérience. La transparence renforce la confiance dans la communauté et incite les utilisateurs à adopter des comportements respectueux.

- Après chaque trajet, une notification invite à noter l'expérience (sur 5 étoiles) et à laisser un commentaire.
- Les profils affichent une moyenne de notes et les commentaires récents.

5. Filtrage et recherche de trajets

Cela garantit que les utilisateurs trouvent rapidement des trajets adaptés à leurs besoins,

rendant l'application pratique et efficace. Les utilisateurs doivent pouvoir rechercher des trajets selon leurs besoins (destination, horaire).

Les champs de recherches peuvent être basés sur :

- Ville ou adresse.
- Date du trajet.
- Place disponible.

6. Moyen de communication entre les passagers et le conducteur

Il Permet de régler les détails du trajet sans partager de coordonnées personnelles (favorisant la sécurité). De plus, il est Indispensable pour éviter les frictions dans l'organisation des trajets. A cet effet, il sera question de mettre en place :

- Système d'appel entre le le conducteur et le passager
- Messagerie interne simple pour permettre aux utilisateurs de communiquer directement après avoir sélectionné un trajet

7. Système De Paiement

Option 1 : Paiement en ligne avec un tiers

- Stripe et PayPal offrent des kits de développement faciles à intégrer.
- Carte bancaire, Visa, MasterCard, ...
- Un compte de séquestre pourrait être utilisé pour retenir l'argent jusqu'à la fin du trajet, garantissant une protection pour le passager et le conducteur.

Option 2 : Paiement en espèces

- Lors de la réservation, l'utilisateur choisit "Paiement en espèces".

- L'application affiche un rappel au passager de préparer le montant exact pour le trajet.

B. CAS DE LA V1

1. Notifications intelligentes

L'objectif est d'améliorer l'engagement et la satisfaction des utilisateurs en rendant Edudrive proactive et facile à utiliser. Les utilisateurs reçoivent des notifications push ou par email pour :

- Rappeler les trajets réservés.
- Suggérer des trajets correspondant à leurs recherches ou préférences.

2. Partage de localisation en temps réel

Les conducteurs peuvent partager leur position en direct avec les passagers, facilitant les points de rencontre. Cela Renforce la coordination et la sécurité entre conducteurs et passagers.

3. Map interactive pour les trajets

Une carte intégrée permet aux utilisateurs de visualiser leur trajet planifié, avec les points de départ et d'arrivée clairement indiqués. Les passagers peuvent également voir les arrêts intermédiaires lorsqu'ils partagent leur trajet avec d'autres utilisateurs. Cela Offre une meilleure compréhension des trajets pour les utilisateurs. De plus, cela Simplifie la coordination entre passagers et conducteurs en rendant visuels les points de rencontre.

4. Support en ligne intégré

Un support rapide et réactif améliore la satisfaction des utilisateurs et renforce la crédibilité de l'application. Une section dédiée pour répondre aux questions et résoudre les problèmes rapidement. On peut avoir par exemple :

- Ajout d'un système de chat (bot ou agent humain).

- Création d'une FAQ couvrant les questions courantes (paiements, trajets, profils).

5. Statistiques personnelles

Les utilisateurs peuvent consulter des données sur leurs trajets : kilomètres parcourus, économies réalisées, nombre de trajets effectués, etc.

- Développement d'un tableau de bord intégré dans les profils utilisateurs.
- Calcul automatique des statistiques à partir des données des trajets.

Cela motive les utilisateurs à utiliser l'application régulièrement en leur montrant des avantages concrets.

C. CAS DE LA V_x

1. Covoiturage pour événements étudiants

Fonctionnalité permettant d'organiser des trajets groupés pour des événements étudiants (galas, conférences, soirées). L'idée ici est de diversifier les cas d'usage de l'application et attirer de nouveaux utilisateurs. On a pensé pour cela à :

- Collaboration avec les organisateurs pour promouvoir l'application.
- Interface spécifique pour publier des trajets liés à un événement.

2. Programme de fidélité

L'objectif est de fidéliser les utilisateurs en ajoutant une dimension ludique et avantageuse. Un système de points et de récompenses pour encourager les utilisateurs actifs (trajets fréquents, parrainage). La mise en place peut se faire grâce à :

- Accumulation de points à chaque trajet effectué ou utilisateur parrainé.
- Points échangeables contre des réductions ou cadeaux (bons d'achat, carburant).

3. Covoiturage longue distance

Cela permettra d'étendre la portée de l'application et répond aux besoins fréquents des étudiants. Cette Fonctionnalité pour des trajets entre différentes régions de France, particulièrement pour les retours en famille ou les vacances. On pourra par exemple établir des filtres spécifiques pour les trajets longue distance ainsi qu'une Gestion des arrêts intermédiaires.

4. Offres exclusives pour étudiants

Collaboration avec des partenaires pour offrir des avantages exclusifs aux utilisateurs (réductions sur l'essence, bons cadeaux, etc.). l'objectif est de renforcer la pertinence d'EduDrive comme une plateforme dédiée aux étudiants.

- Développement d'un programme de partenariat.
- Promotion des offres directement sur l'application.

5. Algorithme de correspondance intelligent

Un système qui suggère les meilleurs trajets pour chaque utilisateur en fonction de leurs préférences et habitudes. La mise en place se fera par Machine Learning pour analyser les données utilisateur (préférences, historiques). On pourra aussi avoir une Interface intuitive pour afficher les suggestions. Cela Simplifie encore plus l'expérience utilisateur et personnalise l'application selon les besoins individuels.

6. Mode hors ligne

Ce mode Permet aux utilisateurs de consulter et de préparer leurs trajets même sans connexion Internet. On peut avoir un Téléchargement des informations essentielles avant le départ (trajets réservés, coordonnées). De plus, on peut avoir une Synchronisation automatique des données lorsque la connexion est rétablie. L'objectif est de renforcer la fiabilité de l'application dans toutes les conditions, notamment en déplacement.

7. Amélioration de la localisation en temps réel des conducteurs et passagers

Les conducteurs et passagers peuvent partager leur position en temps réel sur la carte, permettant de suivre le trajet en direct et de localiser facilement les points de rencontre. Cela Améliore la coordination en rendant visibles les mouvements en temps réel et enforce la sécurité en permettant aux passagers et conducteurs de se localiser facilement. L'accent sera de rendre la Map plus fiable tant au niveau de la fonctionnalité que tu design avec une sorte de copie de la map comme Sur NSAPTCHAT avec des avatars

- **Mise en place :**

- Utilisation des services de géolocalisation en temps réel (via GPS) pour afficher la position dynamique des utilisateurs sur la map.
- Fonctionnalité activable pour les trajets en cours uniquement, afin de protéger la vie privée.
- Affichage des avatars ou icônes personnalisés pour les utilisateurs (comme sur Snapchat).

8. Map sociale : points d'intérêt et communauté

En plus des trajets, la carte pourra afficher des points d'intérêt pour les étudiants (cafés étudiants, bibliothèques, événements locaux) et intégrer des fonctionnalités sociales. On notera Collaboration avec des tiers pour identifier et afficher des points d'intérêt. Ajout d'une option pour créer des rendez-vous ou événements directement depuis la carte. Pourquoi c'est important ? parce que cela Transforme Edudrive en une application conviviale au-delà du covoiturage, en renforçant son utilité dans la vie quotidienne des étudiants.

II. PARCOURS UTILISATEUR

Le parcours utilisateur, ou "User Journey", est une représentation détaillée des étapes qu'un utilisateur suit pour atteindre un objectif spécifique lorsqu'il interagit avec un produit ou un service. Il s'agit d'un outil essentiel dans la conception centrée sur l'utilisateur (UX design), car il permet de visualiser non seulement les interactions de l'utilisateur avec l'interface, mais également ses pensées, ses besoins et ses attentes à chaque étape du processus.

Dans le cadre de notre application de covoiturage pour étudiants, le parcours utilisateur vise à simplifier et fluidifier l'expérience, depuis l'ouverture de l'application jusqu'à l'accomplissement de la tâche recherchée, qu'il s'agisse de réserver un trajet, de proposer une place de covoiturage ou de gérer son profil utilisateur. En définissant clairement les étapes critiques et les conditions associées à chaque interaction, nous pouvons non seulement identifier les points de friction potentiels, mais aussi améliorer continuellement la satisfaction et la fidélité des utilisateurs.

Le parcours utilisateur inclut des étapes clés telles que la connexion/inscription, la navigation dans l'interface, l'utilisation des fonctionnalités principales comme la création ou la réservation de trajets, et l'interaction entre conducteurs et passagers. Un processus bien conçu permet de guider efficacement l'utilisateur, en assurant une expérience intuitive, sécurisée et agréable.

