



Rapport du Projet pluridisciplinaire

2^{ème} année Classes Préparatoires (2CP)

PRJP06_EQ22

Thème :

Conception d'une application de covoiturage

Réalisé par :

Noms & Prénoms	Groupes
Chef d'équipe	
– ZOUBIR Mohamed Akram	– 03
Membres	
– ZAIDI Yasmine	– 03
– BENDAHRMANE Nesrine	– 03
– SISABER Rania	– 03
– BOUSSEBATA Issam	– 03
– HENNANE Douaa El IKhlas	– 05

Encadré par :

Noms & Prénoms
– Mme BENABIDALLAH
– Mme RAHMANI
–

Année : 2022 / 2023

Remerciements

Après la réalisation de ce travail, nous tenons tout d'abord à remercier le Tout-puissant de nous avoir accordé la force et la détermination nécessaires pour mener à bien ce projet.

Nous exprimons également notre reconnaissance envers nos familles pour leur soutien inconditionnel tout au long de ces 10 semaines de travail.

Notre gratitude va aussi à nos encadrants, Mme RAHMANI et Mme BENABIDALLAH, pour leur disponibilité, leur patience et leurs précieux conseils tout au long de ce projet.

Les personnes ayant contribué de près ou de loin à la réussite de ce travail méritent également nos remerciements. Nous leur sommes reconnaissants pour leur soutien, leur encouragement, leurs conseils et les ressources qu'ils ont partagés avec nous.

Résumé

Le projet pluridisciplinaire offre une opportunité unique à chaque membre de l'équipe de développer ses compétences d'ingénierie dans différents domaines. Notre équipe a été chargée de développer une application de covoiturage, et nous avons travaillé intensivement pendant dix semaines pour mener à bien ce projet. Nous avons suivi un processus en plusieurs étapes, commençant par l'analyse des besoins et la documentation, puis passant à la phase de conception, comprenant le découpage modulaire et l'identification des cas d'utilisation. Ensuite, nous sommes passés à la phase de réalisation, suivie de la préparation des manuels et du rapport du projet, qui documente toutes les étapes et les outils utilisés pour créer une application de covoiturage fonctionnelle.

Table de Matières

Résumé	3
Introduction	10
1. Introduction	10
2. Problématique.....	11
3. Objectifs	11
Chapitre 01 : Analyse des besoins et planification	12
1. Introduction	12
2. Solutions existantes.....	12
2.1. Yassir	12
2.2. Karos.....	13
2.3. Blablacar	13
4. Notre solution.....	14
5. Plan de travail.....	14
5.1 Recherche et documentation	15
5.2 Conception	15
5.3 Réalisation.....	15
5.4 Tests et rédactions des documents	15
6. Diagramme de Gantt.....	15
7. Conclusion.....	16
Chapitre 02 : Conception.....	17
1. Introduction	17
2. Analyse du cahier des charges.....	17
3. Le choix d'une application mobile	18
4. Description du système réalisé	18
4.1 Fonctionnalités principales.....	18
4.2 Fonctionnalités supplémentaires	20
5. Identifications des acteurs	22
5.1 Les utilisateurs.....	22
5.2 L'administrateur	22
6. Diagrammes UML.....	22
6.1 Diagramme de cas d'utilisation.....	22
6.2 Diagramme de classes	24

6.3	Diagrammes de séquences	29
7.	Charte graphique et ergonomie de l'application.....	32
7.1	Le nom de l'application	32
7.2	Le logo	32
7.3	Palette de couleurs et police	33
8	Conclusion	34
Chapitre 03 : Réalisation		35
1.	Introduction	35
2.	Outils et moyens techniques	35
2.1	Outils d'implémentation	35
2.2	Outils de design	35
2.3	Outils de communication.....	36
3.	Choix des langages de programmation	37
3.1	Flutter.....	37
3.2	Dart.....	37
4.	Étapes de la réalisation	37
4.1	Implémentation de la base de données	37
4.2	Les APIs utilisées.....	39
5.	Intégration des fonctionnalités de l'application.....	39
6.	Tests des fonctionnalités de l'application	40
7.	Présentation des interfaces principales	40
7.1	Ecran de chargement	40
7.2	Pages du formulaire d'inscription	41
7.3	Page des informations de la voiture.....	41
7.4	Page d'accueil	42
7.5	Page de création d'un trajet	42
7.6	Pages recherche d'un trajet	43
7.7	Page messagerie.....	43
7.8	Page profil.....	44
7.9	Page historique	44
7.10	Page notifications.....	45
7.11	Page Mes trajets.....	45
7.12	Page évaluation et commentaires	46
7.13	Page Administrateur	46
8	Conclusion	46

Conclusion.....	47
1. Conclusion générale	47
2. Perspectives	47
Annexe : Les défis rencontrés.....	48
1. Les applications modèles.....	48
2. API Google Map	48
3. Notifications	48
Bibliographie.....	49

Table de Figures

Figure 1: Application de Yassir.....	12
Figure 2 : Application de Karos	13
Figure 3: Application de BlaBlaCar	14
Figure 4: Diagramme de Gantt	15
Figure 5: Diagramme de cas d'utilisation	22
Figure 6:Diagramme de classes.....	24
Figure 7:Diagramme de séquence du cas d'utilisation : 'S'authentifier'	30
Figure 8:Diagramme de séquence du cas d'utilisation : 'S'inscrire'	30
Figure 9:Diagramme de séquence du cas d'utilisation : 'Rechercher un trajet'	31
Figure 10: Diagramme de séquence du cas d'utilisation : 'Publier un trajet'	32
Figure 11: Logo et nom de l'application	33
Figure 12:Palette de couleurs et police.....	33
Figure 13:Visual Studio Code	35
Figure 14: Firebase.....	35
Figure 15:Adobe Illustrator	35
Figure 16:Figma	36
Figure 17:Google Drive.....	36
Figure 18:Google Meet	36
Figure 19:Google Sheet.....	36
Figure 20:GitHub	36
Figure 21:Discord.....	36
Figure 22:Flutter.....	37
Figure 23:Dart	37
Figure 24: Collection Users.....	38
Figure 25: Collection Trips	38
Figure 26: Collection Car	38
Figure 27:Collection Notification	38
Figure 28:Collection chats.....	39
Figure 29:Interface Slogans.....	40
Figure 30:Interface Logo.....	40
Figure 31:Formulaire d'inscription.....	41
Figure 32:Interface informations de la voiture.....	41
Figure 33:Interface page d'accueil	42

Figure 34:Interface créer un trajet	42
Figure 35:Interface des résultats de recherche	43
Figure 36:Interface de formulaire de recherche	43
Figure 37:Interface Chat.....	43
Figure 38:Interface Profil	44
Figure 39:Interface modifier profil.....	44
Figure 40:Interface historique	44
Figure 41:Interface notifications	45
Figure 42:Interface Mes trajets.....	45
Figure 43:Interface évaluations et commentaires.....	46
Figure 44:Compte administrateur.....	46

Table de Tableaux

Table 1: La classe Utilisateur	25
Table 2: La classe administrateur	26
Table 3: La classe véhicule	26
Table 4: La classe évaluation	26
Table 5: La classe trajet	28
Table 6: La classe préférences	28
Table 7: La classe historique	29

Introduction

1. Introduction

Le problème de la congestion routière et du manque de transport est un enjeu d'importance majeure dans de nombreuses métropoles d'Alger. En effet, les embouteillages chroniques constituent un coût significatif en termes de temps et d'argent pour les usagers de la route, et ils ont également un impact néfaste sur l'environnement. En outre, le manque d'options de transport en commun efficaces et accessibles peut limiter considérablement la mobilité des personnes qui ne disposent pas d'une voiture ou ne peuvent pas se permettre les coûts associés à la possession d'un véhicule.

Face à ce problème, plusieurs solutions sont envisageables, telles que le covoiturage.

Le covoiturage est une solution efficace et durable pour répondre aux besoins de transport et réduire ainsi la congestion routière. Il s'agit d'une pratique consistant à partager son véhicule personnel avec d'autres personnes afin d'effectuer l'ensemble ou une partie d'un trajet. Le covoiturage présente de nombreux avantages tels que la réduction des coûts de transport en partageant les frais, la diminution de l'empreinte carbone grâce à la réduction des émissions de gaz, la diminution du trafic routier et les échanges entre les utilisateurs qui se font dans une ambiance conviviale.

De plus en plus populaire pour les déplacements quotidiens, les déplacements professionnels et les trajets longue distance. Cette pratique permet de réduire les coûts et l'empreinte carbone, tout en favorisant les liens sociaux et la convivialité.