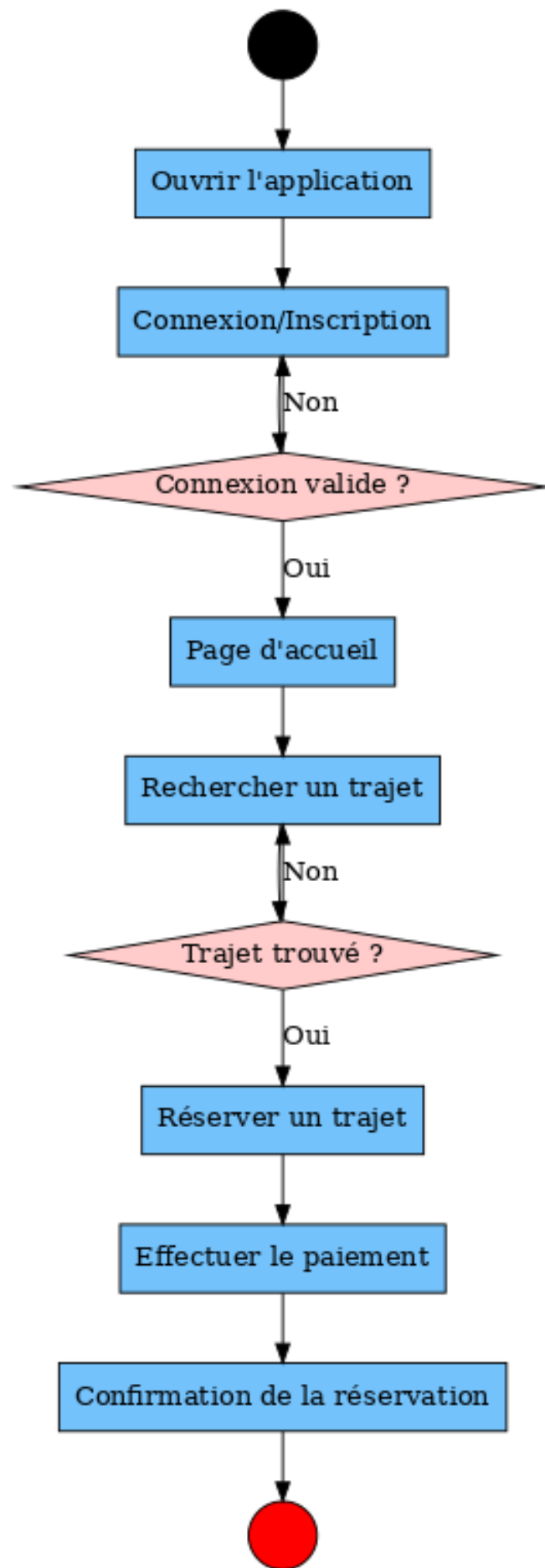


Figure 1: parcours de l'utilisateur

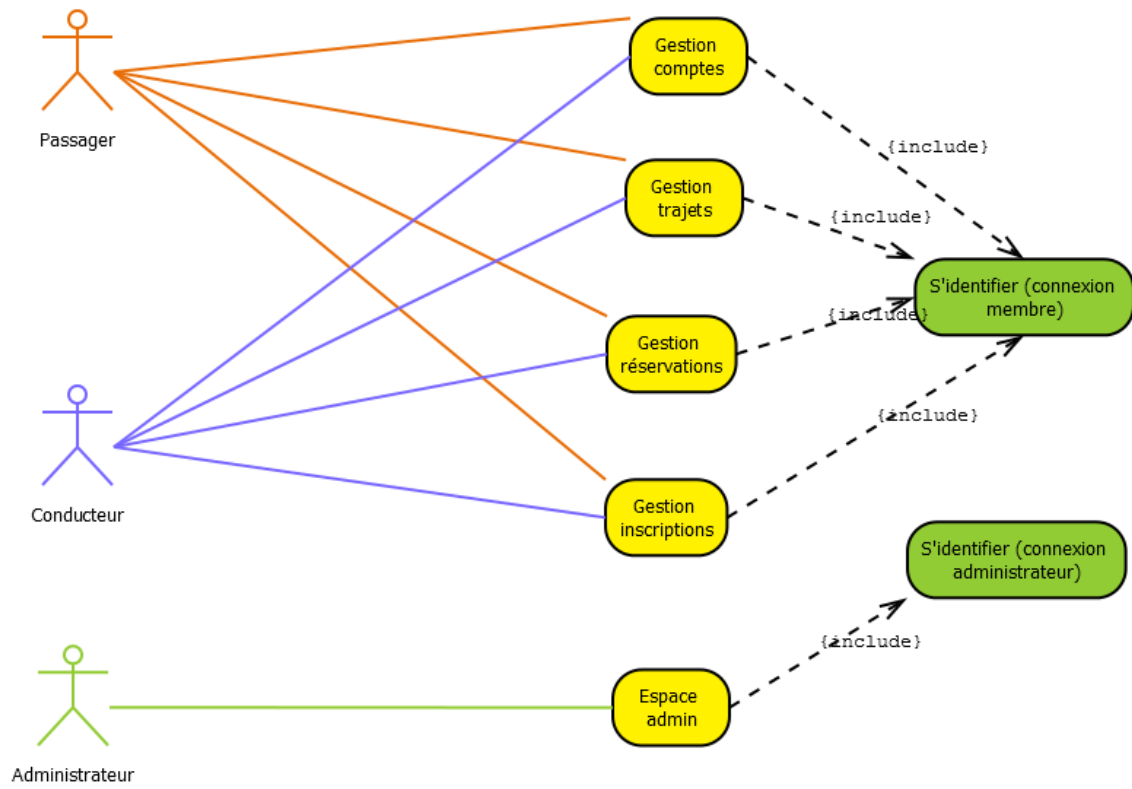


Figure 2: cas d'uitlisation global de l'application

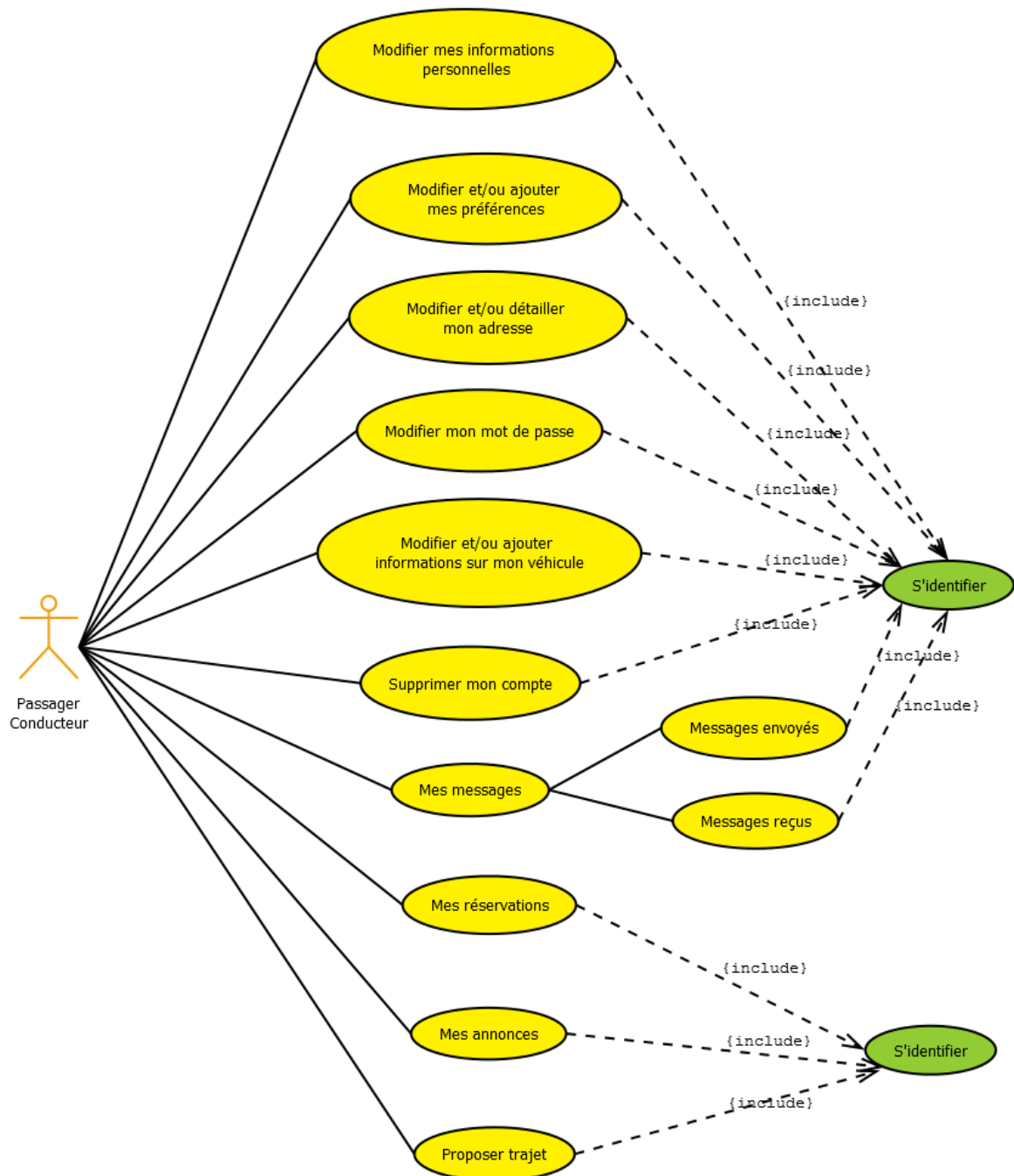


Figure 3: Diagramme de cas d'utilisation "Espace personnel"

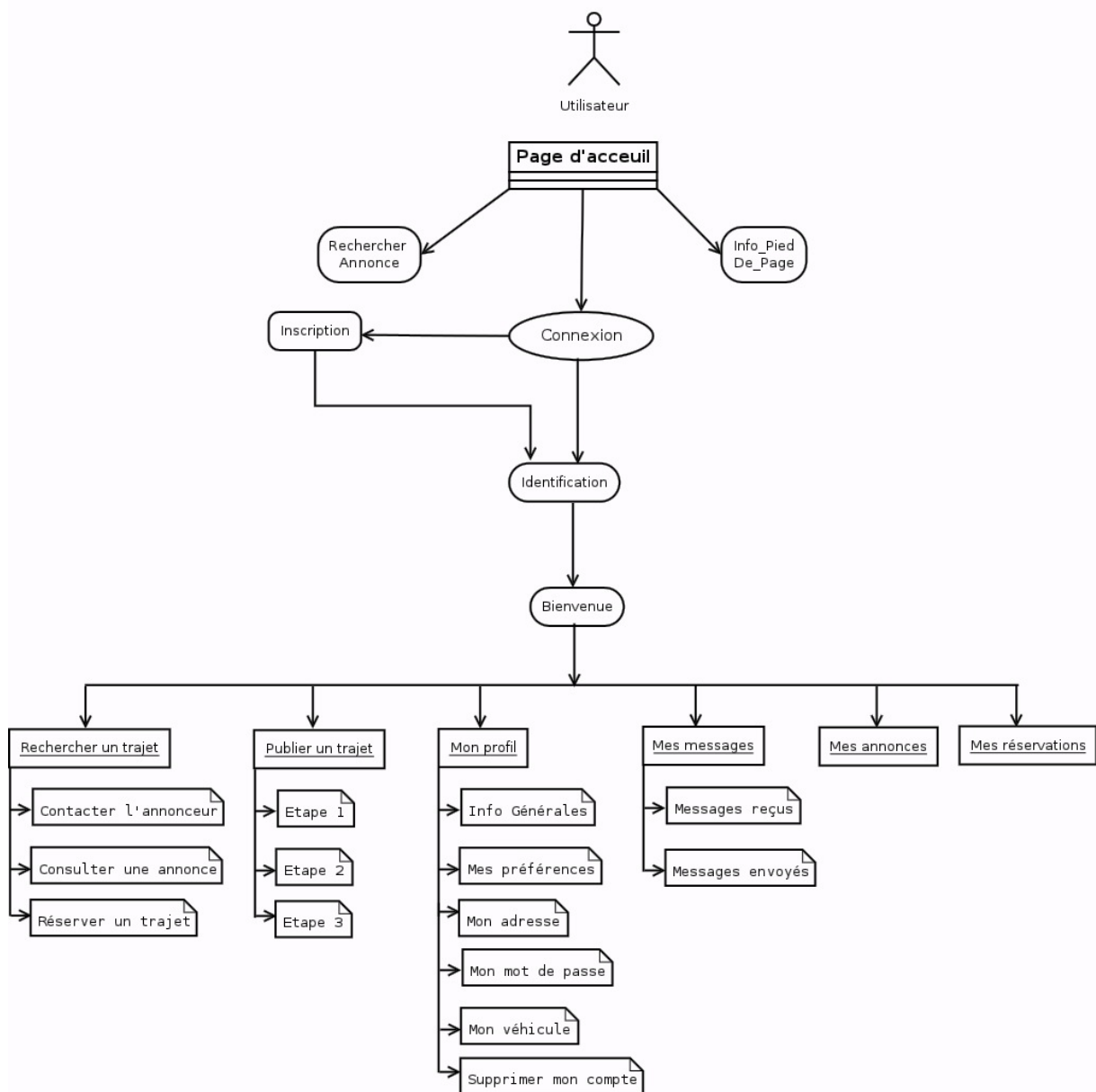


Figure 4: Architecture de l'application.

A. L'AUTHENTIFICATION

L'authentification est une étape cruciale dans le fonctionnement de l'application de covoiturage, car elle permet de sécuriser l'accès et de vérifier l'identité des utilisateurs, tout en facilitant leur expérience. Cette phase est divisée en plusieurs étapes importantes, couvrant à

la fois la connexion et l'inscription des utilisateurs, ainsi que la validation des documents et profils.

1. Écran d'accueil

L'utilisateur est d'abord accueilli sur un écran qui lui propose deux options principales :

- **Créer un compte** (pour les nouveaux utilisateurs)
- **Se connecter** (pour les utilisateurs existants)

2. Inscription

Les nouveaux utilisateurs doivent passer par un processus d'inscription, qui comprend :

- **Choix du mode d'inscription** : via un formulaire classique (prénom, nom, email, numéro de téléphone, mot de passe) ou une méthode rapide par connexion à un compte Google.
- **Vérification de l'identité étudiante** : Une preuve d'inscription dans un établissement (carte étudiante ou certificat d'inscription) est demandée.
- **Téléchargement des documents** : L'utilisateur soumet ses documents pour validation.
- **Validation des documents** : Les documents sont vérifiés manuellement par un administrateur ou automatiquement par une intelligence artificielle. Si les documents sont invalides, l'utilisateur reçoit une notification pour corriger les informations ou soumettre de nouveaux documents.
- **Confirmation d'inscription** : Après validation, l'utilisateur reçoit une confirmation par email ou notification, et son compte est activé.

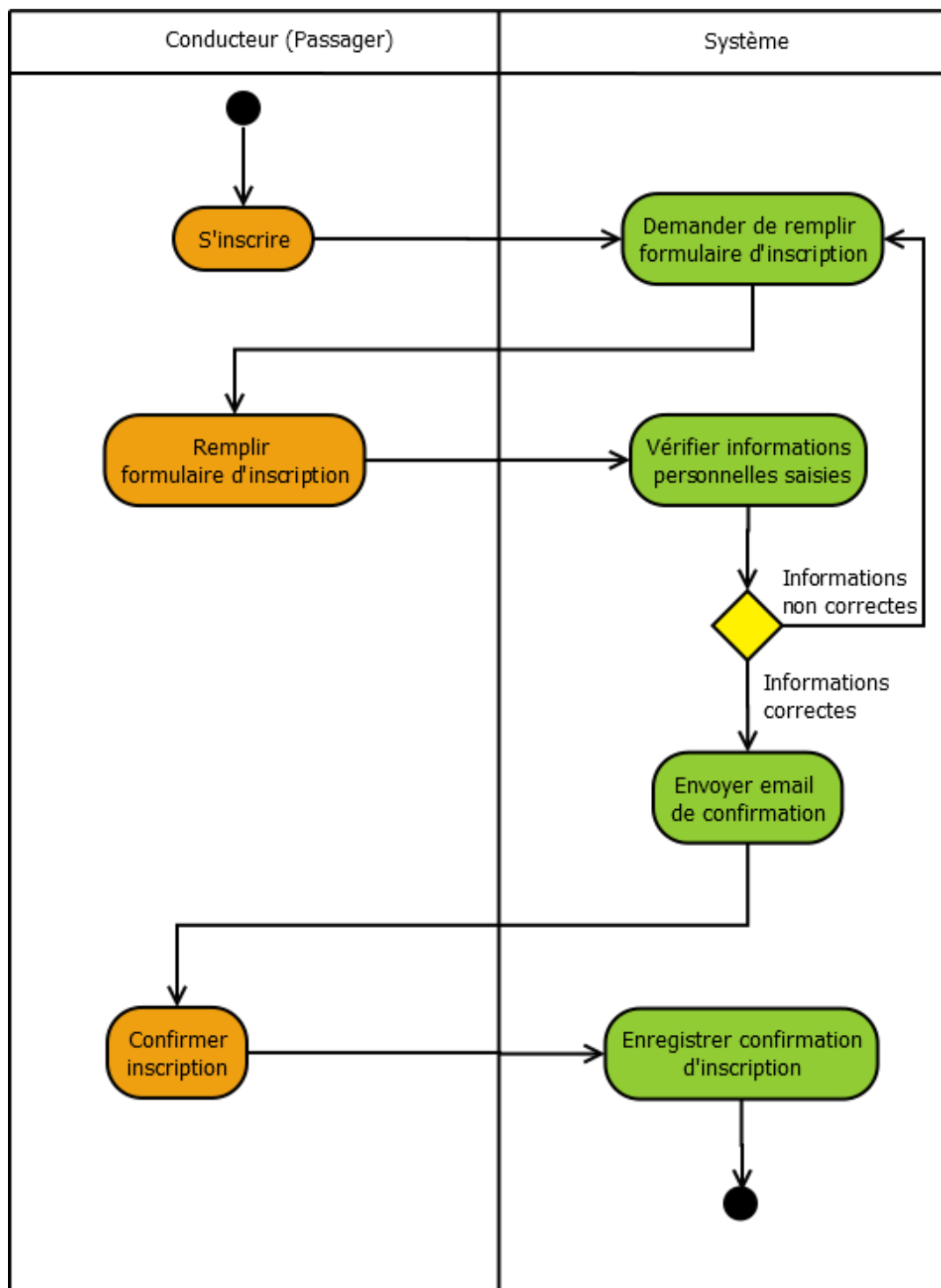


Figure 5: Diagramme d'activité "Inscription"

3. Connexion

Les utilisateurs déjà inscrits peuvent se connecter de manière sécurisée via plusieurs options :

- **Connexion par email/mot de passe** : L'utilisateur entre ses informations classiques de connexion.
- **Connexion rapide** : Grâce à des services externes comme Google, ou via des méthodes biométriques (empreinte digitale, reconnaissance faciale) pour une expérience plus rapide.
- **Récupération de mot de passe** : En cas d'oubli, une option permet de récupérer son mot de passe via un email ou un SMS.

4. Vérification du Profil

Une fois connecté, l'utilisateur accède à son profil qui contient ses informations personnelles. Si la vérification étudiante est en attente, l'accès aux fonctionnalités de covoiturage (création de trajets ou réservation) peut être restreint.

- **Cas valide** : Si le profil est validé, l'utilisateur peut accéder pleinement à l'application.
- **Cas invalide** : Si la vérification des documents échoue, l'utilisateur est informé et doit soumettre de nouveaux documents pour être validé.

B. NAVIGATION

La navigation dans l'application repose sur un tableau de bord simplifié et une barre de menu fixe, assurant un accès facile aux différentes sections.

1. Tableau de bord

Après l'authentification, l'utilisateur est redirigé vers l'**écran d'accueil**, où deux principales actions sont mises en avant :

- **Trouver un trajet** (pour les passagers)
- **Proposer un trajet** (pour les conducteurs)

L'utilisateur voit également des informations supplémentaires comme :

- **Statistiques rapides** : Nombre de trajets effectués, économies réalisées, distance parcourue, etc.

2. Menu principal

Une **barre de navigation** fixe située en bas ou en haut de l'écran propose des onglets pour accéder rapidement aux sections suivantes :

- **Accueil** : Retour à l'écran principal.
- **Mes trajets** : Consultation des trajets réservés ou proposés, passés ou à venir.
- **Profil** : Accès aux informations personnelles et paramètres de compte.
- **Support** : Assistance technique, FAQ, et centre d'aide.

C. FONCTIONNALITES AU NIVEAU DU PASSAGER

Les utilisateurs cherchant à réserver un trajet bénéficient d'un processus simple et intuitif.

1. Recherche de trajets

- L'utilisateur entre les informations dans le **champ de recherche** : destination, date de départ, budget maximum, etc.
- **Liste des trajets disponibles** : Triée par pertinence (proximité, avis du conducteur, prix, etc.).

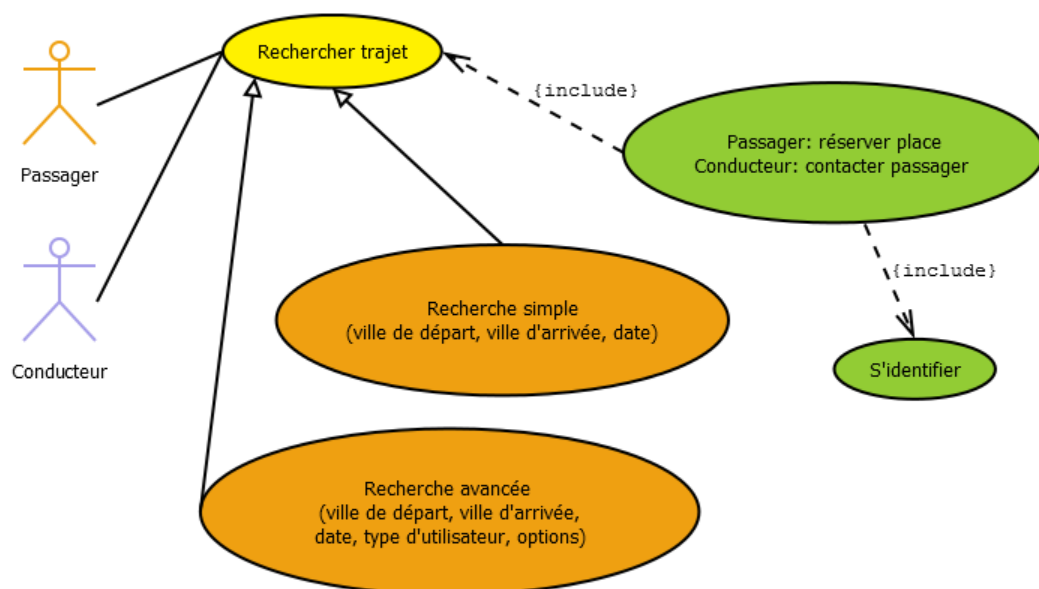


Figure 6: cas d'utilisation de la recherche de trajet

2. Détails du trajet

- Lorsqu'un trajet est sélectionné, l'utilisateur accède à des informations détaillées :
 - Nom et photo du conducteur
 - Avis laissés par d'autres passagers
 - Détails du véhicule
 - Lieu de départ exact (adresse ou point de rencontre)

3. Réservation du trajet

- **Paiement en ligne** via des solutions comme Stripe, PayPal, ou en espèces (si le conducteur accepte).
- Une **confirmation** est envoyée par email et notification une fois la réservation validée.

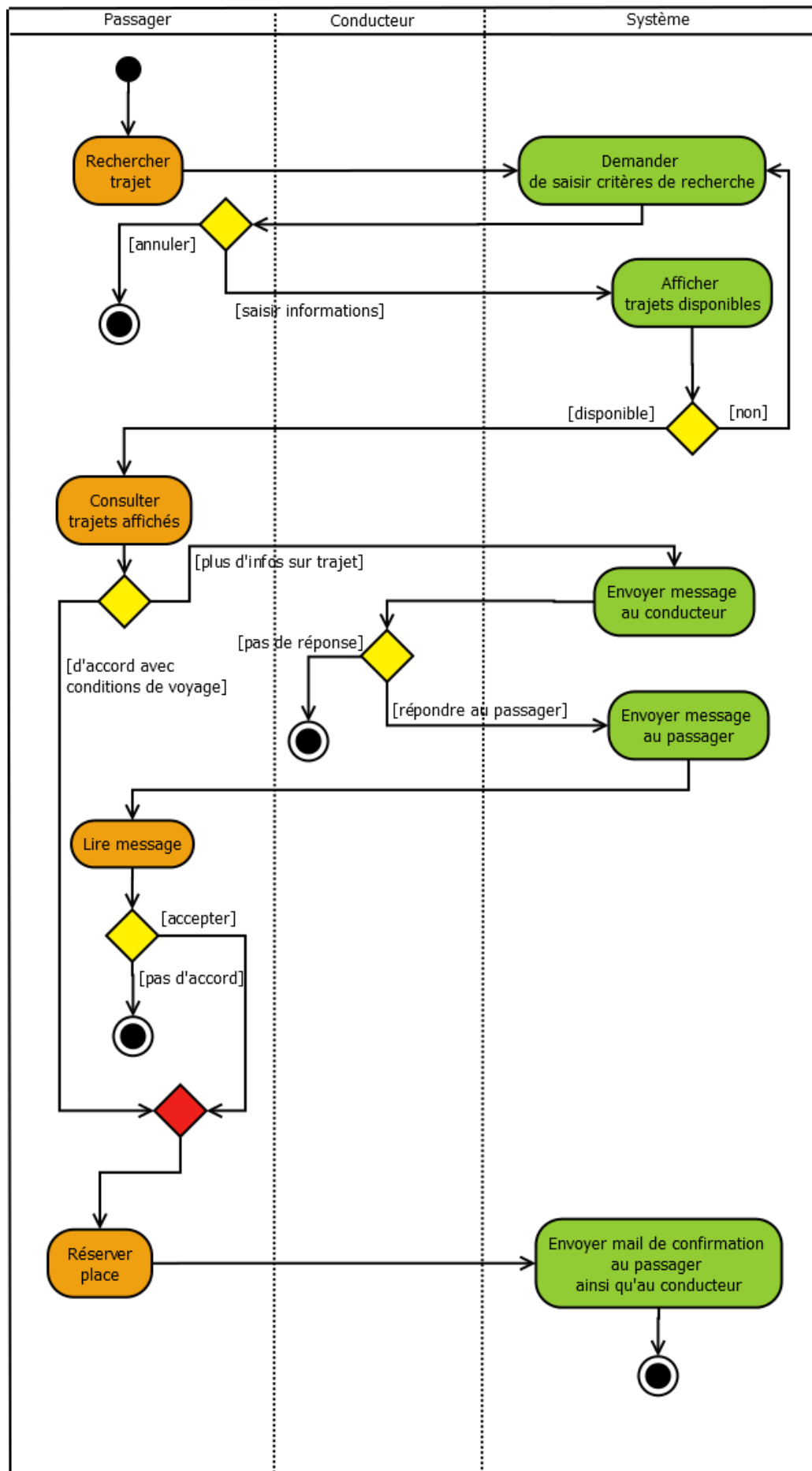


Figure 7: diagramme d'activité reservation passager

D. FONCTIONNALITES AU NIVEAU DU CONDUCTEUR

Les conducteurs peuvent facilement proposer des trajets, gérer les réservations, et interagir avec leurs passagers.

1. Création d'un trajet

- **Étapes guidées** : Le conducteur entre les détails du trajet, tels que les points de départ et d'arrivée, la date, l'heure, et le nombre de places disponibles.
- **Tarification** : Le conducteur fixe une contribution financière par passager.