Cependant, malgré ses nombreux avantages, le covoiturage reste sous-utilisé en raison de certains freins, tels que le manque de confiance entre les conducteurs et les passagers, la difficulté à trouver des partenaires de covoiturage compatibles, et la complexité de l'organisation des trajets.

Le rapport du projet fournira une vue d'ensemble détaillée des différentes phases du projet, allant de l'analyse des besoins à la mise en production de l'application. Il présentera de manière approfondie les étapes de conception et de réalisation de notre application, ainsi que ses fonctionnalités clés.

Le rapport sera organisé en plusieurs chapitres qui aborderont les points suivants :

Introduction.

Analyse des besoins et planification.

Conception.

Réalisation.

2. Problématique

Le manque de moyen de transport constitue une problématique majeure touchant de nombreux territoires, en particulier les zones rurales et périurbaines où l'offre de transport publique se révèle souvent limitée voire inexistante. Les étudiants et employés de l'Ecole Nationale Supérieure de l'Informatique, pour leur part, éprouvent des difficultés à se déplacer vers et depuis cet établissement, particulièrement pour ceux qui résident en dehors des zones urbaines et soumis à des horaires de travaux ou de cours atypiques.

Le covoiturage se présente comme une solution optimale face à ce problème de transport dont souffre la communauté de l'ESI d'Alger.

Ce besoin nous a poussé à poser la problématique suivante :

Comment concevoir et mettre en place une application de covoiturage efficace pour répondre aux besoins de mobilité des étudiants, personnels et enseignants de l'ESI ?

3. Objectifs

L'objectif fondamental de notre projet consiste à mettre en œuvre une application mobile de covoiturage destinée aux étudiants, personnels et enseignants de l'ESI. L'application mobile propose de rechercher une offre grâce à plusieurs critères et mettre directement les personnes faisant partie de l'environnement de l'ESI ayant des itinéraires similaires, quotidiens ou occasionnels, en contact pour partager un même véhicule selon le principe du covoiturage, et ainsi remédier aux différents problèmes de transport auxquels ils sont confrontés quotidiennement.

Chapitre 01 : Analyse des besoins et planification

1. Introduction

Avant d'entamer la phase de conception de notre application mobile, il est impératif de bien analyser le cahier des charges afin de déterminer d'une part toutes les exigences mentionnées, et d'autre part les outils qui serviront au mieux à l'élaboration de l'application. Pour cela, nous porterons une attention particulière aux applications de transport et de covoiturage.

2. Solutions existantes

Afin de travailler sur un exemple concret, nous avons consulté différentes applications mobiles de transport et de covoiturage. Après avoir étudié les fonctionnalités de ces applications, nous avons combiné ces informations avec ce qui nous a été demandé sur le cahier des charges afin d'aboutir à un résultat aussi propre que professionnel. On a sélectionné alors les applications suivantes :

2.1. Yassir

L'application Yassir propose un service de transport mettant en relation des chauffeurs avec des personnes désirant se déplacer par le biais de leur smartphone.

Services offerts :

- Commander une course à n'importe quel moment.
- Choisir le type de véhicule qui convient au trajet.
- Elle propose divers types de paiement.
- Estimer le prix avant la phase de recherche.
- Localiser le chauffeur le plus proche.
- Fournir toutes les informations concernant le trajet et le chauffeur.

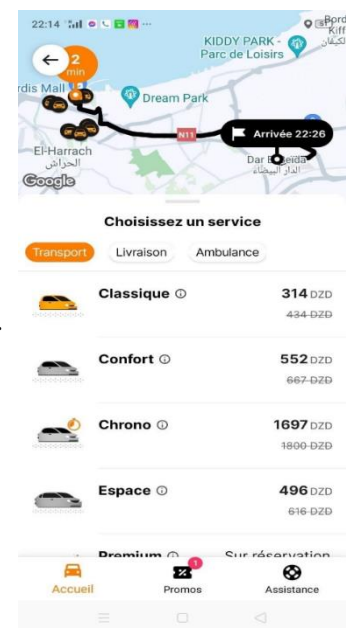


Figure 1: Application de Yassir

2.2. Karos

Karos est une application mobile de covoiturage dédiée aux trajets domicile-travail.

Services offerts :

- Les utilisateurs peuvent soit proposer un trajet en indiquant leur point et heure de départ et leur destination, soit chercher un trajet disponible.
- Un utilisateur peut proposer un trajet en tant que conducteur, chercher un trajet en tant que passager ou même alterner entre les deux en fonction des circonstances.
- L'utilisateur fournit ses adresses et heures habituelles (domicile/travail) et peut découvrir les covoitureurs qui partagent le même itinéraire.
- Les utilisateurs disposent de divers modes de paiements.
- A la fin de chaque trajet, un jeu est proposé afin de faire gagner des chèques cadeaux valables sur le site Amazon.

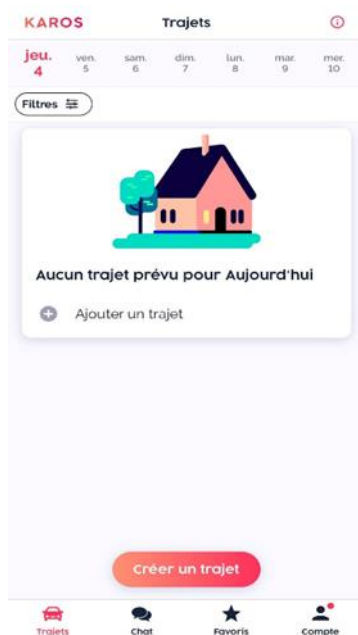


Figure 2 : Application de Karos

2.3. Blablacar

Blablacar est le leader mondial du covoiturage longue distance. Il s'agit d'une plateforme numérique, web et mobile qui permet de mettre en relation des conducteurs et des passagers souhaitant partager les frais d'un même trajet.

Services offerts :

- Les conducteurs publient une annonce dans laquelle ils décrivent leur trajet et indiquent le prix et le nombre de places libres.

- Les passagers peuvent consulter la liste des conducteurs partageant le même trajet grâce à un moteur de recherche (les informations fournies : départ, arrivée, date, heure, informations sur le conducteur et son véhicule, prix...)
- La plateforme propose des prix raisonnables avec différents modes de paiement.
- On peut y distinguer deux types de covoiturages : le domicile-travail (Blablacar Daily) et le covoiturage longue distance (Blablacar).

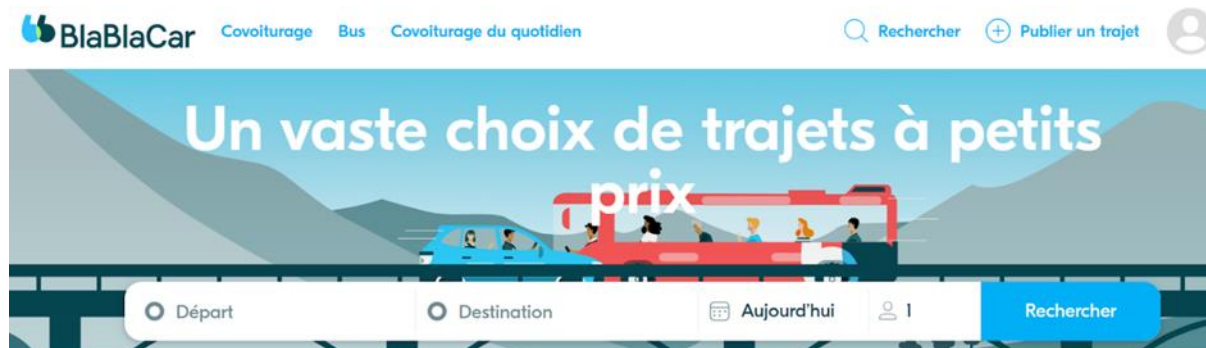


Figure 3: Application de BlaBlaCar

4. Notre solution

Le projet consiste en la conception et la réalisation d'une application mobile permettant de mettre en contact des personnes souhaitant avoir recours au covoiturage.

Nombreux sont les étudiants et employés au sein de l'ESI (école nationale supérieure d'informatique) qui doivent faire face à un manque de moyens de transport. Ainsi, notre application de covoiturage est conçue pour assurer la sécurité de nos utilisateurs en limitant l'accès uniquement aux étudiants et employés de l'école. Cette restriction garantit que les membres de la communauté ESI se déplacent avec des personnes de confiance, réduisant ainsi les risques de comportements inappropriés ou de mauvaise utilisation de l'application. De plus notre équipe de modération surveille en permanence les activités sur la plateforme pour s'assurer que les conducteurs et les passagers respectent les normes de sécurité et les règles établies.