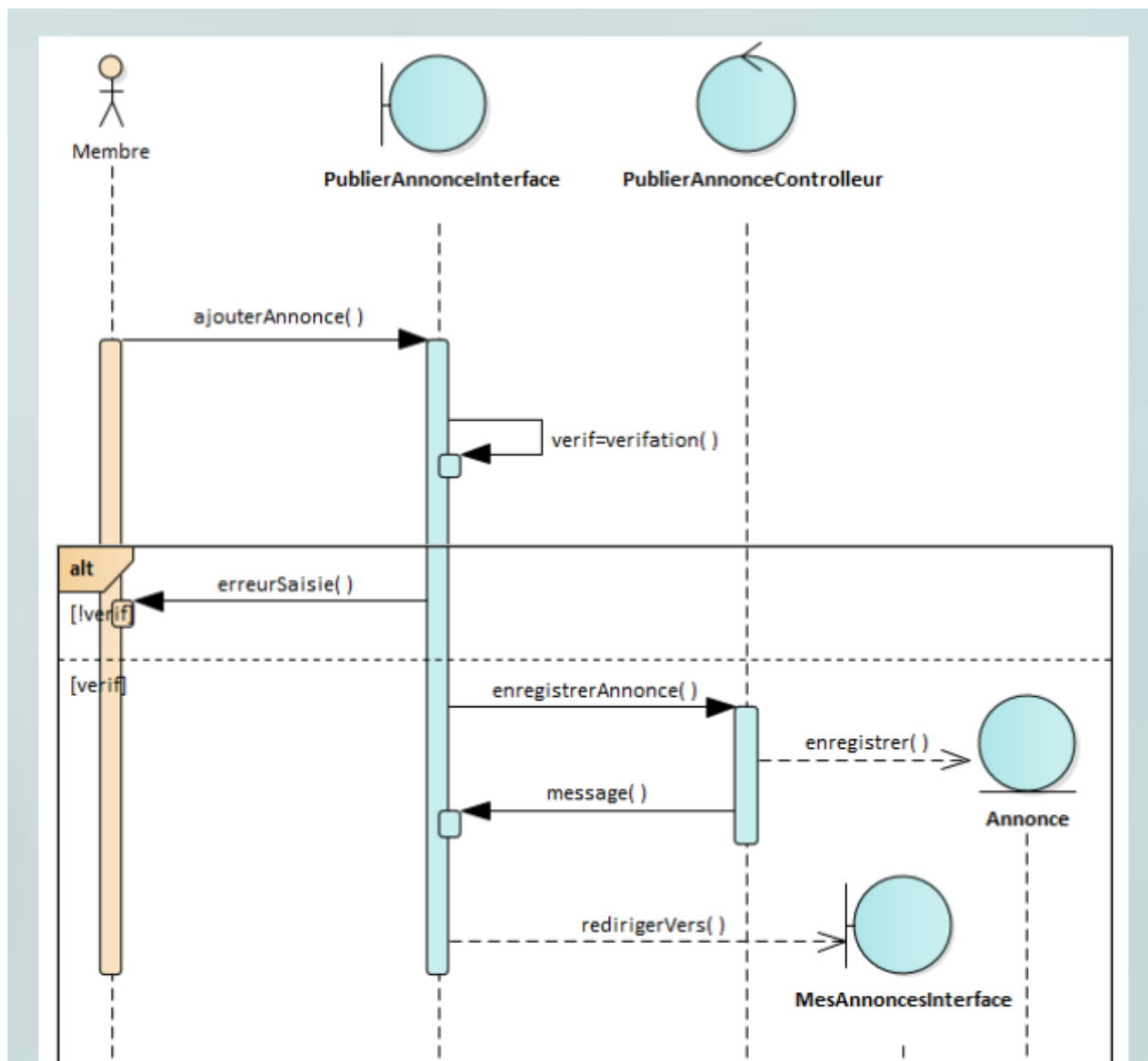


Figure 8: diagramme de Séquence "créer une annonce"

2. Gestion des réservations

- Les conducteurs peuvent voir la **liste des passagers inscrits** et le **statut des paiements** (confirmé, en attente).
- Possibilité de modifier ou d'annuler le trajet en fonction des circonstances.

E. INTERACTION ENTRE LES UTILISATEURS

L'application intègre plusieurs fonctionnalités pour faciliter la communication et garantir la transparence entre passagers et conducteurs.

- **Système de messagerie interne** : Les utilisateurs peuvent communiquer directement au sein de l'application pour coordonner les détails du trajet.
- **Appels téléphoniques sécurisés** : Option d'appeler un utilisateur sans partager son numéro de téléphone grâce à un service de redirection anonyme.
- **Système de notation** : Après chaque trajet, passagers et conducteurs peuvent laisser un avis et une note (sur 5 étoiles), contribuant à la réputation des utilisateurs et à la confiance au sein de la communauté.

F. PARAMETRES GENERAUX ET IMPORTANTS

L'application permet à l'utilisateur de personnaliser son expérience via plusieurs paramètres, offrant une flexibilité supplémentaire.

- **Paramètres du profil** : L'utilisateur peut modifier ses informations personnelles, y compris la photo de profil, les préférences de voyage (ex. : covoiturage avec ou sans animaux, conducteurs non-fumeurs), et la vérification du statut étudiant.
- **Notifications intelligentes** : L'application envoie des rappels de trajets, des suggestions personnalisées basées sur l'historique de recherche, et des notifications en cas d'événements importants (annulation de trajet, confirmation de paiement, etc.).
- **Sécurité et confidentialité** : Les utilisateurs peuvent choisir d'activer l'authentification à deux facteurs pour renforcer la sécurité de leur compte, et décider des informations qu'ils souhaitent partager avec les autres utilisateurs.

III. MAQUETTE VISUELLE DE LA V1

Dans le cadre de la conception de l'application Edudrive, nous avons réalisé une **maquette visuelle** fonctionnelle de la version 1 (V1) en utilisant **Figma**, un outil de prototypage interactif et puissant. L'objectif de cette maquette est de présenter de manière concrète et esthétique l'interface utilisateur et les principales fonctionnalités de l'application. Cette démarche s'inscrit dans l'étape cruciale de prototypage, qui permet de tester et d'améliorer l'expérience utilisateur avant le développement final.

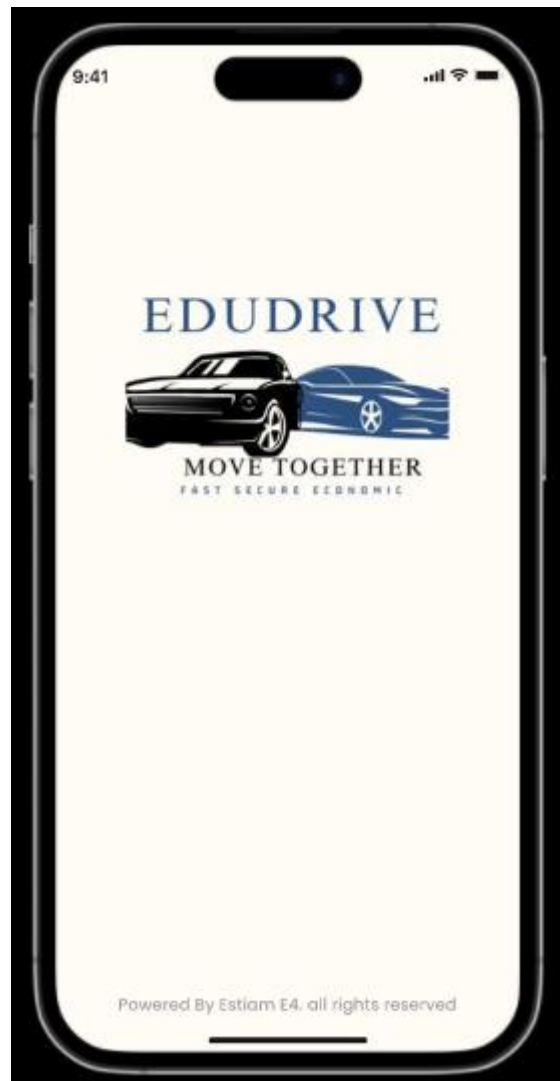
Voici les écrans principaux que nous avons intégrés dans la maquette visuelle de la V1 :

1. **Splash Screen** : Cet écran d'accueil affiche le logo de l'application Edudrive et un message d'introduction destiné à accueillir les utilisateurs. Il constitue la première impression de l'application et contribue à renforcer l'identité visuelle du projet.
2. **Onboarding** : Il s'agit d'une série de trois écrans de présentation qui expliquent les fonctionnalités clés de l'application et ses avantages. L'objectif est de faciliter la prise en main par les nouveaux utilisateurs et de leur montrer comment Edudrive peut les aider à réduire leurs frais de transport.
3. **Authentification** : L'écran d'authentification permet aux utilisateurs de s'inscrire ou de se connecter. Ils peuvent choisir entre une inscription classique avec leur email, ou via des comptes externes comme Google, et devront fournir un justificatif de scolarité pour valider leur statut étudiant.
4. **Accueil** : L'écran principal où les utilisateurs peuvent accéder rapidement aux fonctionnalités essentielles, comme la recherche de trajets pour les passagers ou la création de trajets pour les conducteurs. Un tableau de bord résume les trajets en cours et les économies réalisées.
5. **Sélection des départs et arrivées** : Cet écran permet aux utilisateurs de choisir leurs lieux de départ et d'arrivée, ainsi que la date et l'heure du trajet. Des filtres permettent d'affiner les résultats en fonction des préférences.
6. **Conducteur** : L'espace dédié aux conducteurs, où ils peuvent créer un trajet en entrant des détails comme les points de départ et d'arrivée, la date, l'heure, et le nombre de places disponibles. Ils peuvent également gérer les réservations des passagers.

7. **Les recherches passagers** : Cet écran permet aux passagers de rechercher des trajets disponibles, avec une présentation des résultats qui inclut le nom du conducteur, son véhicule, les avis, ainsi que les détails du trajet.
8. **Contact conducteur – passager** : Une interface de communication interne entre les conducteurs et les passagers, facilitant les échanges pour l'organisation du trajet.
9. **Avis conducteur** : Après chaque trajet, les passagers peuvent laisser un avis et une note sur leur expérience, ce qui contribue à la confiance et à la sécurité des utilisateurs.

Ces différents écrans illustrent les étapes majeures de l'expérience utilisateur et permettent de donner un aperçu complet des interactions clés au sein de l'application.