5. Plan de travail

Dès le lancement de notre projet, et afin d'assurer son bon déroulement, nous avons planifié notre travail en fonction de nos compétences et préférences respectives, ce qui nous a permis d'établir un plan détaillé mettant en évidence les tâches assignées à chaque membre de l'équipe.

Nous avons tenu compte des différentes étapes du projet, depuis l'analyse du cahier des charges jusqu'aux tests et à l'intégration.

Nous avons également établi un rythme de travail régulier, avec des réunions chaque dimanche en présentiel ou à travers Google-Meet, sans compter une séance de travail accompagné de nos encadrants le mardi de chaque semaine.

Nous avons jugé bon de diviser notre travail en quatre phases principales :

5.1 Recherche et documentation

Cette étape initiale implique la familiarisation avec le sujet. Pour cela, nous avons procédé par une recherche approfondie sur le thème du projet avant d'entreprendre l'analyse du cahier des charges fourni. Nous avons également acquis des connaissances sur les outils techniques nécessaires pour la mise en œuvre de notre application.

5.2 Conception

La conception de l'UI (user interface) d'une application mobile implique la création de l'apparence et de l'expérience utilisateur de l'application. Une recherche et documentation à satiété ont suffi pour clôturer la recherche théorique et entamer la conception du projet. Durant cette étape, nous avons défini la structure de nos données, avons fait le découpage modulaire à l'aide de diagrammes UML et réaliser le prototype de l'application pour avoir une vue précoce du résultat final.

5.3 Réalisation

Une fois que la conception est finalisée, la phase de réalisation constitue le développement des fonctionnalités de l'application, cela inclut la programmation du code source de l'application, l'intégration des APIs et services, tests et débogage afin de s'assurer que l'application est bien fonctionnelle.

5.4 Tests et rédactions des documents

La dernière phase de la réalisation du projet consiste à tester et à vérifier le bon fonctionnement de l'application ainsi qu'à la résolution des éventuels problèmes rencontrés. Durant cette phase, le travail se concentre également sur la rédaction des livrables du projet afin de documenter le travail effectué et de fournir des informations utiles aux utilisateurs.

6. Diagramme de Gantt

Pour assurer une bonne répartition des tâches tout au long de la réalisation de notre projet, nous avons eu recours au diagramme de Gantt qui nous a aidé à visualiser l'avancement de notre travail, de répertorier les tâches à effectuer et de respecter les délais à respecter.

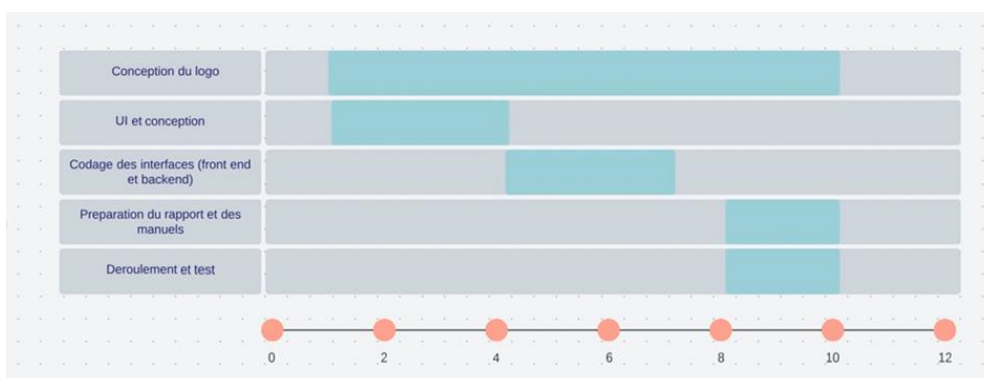


Figure 4: Diagramme de Gantt

7. Conclusion

Dans ce premier chapitre, nous avons exposé notre stratégie de gestion qui nous a aidé à mieux planifier, et organiser notre travail, ainsi que l'étude de quelques solutions existantes.

Chapitre 02 : Conception

1. Introduction

Pour parvenir à une organisation plus efficace et une gestion optimale du travail et donc développer une application de qualité, il est essentiel de suivre une approche méthodologique rigoureuse. Pour cela nous sommes parvenus au langage de modélisation UML pour la représentation claire qu'il propose à travers ses différents diagrammes.

Les différentes étapes de conception que nous allons présenter dans ce chapitre se résument comme suit :

- Analyse détaillée du cahier des charges.
- Justification du choix de l'application mobile.
- Description du système réalisé.
- Présentation des diagrammes UML.
- Conception de la charte graphique et de l'ergonomie de l'application.

2. Analyse du cahier des charges

Avant d'entamer la phase de conception de notre application mobile, il est impératif de bien analyser le cahier des charges afin de déterminer d'une part toutes les exigences mentionnées, et d'autre part les outils qui serviront au mieux à l'élaboration de l'application.

Le cahier des charges qui nous a été fourni comprend une fiche de présentation du problème suivi d'une fiche de synthèse des objectifs. Nous avons été chargés de concevoir et de réaliser une application web ou mobile de covoiturage permettant de connecter des personnes souhaitant partager un trajet commun.

Nous avons jugé que le développement d'une application mobile serait la meilleure solution pour répondre aux besoins des utilisateurs de notre service de covoiturage, offrant les fonctionnalités suivantes :

- **Inscription** : permettre aux utilisateurs de s'inscrire sur la plateforme en créant un compte utilisateur.
- **Création de trajets** : Les utilisateurs peuvent publier un trajet avec le point de départ, la destination, l'heure de départ, le nombre de places disponibles, les préférences (fumeur ou non, accepter les animaux, transport de bagages...).
- **Recherche de trajets** : Les passagers peuvent rechercher des trajets en spécifiant leur départ, leur destination et la date souhaitée, en ayant recours à la liste des trajets disponibles avec le même tronçon recherché (il représente un sous-trajet du trajet proposé).
- **Profils utilisateurs** : Les utilisateurs peuvent gérer leur profil, y compris leurs informations personnelles. Ils peuvent aussi gérer les informations de leurs véhicules (marque, type, matricule, assurance, modèle).
- **Commentaires et évaluations** : Les passagers peuvent laisser des commentaires sur le trajet qu'ils ont effectué, ce qui permet aux utilisateurs d'avoir une idée de la qualité du voyage et du conducteur.

- **Messagerie** : Mettre en place un système de messagerie pour permettre aux demandeurs et offrants de rentrer en contact afin de finaliser, ajuster, modifier ou annuler un trajet.
- **Sécurité** : L'application doit assurer la sécurité pour les utilisateurs.

Il convient de noter que ces fonctionnalités demeurent d'ordre général et feront ainsi l'objet de plus de détails dans la section de description du système réalisé surcroît, plusieurs autres fonctionnalités jugées indispensables ont été ajoutées dans le but de se démarquer et de répondre à tous les besoins de nos utilisateurs.

3. Le choix d'une application mobile

Pourquoi une application mobile ?

Tout d'abord, une application mobile permet une plus grande mobilité et flexibilité pour les utilisateurs. Ils peuvent utiliser l'application à partir de n'importe où, à tout moment, sans avoir à transporter un ordinateur portable.

En outre, les applications mobiles permettent également une meilleure utilisation des fonctionnalités du téléphone, telles que la géolocalisation, l'appareil photo et les notifications. Cela permet une expérience utilisateur plus interactive et personnalisée, avec des alertes en temps réel sur les itinéraires, les conducteurs disponibles, les mises à jour de trajet...

Enfin, les applications mobiles sont souvent plus conviviales et plus faciles à utiliser que les applications web, grâce à leur interface simplifiée et à leur expérience utilisateur plus intuitive.

4. Description du système réalisé

Au lancement de l'application par un visiteur, la première page affichée est celle contenant le logo de l'application, suivie d'une animation présentant des slogans en lien avec le covoiturage. Cette animation invite le visiteur à s'inscrire ou à s'authentifier pour accéder à l'ensemble des services offerts par l'application.

Une fois arrivé sur la page d'accueil de l'application, l'utilisateur est accueilli par une carte (map). En haut de la page, il trouve une barre de recherche ainsi qu'un bouton menant vers la liste des notifications lui permettant de voir les mises à jour importantes en temps réel. En bas de la page, un bouton circulaire lui permettra de créer un nouveau trajet. Il y trouve également une barre de navigation donnant accès à la messagerie, à ses trajets (trajets en cours, trajets réservés, trajets proposés) et au profil de l'utilisateur, en plus de la page actuelle.

Cette mise en page a été conçue pour offrir une expérience utilisateur conviviale et intuitive. Toutes les fonctionnalités de l'application sont facilement accessibles et l'utilisateur peut facilement naviguer pour trouver les informations dont il a besoin.

Notre application de covoiturage implémente les fonctionnalités suivantes :

4.1 Fonctionnalités principales

4.1.1 Inscription

Le système doit permettre aux conducteurs ainsi qu'aux passagers de pouvoir s'inscrire de façon autonome via l'application. Notons que seulement les personnes appartenant à la communauté de l'ESI (étudiants, enseignants, employés) ont la possibilité de s'inscrire en utilisant l'e-mail fournie par l'administration (exemple : lr_sbr@esi.dz, ens sbr@esi.dz). Un formulaire leur est fourni, exigeant de saisir en premier lieu l'adresse email. Pour passer à l'étape suivante, le potentiel utilisateur devra ouvrir le mail envoyé par le système afin de confirmer la validité de son adresse. L'utilisateur doit également fournir les informations suivantes : nom, prénom, genre, statut (étudiant, enseignant, employé) et les informations de son véhicule (facultatif).

4.1.2 Créer des trajets

Les utilisateurs peuvent proposer des trajets qu'ils souhaiteront effectuer en tant que conducteur. Pour créer un trajet, l'utilisateur doit sélectionner le point de départ et d'arrivée, ainsi que la date et l'heure du trajet. Il peut aussi spécifier le nombre de places disponibles dans son véhicule, la méthode de paiement qu'il exige et ajouter des détails supplémentaires tels que les préférences en matière de musique, de nourriture, de bagage ou autres. Notons que l'utilisateur peut créer un trajet seulement s'il possède un véhicule.

4.1.3 Recherche de trajets

Les utilisateurs peuvent rechercher des trajets disponibles en spécifiant le lieu de départ, la destination, la date et l'heure du trajet. Une liste des trajets correspondants leurs est affichée, ou ils peuvent consulter d'avantages informations sur le trajet et le conducteur (préférences, évaluations, commentaires, prix, date, heure, point de départ, destination, nombre de places disponibles, informations sur le véhicule et le conducteur).

4.1.4 Évaluations

Les passagers ont la possibilité de voir les évaluations des conducteurs avant de commander un trajet. Les évaluations sont basées sur une échelle de 1 à 5 étoiles, ce qui permet aux utilisateurs de donner leur avis sur leur expérience partagée avec un conducteur à la fin de chaque trajet pour noter la ponctualité, la sécurité et la courtoisie de ce dernier en appuyant simplement sur le nombre d'étoiles correspondant à leurs expériences, où une étoile représente un trajet négatif et cinq étoiles un trajet positif. Tout chauffeur a la possibilité de voir son évaluation depuis son profil.

4.1.5 Commentaires

À la fin de chaque expérience, le passager est en mesure de laisser des commentaires exprimant plus en détail son expérience avec le conducteur. Les commentaires peuvent inclure des informations sur le chauffeur, telles que leur convivialité, leur comportement, le niveau de confort ou d'inconfort pendant le trajet. Les autres utilisateurs peuvent ainsi prendre une décision éclairée avant de prendre un trajet puisqu'ils peuvent lire les commentaires des autres utilisateurs.

4.1.6 Gestion des trajets

Le système doit permettre l'ajout, modification et suppression d'un trajet par un conducteur ou un passager. Ces derniers peuvent consulter leurs anciens trajets à travers l'historique. Si un utilisateur met à jour un trajet, les autres utilisateurs sont instantanément notifiés dès la mise à jour d'un trajet.

4.1.7 Gestion des réservations

Les clients auront également la possibilité de réserver au préalable en choisissant un des trajets proposés après avoir effectué la recherche. En cas de changement de plan, un système de notification envers le conducteur ou le passager est disponible afin d'aviser que le passager n'embarque plus avec le conducteur ou que le conducteur souhaite annuler son trajet ou bien le modifier.

4.1.8 Messagerie (Groupe de discussion)

L'intégration d'une fonctionnalité de chat dans une application de covoiturage est très utile car elle permet une communication rapide et efficace entre un conducteur et un passager. Afin d'améliorer l'expérience utilisateur, nous avons opté pour la création d'un groupe de discussion regroupant le conducteur et les passagers associé au trajet plutôt qu'un simple chat. Cela permet aux passagers de vérifier rapidement si une question a déjà été posée, ce qui facilitera la gestion du trajet en ayant une communication plus fluide.

-Pour distinguer les membres du groupe, les messages du conducteur sont affichés en rose tandis que les messages des passagers sont affichés en bleu.

-L'utilisateur est en mesure d'envoyer sa position actuelle ainsi que des photos dans la messagerie. Par exemple, si le conducteur désire récupérer un passager d'un endroit précis, il suffit que le passager clique sur le bouton tout à fait à droite en bas de la page pour qu'il envoie sa position actuelle.

4.1.9 Gestion des comptes

L'utilisateur est en mesure de gérer son compte en modifiant ses informations personnelles ou celles de son véhicule en accédant à son profil.

4.2 Fonctionnalités supplémentaires

4.2.1 Historique

L'historique se réfère aux informations relatives aux trajets effectués par un utilisateur. Il pourra ainsi garder des traces des trajets effectués avec les détails suivants : lieu de départ, destination, date et heure, prix, nom et profil du conducteur.

4.2.2 Notifications

Le système de covoiturage envoie instantanément des notifications aux utilisateurs pour les tenir au courant des mises à jour du trajet. Les passagers peuvent recevoir des notifications concernant l'état de leur commande, tel que la validation ou le refus de leur demande. Ils sont également notifiés si des modifications ont été apportées au trajet. Le conducteur quant à lui, peut recevoir des notifications de demande de covoiturage ainsi que les mises à jour du trajet.

4.2.3 Adresse de l'école et position actuelle

Afin de faciliter l'étape de saisie des adresses de départ et de destination, nous avons mis en place un bouton contenant le logo de l'ESI avec lequel il est possible de sélectionner l'adresse de notre école seulement en cliquant dessus. L'utilisateur peut aussi sélectionner sa position actuelle en cliquant sur un bouton qu'il peut trouver sur la barre d'adresse.

4.2.4 Mode de paiement

Avant la publication d'une nouvelle annonce, le conducteur est en mesure d'imposer un mode de paiement. Les covoiturages peuvent être proposés en échange de services ou paiement en espèces ou autres. Par exemple, un conducteur peut offrir un covoiturage en échange de partage des frais de la voiture, comme il peut demander de l'aide dans la résolution d'un TP ou devoir.

4.2.5 Service clients

Une adresse e-mail ainsi qu'un numéro téléphone sont fournies dans les paramètres de l'application, permettant à l'utilisateur de contacter l'administrateur afin de signaler un problème, des erreurs ou de suggérer des améliorations à notre système. Cette fonctionnalité assure une communication rapide et efficace entre les utilisateurs et l'équipe de support de l'application.

4.2.6 Changement du mot de passe

En cas d'oubli du mot de passe lors de l'authentification, l'utilisateur peut changer ce dernier en appuyant sur 'mot de passe oublié ?'. Un lien lui est instantanément par mail le menant vers une page où il pourra fournir son nouveau mot de passe. Dans le cas où il souhaite changer son mot de passe, il peut également le faire en accédant aux paramètres, où il devra saisir son mot de passe actuel suivi du nouveau, et doit enfin le confirmer.

4.2.7 Suppression du compte

L'utilisateur est en mesure de supprimer son compte en accédant aux paramètres de l'application. Il lui suffira simplement d'introduire son mot de passe pour confirmer la suppression de son compte.

4.2.8 Les sous-trajets

Lorsqu'un utilisateur recherche un trajet, une liste d'annonces lui est affichée. Ce système de recherche est conçu de la sorte qu'au minimum 40% du trajet soit inclut dans les trajets proposés, il est donc un sous-trajet de ces derniers. En somme, si l'utilisateur ne trouve pas un conducteur qui partage exactement le même itinéraire, il pourra effectuer une partie du trajet et s'arranger avec le conducteur via la messagerie.

4.2.9 Mes trajets

Sur la barre de navigation, on peut trouver une section de trajets. L'utilisateur y trouve les trajets en cours, les trajets réservés (futurs trajets) ainsi que les trajets créés.

4.2.10 Calcul de la distance totale

La distance totale entre le départ et l'arrivée est toujours affichée à l'utilisateur, une fois la création ou la recherche d'un trajet est effectuée.