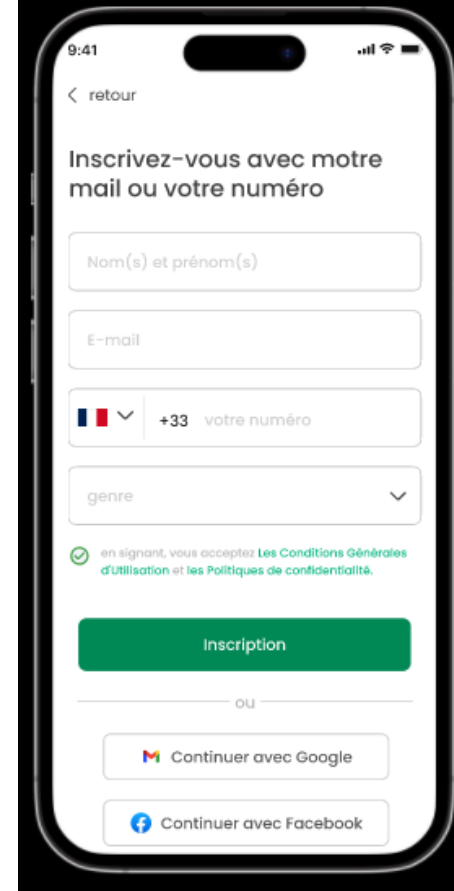
SPLASH

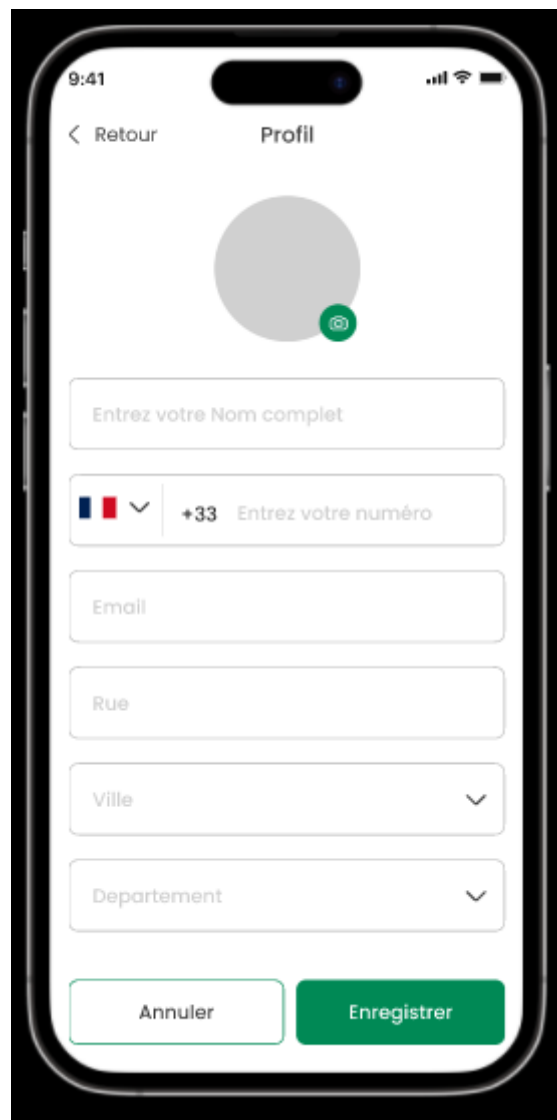
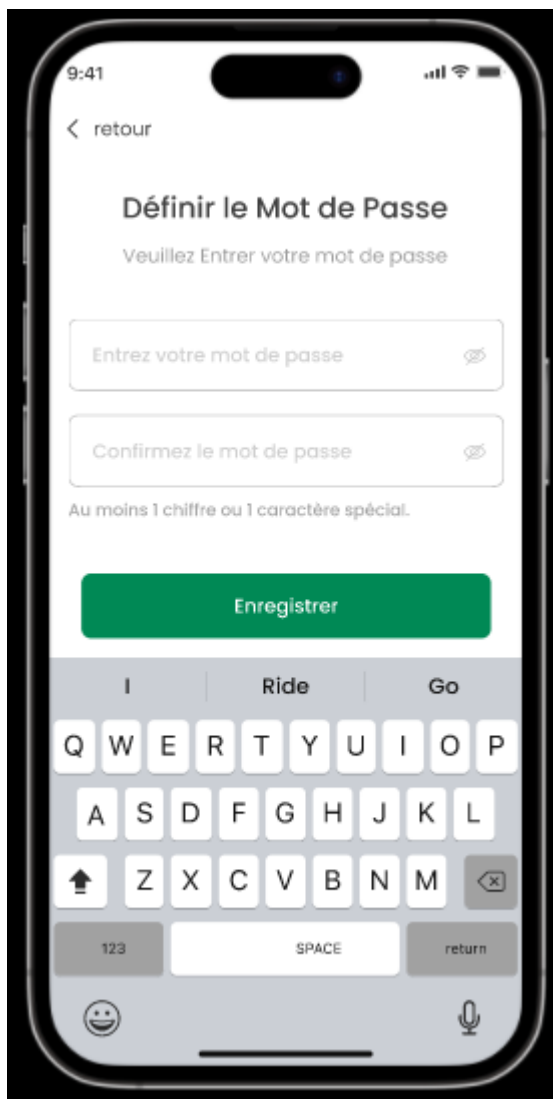


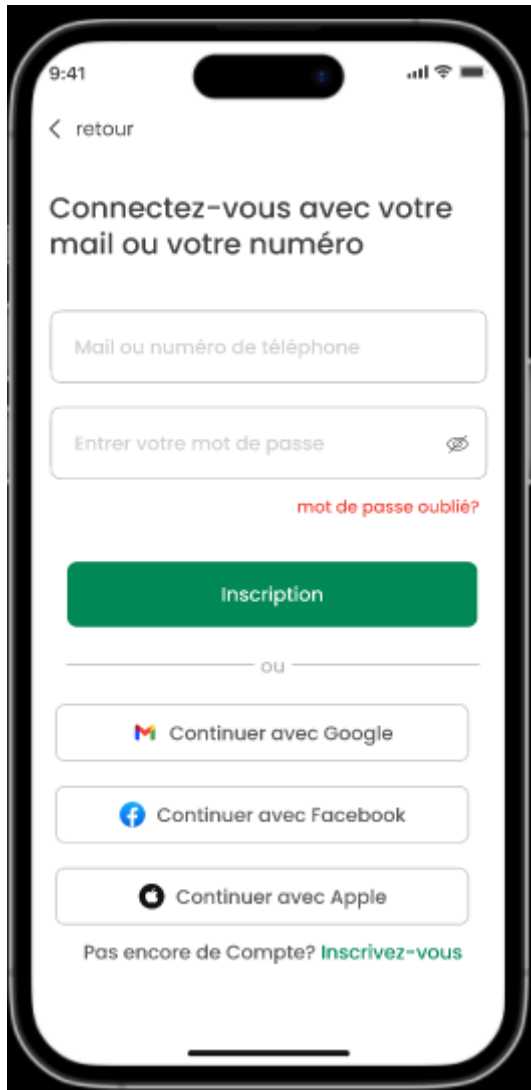
ONBOARDING



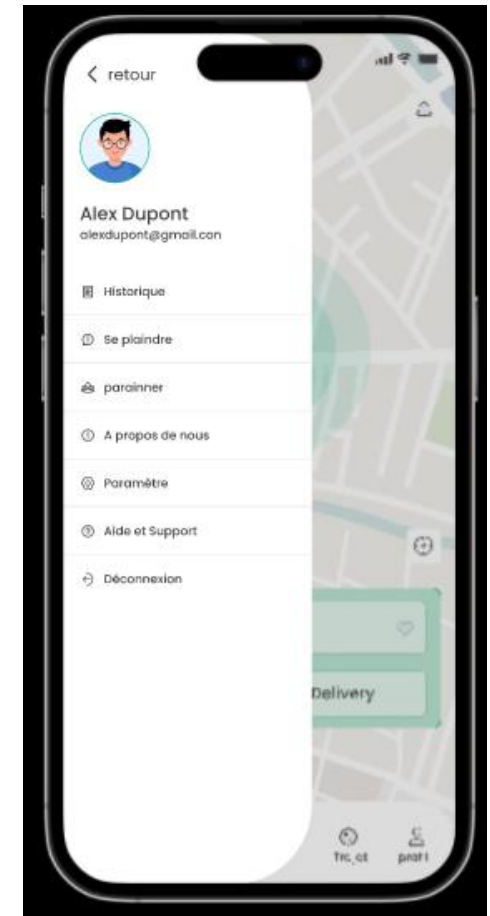
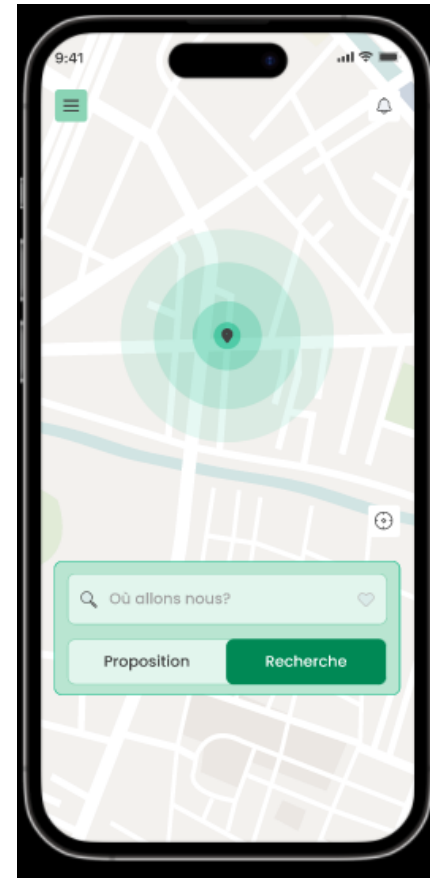
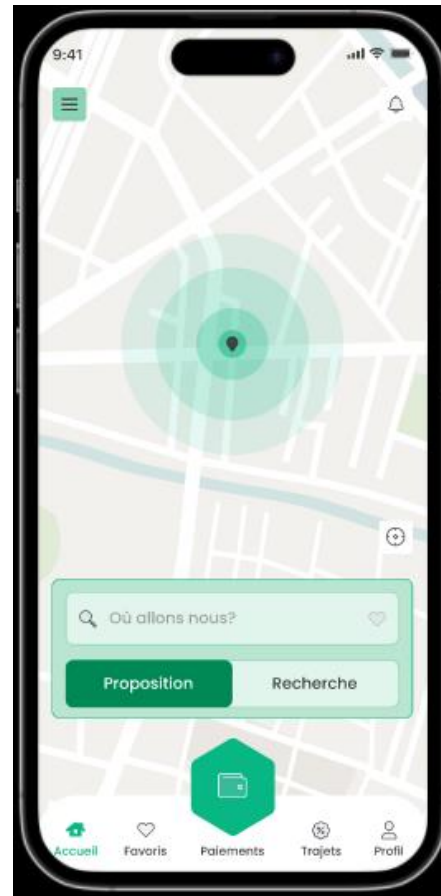
AUTHENTIFICATION



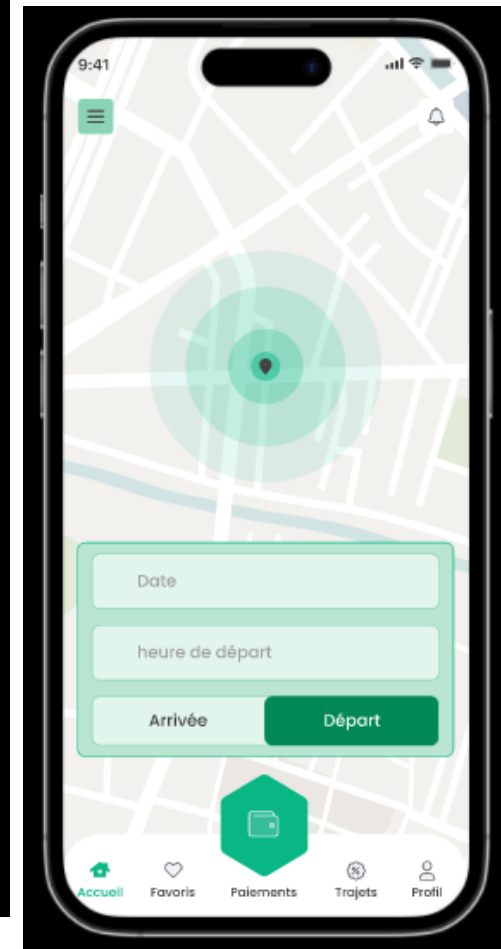
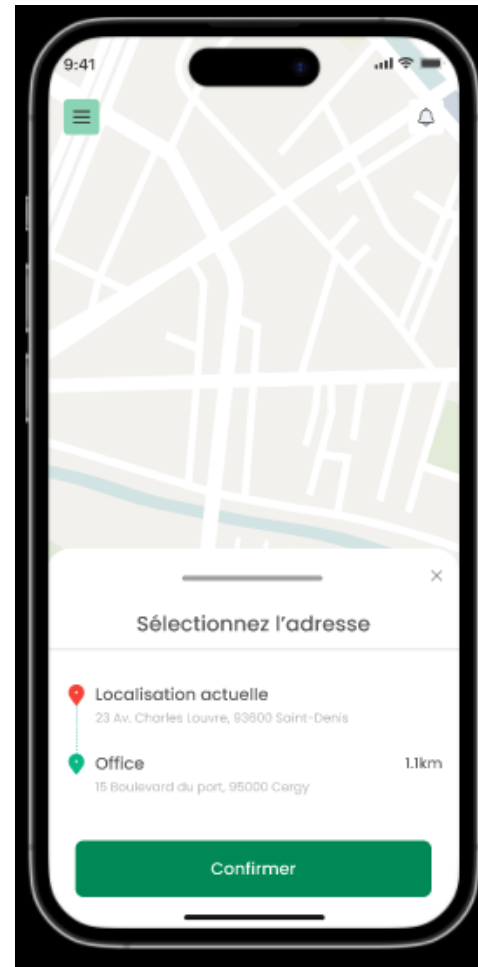
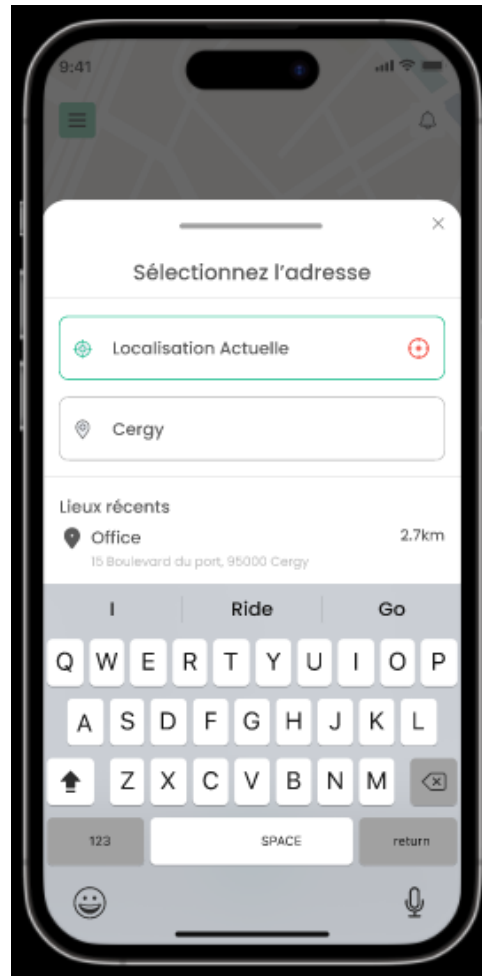
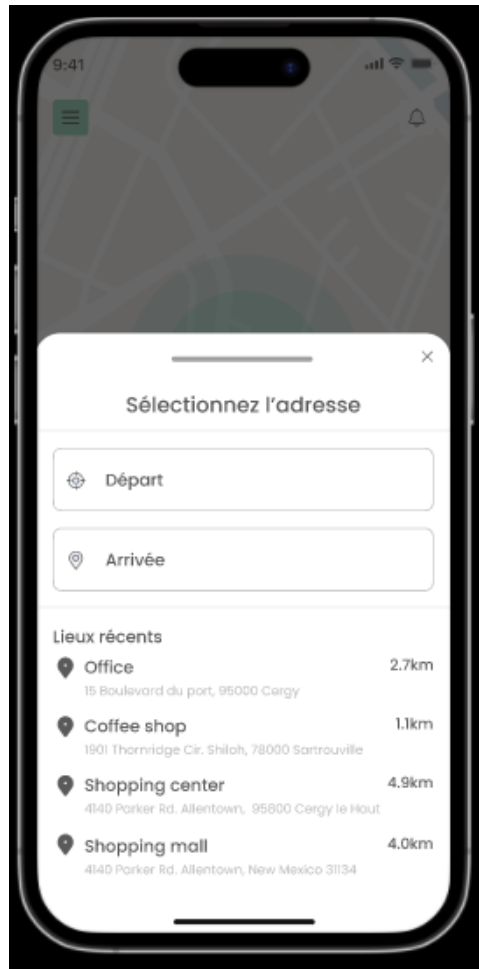