5. Identifications des acteurs

Un acteur représente un utilisateur humain, matériel ou un logiciel interagissant directement avec le système étudié.

Dans le système élaboré, nous avons identifié les acteurs suivants :

5.1 Les utilisateurs

Les utilisateurs de notre application sont en mesure d'une part d'alimenter le système en itinéraires en créant et publiant des trajets, et d'autre part de lancer diverses recherches afin de trouver des trajets qui répondent à leurs besoins et de les réserver.

5.2 L'administrateur

L'administrateur est un utilisateur anodin qui dispose de quelques fonctionnalités supplémentaires. Il s'agit du responsable de la gestion globale du système. Il a accès à tous les comptes des utilisateurs de l'application et a aussi la possibilité de désactiver ou supprimer ces derniers.

6. Diagrammes UML

6.1 Diagramme de cas d'utilisation

Un cas d'utilisation représente un ensemble de séquences d'action qui sont réalisées par le système dont la finalité est de rendre un service à un des acteurs. Les cas d'utilisations relatifs à notre application sont représentés dans la figure ci-dessous :

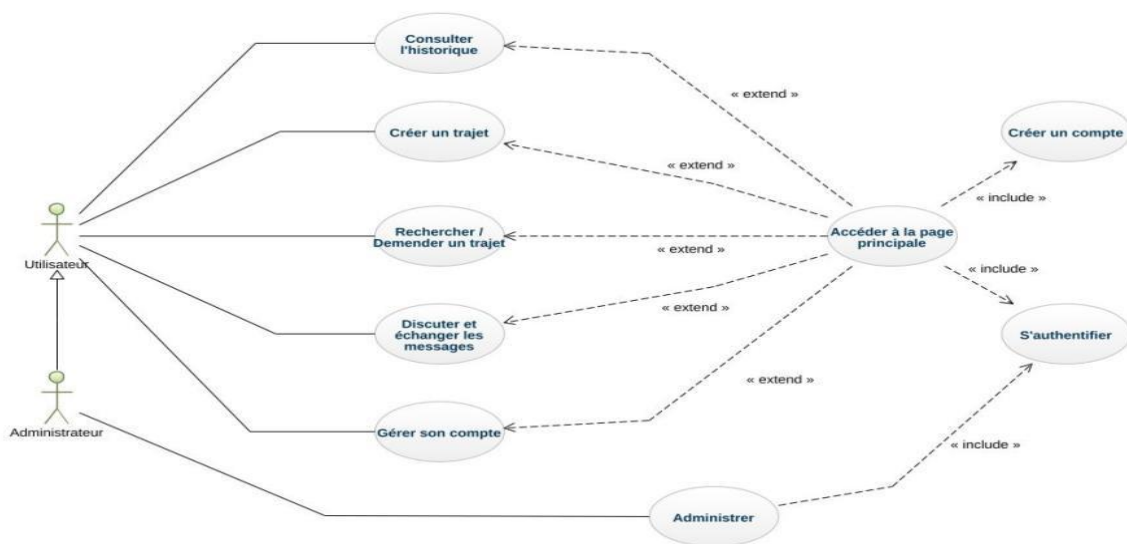


Figure 5: Diagramme de cas d'utilisation

Description du scénario illustré

1. Créer un compte

- L'utilisateur devra télécharger l'application avant son lancement.
- Une interface d'affiche où il devra cliquer sur le bouton "S'inscrire".
- Le système affiche le formulaire d'inscription.

- L'utilisateur remplit les différents champs du formulaire (le système exige à l'utilisateur de s'inscrire avec l'email de l'ESI).
- Le système vérifie les informations de l'utilisateur en envoyant un mail de vérification afin de s'assurer qu'il appartient à l'environnement de l'ESI.

2. Gérer son compte

La gestion d'un compte inclut la suppression et la modification de ce dernier.

- L'utilisateur doit s'authentifier en premier lieu afin d'accéder à l'application.
- Une fois connecté, l'utilisateur accédera à la page d'accueil où se trouve un bouton qui mène vers le profil.
- L'utilisateur est en mesure de modifier son compte en accédant à ses informations personnelles. Le système vérifie par la suite les nouvelles valeurs et les sauvegarde.
- Dans le cas où il souhaite supprimer son compte, il devra accéder aux paramètres de l'application.

3. Authentification

- L'utilisateur lance l'application.
- Il aura le choix de s'authentifier ou bien de s'inscrire.
- Le système affiche le formulaire d'authentification.
- L'utilisateur s'authentifie en saisissant son mail (de la forme @esi.dz) ainsi que son mot de passe.
- Le système vérifie la validité des informations, et affiche la page d'accueil. Dans le cas contraire, un message est affiché demandant à l'utilisateur de ressaisir ses identifiants.

4. Créer un trajet

- Avant de publier un trajet, l'utilisateur devra saisir le point de départ et le point d'arrivée. Etant donné que l'application est dédiée à l'environnement de l'ESI, l'utilisateur pourra sélectionner directement l'adresse de l'école en appuyant sur le logo ESI affiché dans la barre de recherche.
- L'utilisateur devra également saisir la date, l'heure, le prix et le nombre de passagers. Il peut aussi imposer des préférences à ses clients (discussion, animal, musique...etc.).
- Une fois les informations saisies, le système publie l'annonce. La rendant ainsi visible aux utilisateurs de l'application.

5. Consulter l'historique

- Après chaque trajet, les informations de ce dernier sont enregistrées dans l'historique contenant : le nom du conducteur ou passager, le point de départ et le point d'arrivée, la date, l'heure et le prix
- Chaque utilisateur a la possibilité de consulter la liste des trajets parcourus précédemment.

6. Rechercher/Demander un trajet

- L'utilisateur lance l'application et s'authentifie.
- La page d'accueil est affichée où il pourra rechercher un trajet.
- Le client saisit son adresse de départ et d'arrivée. Il peut sélectionner l'adresse de l'ESI en cliquant sur son logo à droite de la barre de recherche, qui remplira automatiquement l'adresse de l'école.
- Une liste des trajets disponibles est affichée, permettant à l'utilisateur de choisir celui qui lui convient le mieux.

- Dans le cas où l'utilisateur trouve un trajet qui répond à ses besoins, il pourra le réserver ou le demander.
- Le système transmet une notification au conducteur choisi suivie des coordonnées du potentiel passager.
- Le conducteur a le choix de valider ou refuser de conduire cet utilisateur.
- En cas de validation, le passager reçoit une notification de validation et peut alors échanger des messages avec le conducteur ainsi que les autres passagers (le système crée un groupe dont les membres sont le conducteur et les passagers du même trajet, afin faciliter l'échange et la communication entre ces derniers). En revanche, si la demande est refusée, l'utilisateur est informé par une notification l'utilisateur reçoit une notification de refus lui proposant de consulter d'autres trajets.

7. Discuter et échanger les messages

Afin d'améliorer l'expérience utilisateur de notre application de covoiturage, nous avons jugé meilleur de créer un groupe de discussion réservée au trajet plutôt qu'un simple chat.

Le système crée un groupe chat associé au trajet dont les membres sont le conducteur et les passagers. Cette fonctionnalité permet de faciliter la communication entre ces derniers.

8. Administrer

- L'administrateur peut recevoir des mails et des appels de la part des utilisateurs.
- Il est en mesure d'accéder aux comptes des utilisateurs et ses informations, ainsi que la capacité de les suspendre.

6.2 Diagramme de classes

Le diagramme de classe constitue l'un des pivots essentiels de la modélisation avec UML. En effet, ce diagramme permet de représenter les classes d'un système, les relations entre ces classes et les différents attributs et méthodes de chaque classe. Il s'agit d'un outil de modélisation statique qui sert à visualiser la structure d'un système, en mettant en évidence les différentes entités qui le composent et les interactions entre elles.

Le diagramme de classes relatif à notre application est représenté dans la figure ci-dessous :

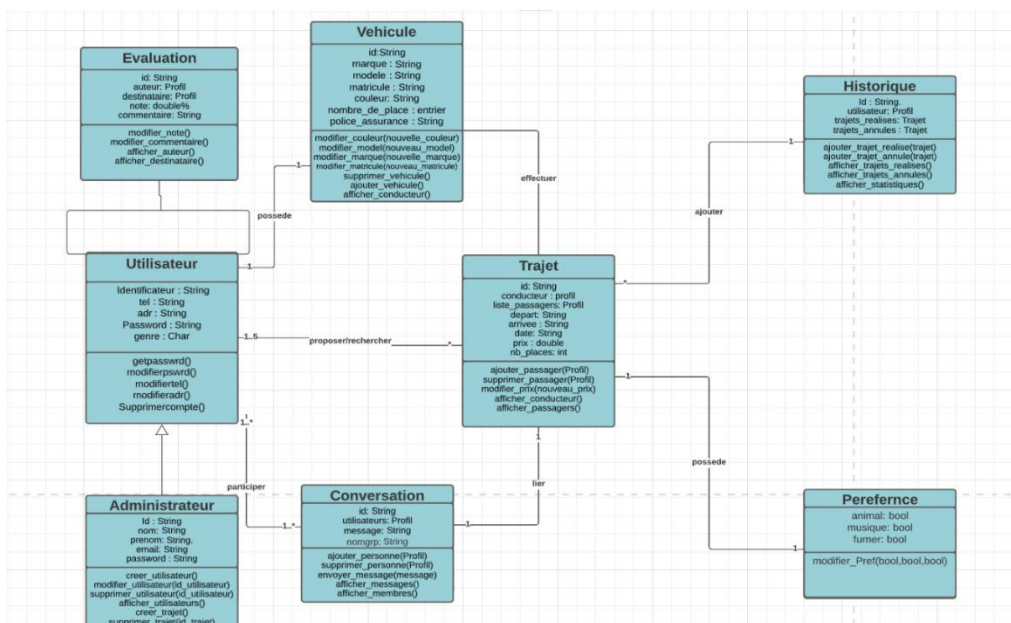


Figure 6: Diagramme de classes

- **Classe utilisateur**

Table 1: La classe Utilisateur

Attribut	Rôle
nom, prénom, tel, genre	Les informations de l'utilisateur (son nom, numéro de téléphone et son genre)
adr	Son adresse email (l'adresse de l'ESI)
passwd	Son mot de passe
Méthode	Rôle
getpassword	Créer son mot de passe
creer_trajet	la creation d'un trajet
rechercher_trajet	la recherche d'un trajet
modifierPasswrđ	Permet la modification du mot de passe (soit pour récupérer le compte ou bien juste une simple modification pour avoir un nouveau mot de passe
modifierTel	Permet la modification du numéro de téléphone
supprimerCompte	la suppression du compte
modifier_information	Permet la modification des informations de l'utilisateur (nom, prénom, téléphone)

- **Classe administrateur**

Un administrateur est un utilisateur avec des fonctionnalités en plus.

Table 2: La classe administrateur

Méthode	Rôle
afficher_utilisateur	Il peut voir le profil de chaque utilisateur à tout moment
supprimer_utilisateur	Désactiver le compte d'un utilisateur

- **Classe véhicule**

Table 3: La classe véhicule

Attribut	Rôle
Id, model, marque, couleur, police d'assurance, matricule	Ce sont les informations du véhicule
Méthode	Rôle
modifier_information	Permet la modification des informations du véhicule (matricule, police assurance, couleur, model)
ajouter_véhicule()	ajouter un nouveau véhicule

- **Classe évaluation**

Table 4: La classe évaluation

Attribut	Rôle
id	L'identifiant de l'évaluation
conducteur	La personne qui va être évalué
passager	La personne qui va évaluer le conducteur

etoile	Combien d'étoiles sur 5
Commentaires	Les commentaires à propos du conducteur
Méthode	Rôle
evaluer ()	Evaluer le conducteur dès qu'un trajet est terminé
afficher_commentaire	Afficher tous les commentaires d'un conducteur

- **Classe Conversation**

Attributs	Rôle
id	L'identificateur de la conversation
utilisateurs	Les utilisateurs dans le trajet (la conversation)
messages	Les messages
nomgrp	Le nom du groupe (la conversation)
Méthode	Rôle
ajouter_personne(Profil)	Ajouter une nouvelle personne dans la conversation
envoyer_message(message)	envoyer un message
afficher_messages()	Afficher les messages d'une conversation
afficher_membres()	Afficher les membres d'une conversation

- **Classe Trajet**

Table 5: Classe Trajet

Attributs	Rôle
id: String conducteur : profil liste_passagers: Profil depart: String arrivee : String date: String prix : double nb_places: int preference : preferences	Les informations du trajet (identifiant, conducteur, passagers, départ, arrivée, le prix, nombre de places disponible
ajouter passager(Profil)	ajouter un passager dans le trajet (seul le conducteur a le droit de le faire)
Méthode	Rôle
afficher_conducteur()	Afficher les informations du conducteur (ainsi que les informations de son véhicule)
afficher_passagers()	Afficher les passagers d'un trajet donné

- **Classe Préférences**

Table 6: La classe préférences

Attribut	Rôle
baggage: bool animaux: bool musique: bool cigarette: bool	Des booléens pour désigner les préférences d'un conducteur (s'il a de la place pour que ses passagers ramènent un bagage ...)
Méthode	Rôle

afficher_Pref(bool,bool,bool)	Afficher les préférences d'un conducteur
-------------------------------	--

- **Historique**

Table 7: La classe historique

Attributs	Rôle
Id : String.	L'identifiant (l'utilisateur)
utilisateur : Profil	L'utilisateur
trajet_planifier : Trajet trajets_realises: Trajet trajets_annules : Trajet	Les trajets d'un utilisateur
Méthode	Rôle
afficher_trajets()	Afficher les informations des trajets (planifier ou bien déjà réalisé)

6.3 Diagrammes de séquences

Les diagrammes de séquence sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique.

Dans le contexte de notre application de covoiturage, nous présentons ci-dessous les diagrammes de séquence qui correspondent aux cas d'utilisation des principales fonctionnalités de notre système.

6.3.1 Diagramme de séquence du cas d'utilisation : 'S'authentifier'

Pour s'authentifier, l'utilisateur doit saisir son adresse mail ESI suivie de son mot de passe. Si le système valide ses informations après une requête au niveau de la base de données, il pourra accéder à son compte, sinon il devra saisir ses coordonnées.

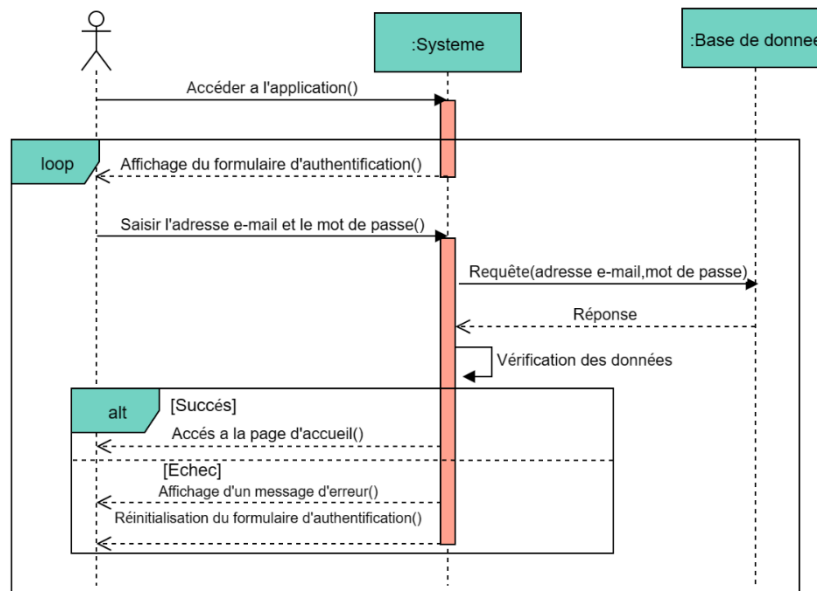


Figure 7:Diagramme de séquence du cas d'utilisation : 'S'authentifier'

6.3.2 Diagramme de séquence du cas d'utilisation : 'S'inscrire'

Si un visiteur souhaite s'inscrire, en lançant l'application il clique sur "SIGN UP".

Un formulaire d'inscription lui est affiché lui demandant de saisir ces informations : nom, prénom, e-mail de l'ESI, numéro de téléphone, statut (étudiant, enseignant, employé), date de naissance, les informations de son véhicule s'il le souhaite.

Une fois ses informations validées par le système, un nouvel utilisateur est créé à la base de données. Le nouvel utilisateur pourra ainsi accéder à l'application et profiter des fonctionnalités qu'elle offre. Dans le cas où ses informations sont incohérentes, il devra remplir le formulaire une nouvelle fois.