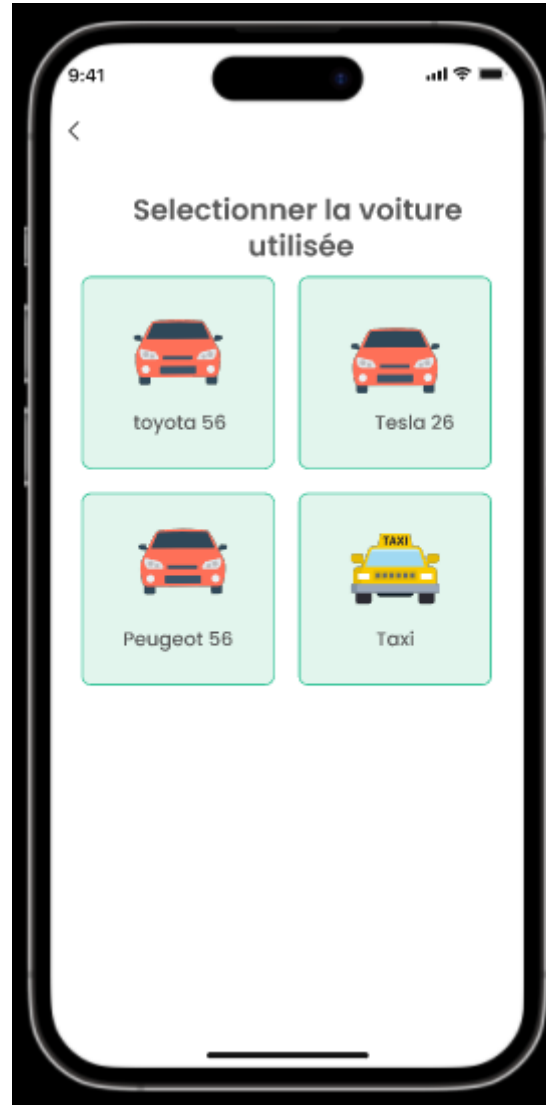
ACCUEIL



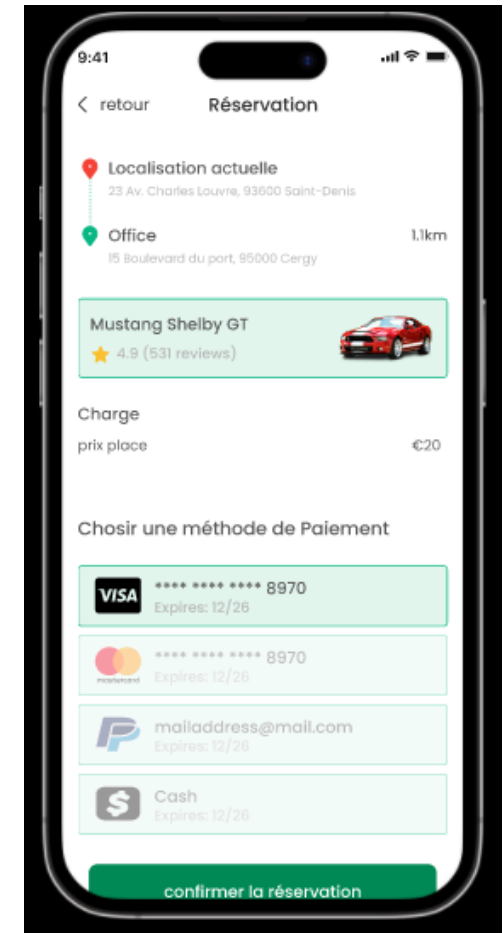
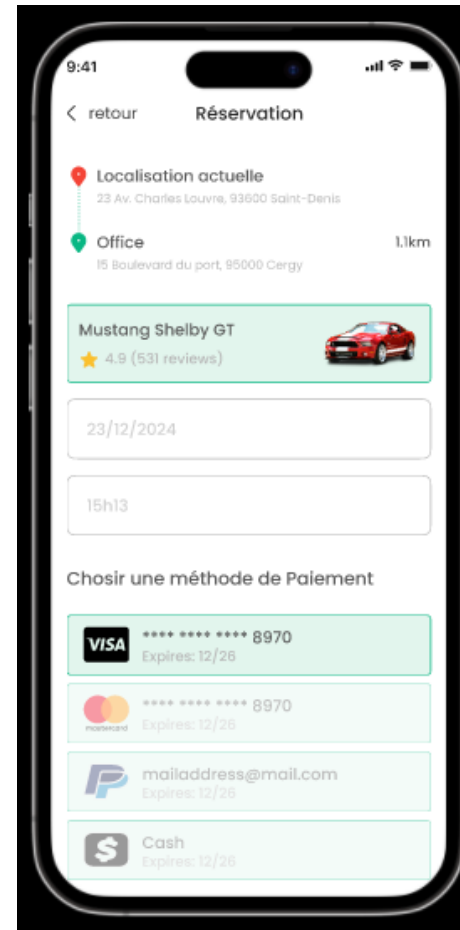
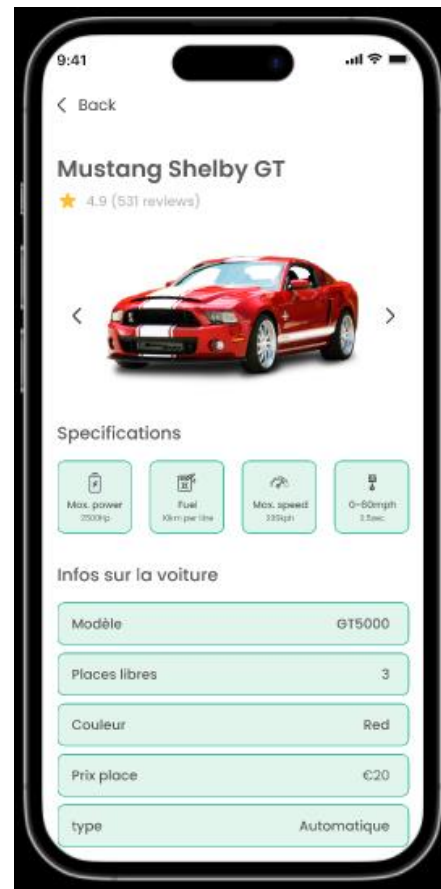
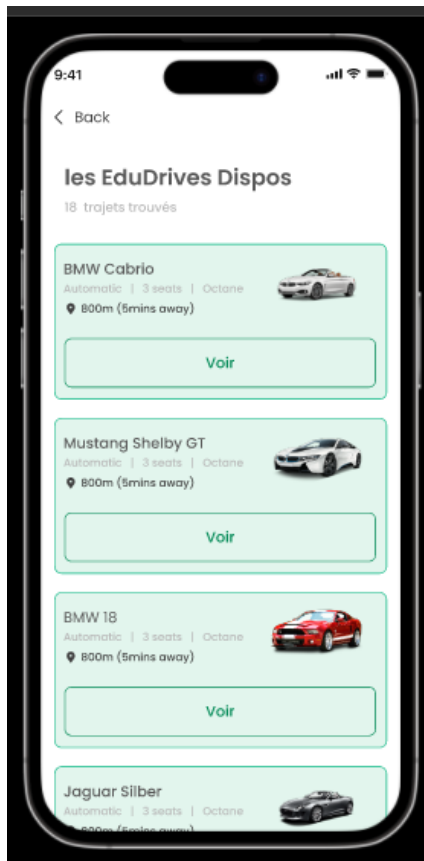
SELECTION DES DEPART ET DES ARRIVEES