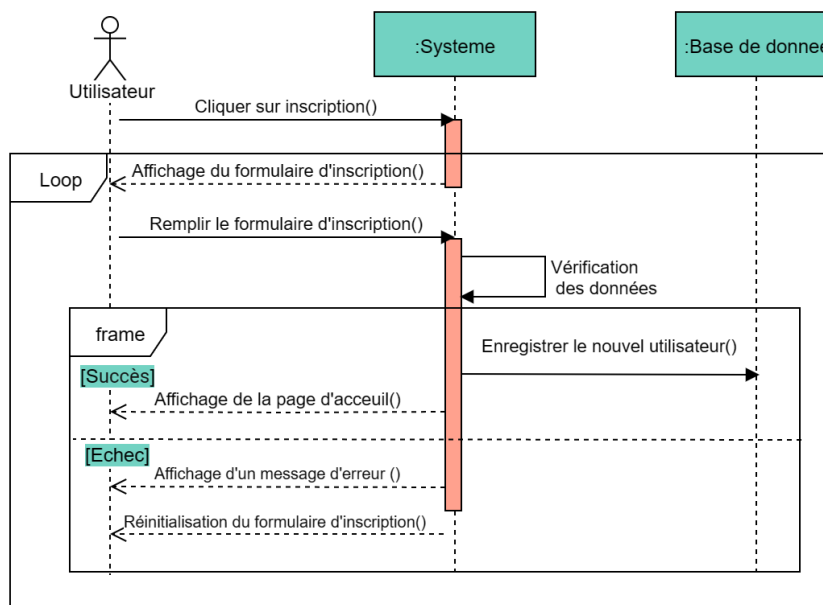
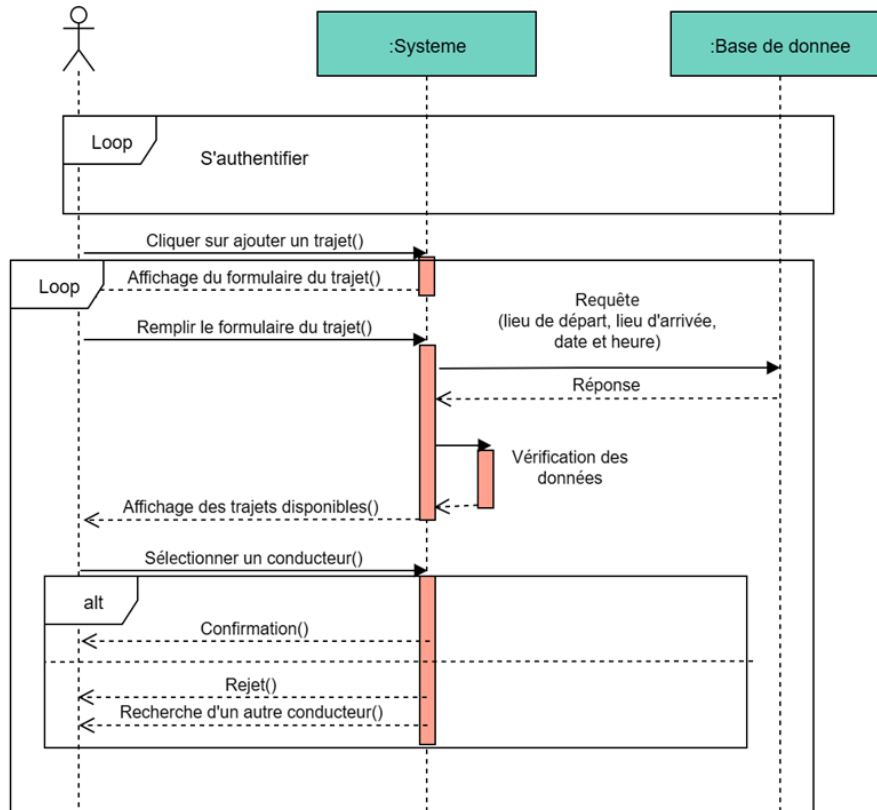


Figure 8:Diagramme de séquence du cas d'utilisation : 'S'inscrire'

6.3.3 Diagramme de séquence du cas d'utilisation : 'Rechercher un trajet'

Après s'être authentifié et avoir accéder à son compte personnel, une barre de recherche est affichée à l'utilisateur en haut de la page d'accueil lui proposant de rechercher un trajet en saisissant l'adresse de départ, la destination ainsi que la date et l'heure de départ. Le système affichera les trajets compatibles aux informations insérées par l'utilisateur.



6.3.4 Diagramme de séquence du cas d'utilisation : 'Publier un trajet'

Pour publier une annonce de trajet, l'utilisateur doit cliquer sur le bouton à droite qu'il peut trouver en dessus de la barre de navigation. Il devra saisir l'adresse de départ, la destination, la date et heure de départ, ses préférences, la méthode de paiement ainsi que le nombre de passagers. Si le système publie ce trajet et l'ajoute dans la base de données si ces informations sont correctes.

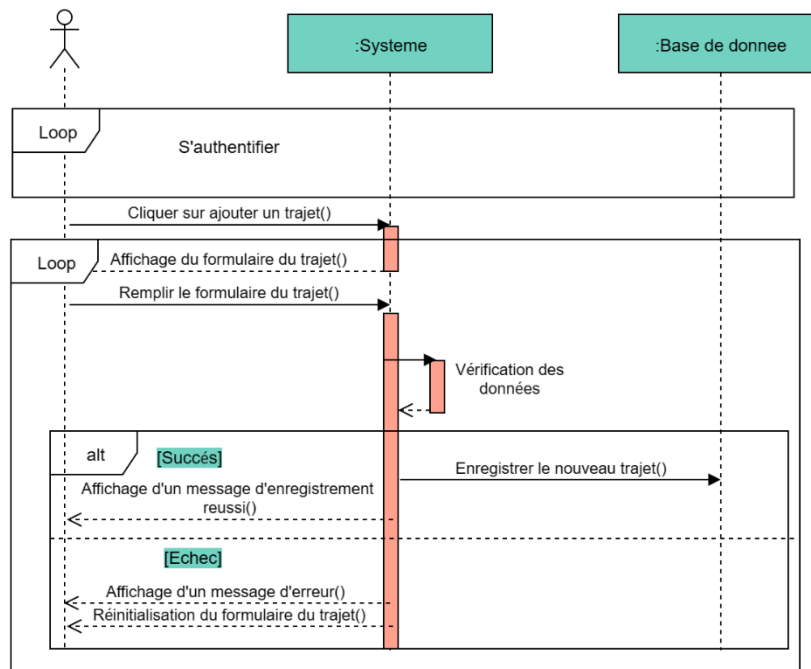


Figure 10: Diagramme de séquence du cas d'utilisation : 'Publier un trajet'

7. Charte graphique et ergonomie de l'application

7.1 Le nom de l'application

Le choix du nom est un élément clé dans l'identité visuelle d'une application. Dans notre cas, nous avons opté pour le nom de 'ESI Way'. Notre équipe a choisi de combiner entre les initiales de notre école 'ESI' avec le mot 'way' qui veut dire 'chemin' en anglais.

Nous voulions un nom qui soit facile à retenir et à prononcer, ainsi qu'un logo qui soit facilement reconnaissable. Nous avons créé un jeu de mots avec le mot 'easy' qui veut dire 'facile' en anglais, et qui évoque l'idée de facilité et de confort et renforce notre lien avec la communauté estudiantine.

7.2 Le logo

Pour le logo on a réalisé un logo simple et minimaliste en combinant la lettre E pour ESI et W pour way, qui sont les initiales du nom de notre application 'ESI Way'.

La lettre E fait aussi référence au compteur de vitesse de la voiture avec son aiguille avec la couleur rose.



Figure 11: Logo et nom de l'application

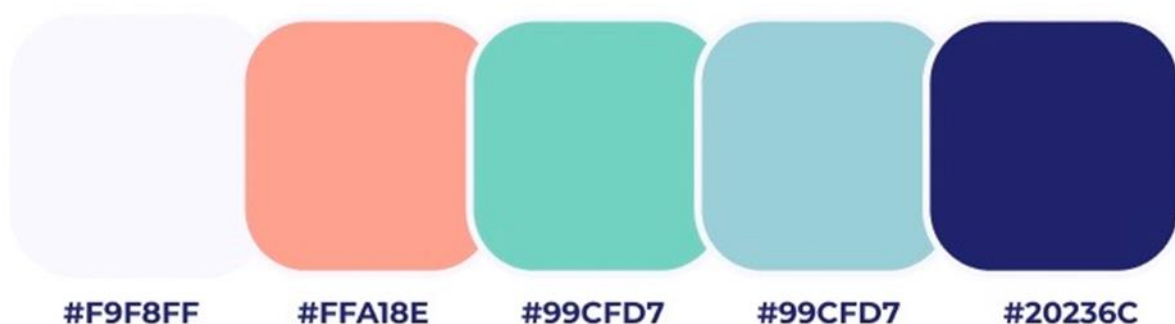
7.3 Palette de couleurs et police

Notre application est majoritairement en rose et en bleu avec ses trois nuances différentes.