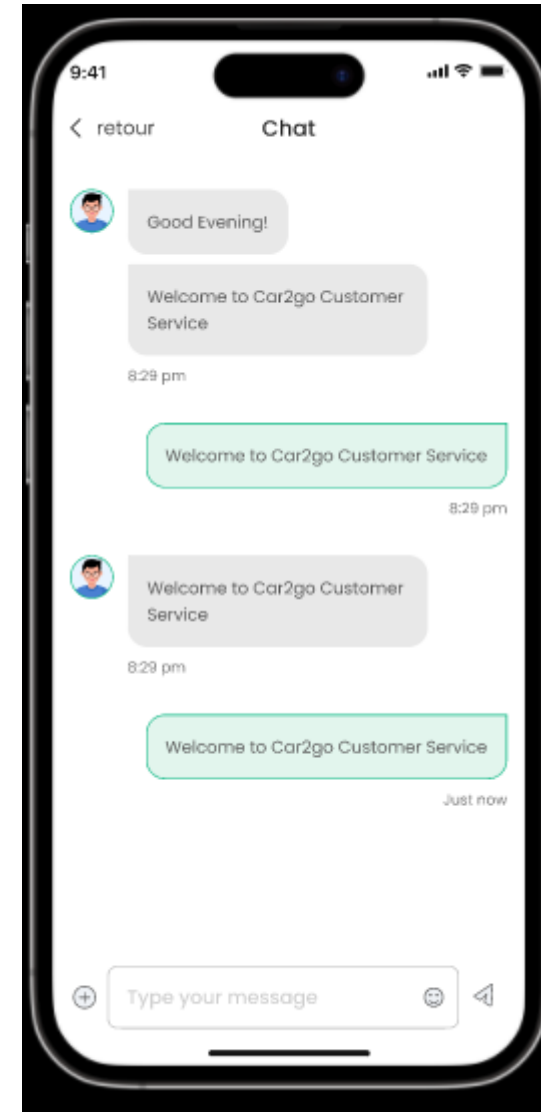
CONDUCTEUR



LES RECHERCHES PASSAGERS

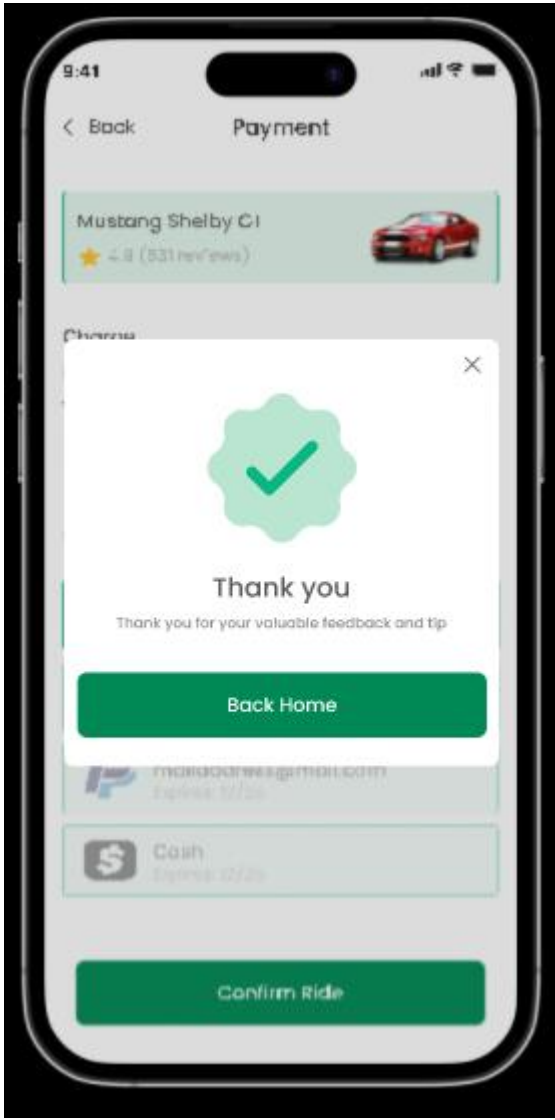
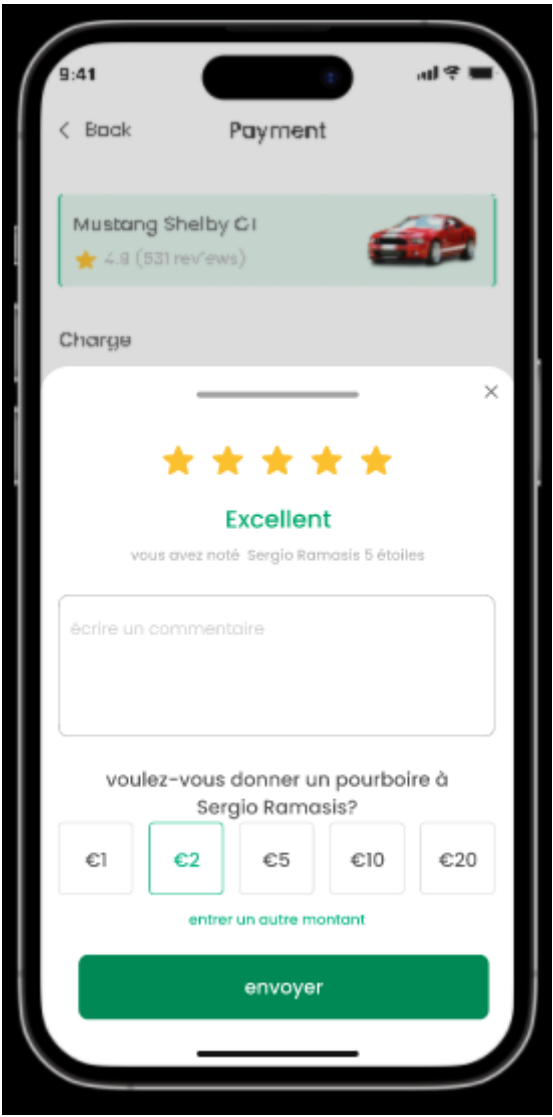
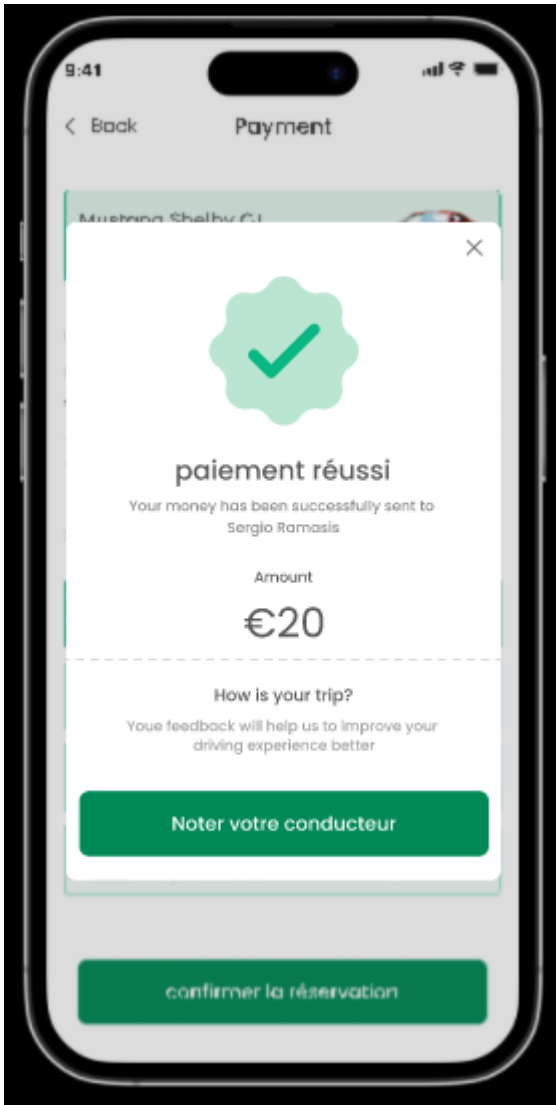


CONTACT CONDUCTEUR – PASSAGER





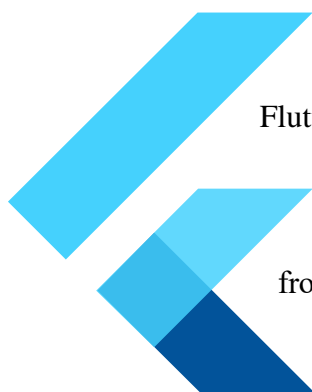
AVIS CONDUCTEUR



IV. CHOIX DES TECHNOLOGIES POUR EDUDRIVE

Pour le développement de l'application Edudrive, nous avons fait des choix technologiques en tenant compte de plusieurs contraintes, notamment la **performance**, la **compatibilité**, la **facilité d'utilisation** et surtout le **coût**. Voici un exposé des technologies que nous allons utiliser pour la **version 1** de l'application.

A. FRONTEND : FLUTTER



Flutter est un **framework open-source** développé par Google qui permet de créer des applications mobiles natives pour **iOS et Android** à partir d'un seul code base. Voici pourquoi nous avons choisi Flutter pour le frontend de l'application Edudrive :

- **Compatibilité multiplateforme** : Flutter permet de développer l'application à la fois pour iOS et Android sans avoir à maintenir deux bases de code distinctes. Cela réduit le temps de développement et les coûts associés.
- **Performance native** : Avec son moteur graphique Skia, Flutter offre des performances proches du natif. Les animations sont fluides et l'interface utilisateur est rapide, ce qui est crucial pour offrir une expérience utilisateur agréable.
- **Widgets personnalisables** : Flutter permet de créer une interface utilisateur élégante et cohérente grâce à des widgets hautement personnalisables. Nous pourrions ainsi respecter les normes esthétiques et fonctionnelles d'une application moderne.
- **Coût** : Flutter est gratuit et open-source, ce qui le rend extrêmement avantageux pour notre budget. De plus, en réduisant le besoin de développeurs spécialisés pour chaque plateforme, nous réalisons des économies importantes.

B. BACKEND : NODE.JS AVEC EXPRESS



Pour le backend, nous avons choisi d'utiliser **Node.js** associé au framework **Express**. Cette solution s'adapte parfaitement à nos contraintes en termes de performance, de coût et de compatibilité avec les autres services que nous allons intégrer.

- ❖ **Performance** : Node.js est conçu pour gérer un grand nombre de requêtes simultanément grâce à son architecture non bloquante basée sur un modèle asynchrone. Cela permet de maintenir des performances élevées même avec un grand nombre d'utilisateurs actifs.
- ❖ **Scalabilité** : Node.js est idéal pour des applications évolutives comme Edudrive. Avec la croissance prévue de notre base d'utilisateurs, il sera simple de mettre à l'échelle les services backend sans compromettre la performance.
- ❖ **Écosystème riche** : Express est un framework minimaliste et rapide qui s'intègre parfaitement avec Node.js. Il dispose d'un vaste écosystème de bibliothèques et d'outils permettant d'ajouter des fonctionnalités comme l'authentification, la gestion des paiements, les notifications, etc.
- ❖ **Coût** : Node.js et Express sont open-source, réduisant ainsi les coûts. De plus, leur popularité permet de trouver facilement des développeurs compétents à des tarifs abordables.

C. AUTHENTIFICATION : FIREBASE AUTHENTICATION



Pour gérer l'authentification des utilisateurs dans l'application Edudrive, nous avons choisi **Firestore Authentication**. Cette solution permet de simplifier et de sécuriser la gestion des identités, tout en offrant une flexibilité importante avec l'intégration de plusieurs méthodes de connexion. Plus concrètement, il permettra :

- **Support multi-plateforme** : Firestore Authentication s'intègre parfaitement avec **Flutter**, ce qui facilite la gestion des utilisateurs sur plusieurs plateformes (iOS et Android). Firestore gère l'authentification sur l'ensemble du cycle de vie de l'utilisateur.
- **Authentification via Google, Facebook et Apple** :
 - **Google** : L'intégration avec Google permet aux utilisateurs de se connecter rapidement et facilement à l'aide de leur compte Google, une option couramment utilisée par les étudiants.
 - **Facebook** : L'authentification via Facebook permet aux utilisateurs de se connecter avec leur compte Facebook, offrant une autre méthode simple pour l'inscription et la connexion.
 - **Apple** : L'authentification via Apple, requise pour les applications iOS, permet aux utilisateurs d'utiliser leur identifiant Apple pour se connecter en toute sécurité.
- **Sécurité et simplicité** : Firestore Authentication simplifie la mise en place d'une authentification sécurisée, avec une protection contre les attaques courantes comme le piratage par mot de passe. Il gère automatiquement des fonctionnalités comme la réinitialisation de mot de passe et la vérification des e-mails.
- **Coût** : Firestore Authentication est gratuit pour un nombre d'utilisateurs et d'authentifications de base. Cela permet de commencer avec des coûts très réduits et de passer à un niveau payant seulement en cas de croissance importante.
- **Intégration de l'authentification par OTP (One Time Password)** : Firestore permet également de gérer l'authentification via OTP, en envoyant un code de validation par SMS, ce qui peut être une méthode de connexion ou d'inscription supplémentaire, particulièrement utile pour les étudiants qui préfèrent la vérification par téléphone.

D. BASE DE DONNEES : POSTGRESQL (RELATIONNELLE)

Nous avons opté pour une **base de données relationnelle** avec **PostgreSQL** pour répondre aux besoins de gestion des relations complexes entre les utilisateurs, les trajets, et les réservations.

- **Structure relationnelle** : PostgreSQL permet de gérer des relations complexes entre les entités de l'application, telles que les conducteurs, les passagers, les trajets, et les avis. Cela garantit l'intégrité des données.
- **Transactions robustes** : PostgreSQL assure des transactions fiables, garantissant la cohérence des opérations critiques comme la réservation de trajets ou les paiements.
- **Performance et scalabilité** : PostgreSQL est optimisé pour des performances élevées avec des requêtes complexes, ce qui est nécessaire pour gérer de nombreux utilisateurs et trajets en temps réel.
- **Coût** : PostgreSQL est gratuit et open-source. Cela répond à nos contraintes budgétaires, tout en offrant un niveau de performance proche des solutions payantes.

E. MAP INTEGRATION: GOOGLE MAPS AND OPEN SOURCE ALTERNATIVES

Dans l'application Edudrive, l'intégration d'une carte est essentielle pour permettre aux conducteurs et passagers de visualiser les trajets, les points de départ et d'arrivée, ainsi que pour suivre la position en temps réel. Pour cela, nous avons choisi d'utiliser **Google Maps** dans un premier temps, avant d'explorer des alternatives open source comme **OpenStreetMap** et des codes sources open source tels que ceux de **Waze**.

1. Google Maps (Phase 1)

Nous commencerons par utiliser **Google Maps** pour les fonctionnalités de géolocalisation et d'itinéraires dans Edudrive. Google Maps est reconnu pour sa **précision** et ses **fonctions avancées** en matière de navigation et de localisation.

Avantages de Google Maps :

- **Précision et richesse des données** : Google Maps dispose d'une base de données d'itinéraires très complète et de points d'intérêt (stations, universités, commerces, etc.), idéale pour les besoins des étudiants.
- **Facilité d'intégration avec Flutter** : Google propose une API de Google Maps bien documentée et facile à intégrer dans Flutter, garantissant un **développement rapide**.
- **Fonctionnalités avancées** : Google Maps propose des fonctionnalités robustes telles que la **navigation en temps réel**, le **calcul d'itinéraires** optimisés, la **recherche d'adresses** et la **visualisation des conditions de circulation**, qui sont essentielles pour un service de covoiturage.
- **Compatibilité multiplateforme** : Fonctionne aussi bien sur Android que sur iOS, sans compromis en termes de performance.

Limites :

- **Coût** : Bien que Google Maps offre un quota gratuit, l'utilisation à grande échelle peut devenir coûteuse en fonction du nombre de requêtes (comme les affichages de cartes, les recherches d'itinéraires, et la géolocalisation).

2. OpenStreetMap (Phase 2 - Open Source Alternative)

Après avoir validé l'application avec Google Maps, nous pourrions explorer l'intégration d'**OpenStreetMap (OSM)**, une solution open source gratuite, pour réduire les coûts et avoir plus de flexibilité dans l'utilisation des données cartographiques.

Avantages d'OpenStreetMap :

- **Gratuité** : Contrairement à Google Maps, OSM est entièrement gratuit, ce qui permet de **réduire les coûts** à mesure que l'application gagne en utilisateurs.
- **Contrôle des données** : OSM offre une **liberté** totale pour accéder et modifier les données géographiques. Cela peut être un avantage si nous voulons personnaliser certains éléments de la carte ou ajouter des informations spécifiques aux étudiants.
- **Communauté active** : Une grande communauté d'utilisateurs et de développeurs enrichit constamment OSM, garantissant la qualité des informations cartographiques.

- **API open source** : L'intégration avec Flutter est possible grâce à des packages disponibles pour OSM, comme **flutter_map** ou **mapbox**.

Limites :

- **Moins de fonctionnalités avancées** : Par rapport à Google Maps, OSM peut manquer de certaines fonctionnalités comme le trafic en temps réel ou les points d'intérêt spécifiques (mais ces aspects peuvent être améliorés avec du développement supplémentaire).

3. Code Open Source de Waze (Phase 3)

À long terme, nous pourrions également intégrer certaines fonctionnalités du **code open source de Waze**, qui offre des solutions avancées en matière de **navigation communautaire**. Waze est une application de cartographie et de navigation populaire qui repose sur des données fournies par les utilisateurs pour signaler en temps réel des événements comme les embouteillages, les accidents, et d'autres conditions de circulation.

Avantages de l'intégration de Waze :

- **Données communautaires en temps réel** : Waze repose sur une **communauté d'utilisateurs** qui signalent en direct les conditions de circulation, une fonctionnalité précieuse pour les conducteurs étudiants.
- **Open source** : Bien que l'application Waze soit propriétaire, certaines de ses fonctionnalités et algorithmes peuvent être adaptés via des projets open source pour enrichir l'expérience utilisateur.

Limites :

- **Complexité d'intégration** : L'intégration du code source de Waze nécessitera plus de temps et d'expertise technique pour adapter les algorithmes et les intégrer à Edudrive.

F. CHAT SYSTEM: FIREBASE ET CUSTOM DEVELOPMENT AVEC NODE.JS

Dans la version initiale de l'application Edudrive, nous utiliserons **Firestore** pour gérer le système de messagerie entre les utilisateurs, car Firestore offre une solution rapide et facilement intégrable pour le **chat en temps réel**. Cependant, à long terme, nous prévoyons de développer notre propre système de chat en utilisant **Node.js**, pour plus de contrôle et de flexibilité.

1. Firestore for Chat (Phase 1)

Firestore propose une solution cloud complète pour les **communications en temps réel** grâce à **Firestore Realtime Database** et **Cloud Firestore**, permettant de gérer les messages instantanés entre les utilisateurs (conducteurs et passagers).

Avantages de Firestore pour le chat :

- **Simplicité d'intégration** : Firestore offre des bibliothèques spécifiques pour Flutter, ce qui rend l'implémentation du chat simple et rapide.
- **Messages en temps réel** : Grâce à **Firestore Realtime Database**, les messages sont synchronisés instantanément entre les utilisateurs, ce qui est essentiel pour une bonne communication dans une application de covoiturage.
- **Fiabilité** : Firestore est une solution éprouvée et **scalable**, ce qui permet de gérer la montée en charge des utilisateurs sans problème de performance.
- **Multi-plateforme** : Firestore fonctionne aussi bien pour les utilisateurs Android qu'iOS, assurant une **compatibilité totale**.

Limites :

- **Coût** : Firestore propose un modèle freemium, mais au-delà d'un certain nombre de messages ou d'utilisateurs actifs, les coûts peuvent augmenter, ce qui pourrait être un problème à long terme.
- **Dépendance externe** : Utiliser Firestore signifie dépendre des serveurs de Google, ce qui peut limiter notre capacité à contrôler certaines fonctionnalités ou à personnaliser le système selon nos besoins futurs.

2. Custom Chat System with Node.js (Phase 2)

À long terme, pour un meilleur contrôle et afin de **réduire les coûts**, nous envisageons de développer notre propre système de chat en utilisant **Node.js**. Node.js est une plateforme JavaScript côté serveur qui permet de construire des applications réseau **scalables** et **performantes**.

Avantages de Node.js pour un système de chat personnalisé :

- **Contrôle total** : En développant notre propre système de chat, nous aurons un **contrôle complet** sur les fonctionnalités, les notifications, et la manière dont les messages sont gérés.
- **Personnalisation** : Nous pourrions ajouter des fonctionnalités spécifiques au covoiturage étudiant, comme des messages préremplis (pour signaler un retard par exemple) ou des systèmes de réponses automatiques.
- **Réduction des coûts** : Contrairement à Firebase, où les coûts augmentent avec l'utilisation, une infrastructure avec Node.js et une base de données auto-hébergée peut à terme **réduire les coûts opérationnels**.
- **Flexibilité** : En utilisant Node.js avec une architecture basée sur des **sockets** (Socket.io par exemple), nous pourrions créer un chat encore plus fluide et rapide, optimisé pour l'expérience utilisateur.

Limites :

- **Temps de développement** : Le développement d'un système de chat personnalisé demande plus de temps et d'efforts, et nécessitera une infrastructure serveurs.
- **Maintenance** : Nous serons responsables de la maintenance et des mises à jour du système, ce qui demande plus de ressources à long terme.

G. CALL SYSTEM: APPEL VIA OPERATEUR INITIAL ET FUTURE INTEGRATION AVEC TWILIO

1. Operator Calls (Phase 1)

Dans un premier temps, nous allons utiliser les **appels téléphoniques directs via les opérateurs mobiles**. Cela signifie que les utilisateurs (conducteurs et passagers) pourront se contacter directement via leurs numéros de téléphone, en dehors de l'application.

Avantages des appels via opérateurs :

- **Simplicité** : Cela ne nécessite aucune intégration complexe dans l'application. Les utilisateurs peuvent simplement cliquer sur un bouton pour appeler l'autre partie via l'application native de téléphone.
- **Aucun coût de développement immédiat** : Étant donné que nous utilisons les services de télécommunication classiques, cela ne génère pas de coûts supplémentaires de développement ou de gestion de serveur pour les appels.

Limites :

- **Expérience utilisateur limitée** : Cette approche n'offre pas de contrôle sur les fonctionnalités des appels depuis l'application. L'utilisateur devra passer par l'interface du téléphone pour effectuer l'appel.
- **Absence de traçabilité** : Il sera difficile de **suivre** les interactions ou d'intégrer des fonctionnalités comme les enregistrements ou la signalisation d'un problème pendant l'appel.

2. Twilio Integration (Phase 2)

Dans une phase ultérieure, nous prévoyons d'intégrer **Twilio** pour offrir une solution d'appel plus intégrée directement dans l'application. Twilio est une plateforme cloud qui permet de gérer des communications téléphoniques, des SMS et même des vidéos, directement depuis une application.

Avantages de Twilio :

- **Appels intégrés** : Les appels peuvent être gérés directement dans l'application, offrant une **expérience utilisateur fluide** et plus intégrée.
- **Fonctionnalités avancées** : Twilio offre des fonctionnalités supplémentaires comme la **programmation d'appels**, la **traçabilité des communications**, et même des options pour **l'enregistrement des appels**.
- **Notifications en cas d'appel manqué** : Nous pourrions envoyer des notifications en temps réel si un utilisateur rate un appel, assurant ainsi une meilleure communication entre passagers et conducteurs.

Limites :

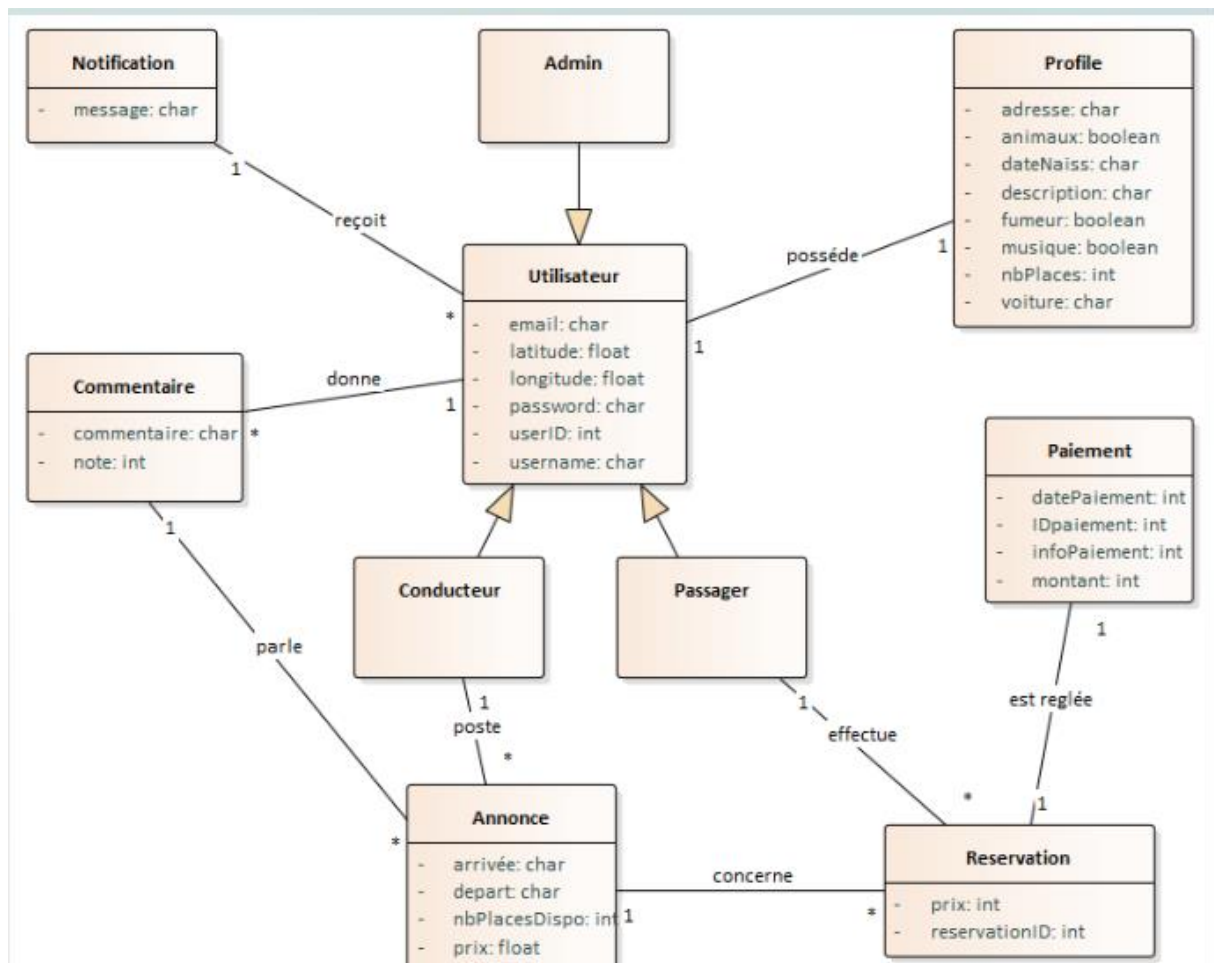
- **Coût** : Bien que Twilio soit une solution puissante, elle peut devenir coûteuse si le volume d'appels est élevé, surtout pour une application avec de nombreux utilisateurs actifs.
- **Complexité d'intégration** : Intégrer Twilio nécessite une **infrastructure backend** pour gérer les appels via des serveurs, ce qui augmente la complexité de développement.

H. NOTIFICATIONS: FIREBASE CLOUD MESSAGING (FCM)

Pour gérer les notifications push, nous allons utiliser **Firestore Cloud Messaging (FCM)**.

- **Facilité d'intégration** : FCM s'intègre parfaitement avec Flutter et Node.js, facilitant la mise en place des notifications push pour avertir les utilisateurs de nouveaux trajets, réservations, ou messages.
- **Gratuit** : FCM est gratuit, ce qui est un avantage significatif pour notre budget.

V. ARCHITECTURE DE LA BASE DE DONNEES



REFERENCES

- L'inflation des étudiants : <https://www.tf1info.fr/conso/face-a-l-inflation-les-etudiants-se-laissent-largement-seduire-par-le-covoiturage-2253874.html>
- <https://newsroom.blablacar.fr/news/etude-impact-social-covoiturage>
- https://www.lemonde.fr/emploi/article/2024/11/14/la-mobilite-un-frein-persistent-a-l-insertion-professionnelle-des-jeunes_6392905_1698637.html?
- <https://prezi.com/p/t51vqnbaydhi/projet-uml-presentation-covoiturage/>
- <https://jacquesjournetbtssio.wordpress.com/ppe-n10/>
- <https://www.figma.com/>
-