Le bleu est souvent associé à la confiance et la sérénité, ce qui est important pour une application de covoiturage où la sécurité est un élément clé.

Le rose quant à lui peut évoquer la joie, l'amitié et la convivialité, cela peut être particulièrement pertinent pour notre application de covoiturage qui implique le partage de trajets entre la communauté de l'ESI.

Concernant la police, nous avons choisi la police Montserrat pour son apparence moderne et épurée, ainsi que pour sa lisibilité.



Montserrat

Figure 12: Palette de couleurs et police

8 Conclusion

Ce chapitre constitue le lancement de la phase de développement, menant une étude fonctionnelle de notre plateforme de mise en relation. Cette étude nous a permis de recenser les besoins et exigences du projet ainsi que les données nécessaires pour le développement de l'application. Dans ce chapitre, nous avons présenté les différentes étapes de la conception, telles que l'analyse du cahier des charges, la description du scénario d'utilisation, les diagrammes UML et la proposition de la charte graphique.

Cette étape conceptuelle constitue une base solide pour le développement de notre application.

Chapitre 03 : Réalisation

1. Introduction

La phase de réalisation représente l'étape de concrétisation du projet.

Après une phase de conception minutieuse, nous pouvons passer à l'étape de réalisation de notre projet. Cette phase est cruciale car elle implique la mise en pratique de tous les éléments évoqués lors de la phase de conception. Elle comprend la programmation, la création de la base de données, l'intégration de l'interface graphique et les tests et corrections des erreurs éventuelles. Nous aborderons également les outils et les technologies utilisés pour la mise en œuvre de l'application.

2. Outils et moyens techniques

2.1 Outils d'implémentation

2.1.1 Visual Code Studio

Visual Studio Code est un éditeur de code développé par Microsoft, conçu pour fonctionner sur Windows, Linux et MacOS. Il offre une grande variété de fonctionnalités, notamment la mise en évidence de la syntaxe, l'achèvement automatique du code source, la refactorisation du code, l'intégration de Git et le débogage. Les utilisateurs peuvent personnaliser leur environnement de travail en modifiant le thème, les raccourcis clavier, les préférences et en ajoutant des extensions pour ajouter des fonctionnalités supplémentaires.



Figure 13: Visual Studio Code

2.1.2 Firebase

Firebase est une plateforme Backend-as-a-service (Baas) proposée par Google qui fournit une suite complète d'outils et de services aux développeurs. Nous avons utilisé Firebase pour implémenter notre base de données et stocker les différentes informations relatives à l'application. Elle offre une variété de fonctionnalités telles que l'authentification des utilisateurs, la base de données en temps réel, le stockage de fichiers, les notifications, l'analyse, la messagerie dans le cloud, la fonctionnalité d'hébergement et bien plus encore.



Figure 14: Firebase

2.2 Outils de design

2.2.1 Adobe Illustrator

Adobe Illustrator est un logiciel de dessin vectoriel destiné à la création d'illustrations, de logos, de motifs textiles, de mises en page et d'autres types de graphismes. Le logiciel utilise des formules mathématiques pour générer des courbes vectorielles qui peuvent être manipulées à l'aide d'outils simples de pointage-cliquetis. Nous avons eu recours à Adobe Illustrator pour la création du logo de ESI Way.



Figure 15: Adobe Illustrator

2.2.2 Figma

Figma est un outil de conception d'interfaces utilisateur (UI) et d'expérience utilisateur (UX) basé sur le cloud. Il permet aux utilisateurs de créer des prototypes interactifs, des maquettes et des designs pour des applications web ou mobiles. Figma est particulièrement populaire pour le travail collaboratif, car plusieurs personnes peuvent travailler simultanément sur un même projet et voir les modifications en temps réel.



Figure 16:Figma

2.3 Outils de communication

2.3.1 Google Drive

Google Drive est un service de stockage en ligne gratuit proposé par Google. Il nous a permis de stocker et de partager des fichiers, y compris des documents tels que les PV, des feuilles de calcul et des présentations. Google Drive propose aussi la sauvegarde automatique, la synchronisation de fichiers, le partage de fichier, l'accès à distance et la collaboration en temps réel.



Figure 17:Google Drive

2.3.2 Google Meet

Google Meet est une plateforme de visioconférence en ligne développée par Google. Nous avons utilisé Google Meet pour organiser des réunions hebdomadaires (au moins une réunion le dimanche de chaque semaine) en plus de la séance en présentiel en présence de nos encadrants, afin de discuter sur l'avancement du projet, des problèmes rencontrés et de répartir les tâches restantes de manière efficace.



Figure 18:Google Meet

2.3.3 Google Sheets

Google Sheets est une application de tableur en ligne. Elle permet de créer, modifier et partager des feuilles de calcul en temps réel avec d'autres utilisateurs. Nous avons utilisé Google Sheets pour organiser nos idées, les tâches à réaliser et aussi pour enregistrer les liens utiles pour la documentation.



Figure 19:Google Sheet

2.3.4 GitHub

GitHub est une plateforme de développement de logiciels qui nous a permis de stocker, gérer et surtout partager les codes sources de notre projet. Il s'agit d'un service web basé sur le cloud qui utilise le système de contrôle de version Git, qui permet aux développeurs de travailler ensemble et de collaborer facilement sur un même projet.

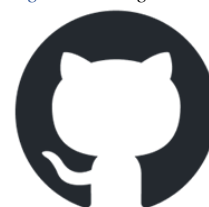


Figure 20:GitHub

2.3.5 Discord

Il s'agit d'une application de communication en ligne qui permet aux utilisateurs de communiquer par chat, appel vocal et vidéo en groupe. Discord nous a été d'une grande utilité car il nous a permis de créer plusieurs discussions relatives aux différentes tâches du projet pour mieux organiser les discussions et retrouver les informations partagées facilement.



Figure 21:Discord

3. Choix des langages de programmation

3.1 Flutter

Notre application de covoiturage a été développée en utilisant le Framework Flutter, qui offre de nombreux avantages pour le développement d'applications mobiles. L'un des principaux avantages de Flutter est sa capacité à créer des interfaces utilisateur modernes et fluides, grâce à son moteur de rendu graphique intégré. En outre, Flutter permet de développer des applications pour les plates-formes iOS et Android à partir d'un même code source, ce qui réduit considérablement le temps de développement. De plus, ils disposent d'un système de widgets personnalisables et préconçus qui peuvent être facilement intégrés pour offrir une meilleure expérience.



Figure 22:Flutter

3.2 Dart

Dart est le langage de programmation utilisé pour développer des applications avec le Framework Flutter. Il est conçu pour être simple, moderne et rapide, avec une syntaxe claire et concise qui permet aux développeurs de produire un code efficace et maintenable. Dart offre également une forte sécurité et une grande stabilité, grâce à son système de typage statique qui permet de détecter les erreurs avant l'exécution du code.



Figure 23:Dart

4. Étapes de la réalisation

4.1 Implémentation de la base de données

Cette étape est cruciale pour notre application de covoiturage car elle nous permet de stocker toutes les informations relatives aux trajets et aux utilisateurs. Nous avons utilisé Firebase pour créer et configurer notre base de données, qui nous donne la possibilité de stocker les données de manière efficace et de les récupérer facilement.

Afin d'assurer la fiabilité des informations stockées dans notre application, nous avons mis en place des mécanismes de vérification rigoureux. Ces mécanismes comprennent notamment la vérification de l'adresse e-mail saisie lors de l'inscription. Cette vérification permet de s'assurer que seuls les utilisateurs disposant d'une adresse e-mail valide de l'Ecole Nationale Supérieure d'Informatique peuvent s'inscrire sur notre application.

Nous avons également défini des normes pour le choix du mot de passe. Ces normes incluent une longueur minimale de 8 caractères ainsi que l'utilisation de chiffres et d'au moins une lettre majuscule et une lettre minuscule.

Nous sommes convaincus que nos mesures de sécurité garantissent la protection des données de nos utilisateurs et la confidentialité de leurs informations. Etant donné que nous avons mis en place des restrictions d'accès strictes pour la base de données. Seul l'administrateur a accès à la base de données, ce qui permet de minimiser les risques de pertes de données ou de violations de la sécurité. Enfin, nous avons testé et validé notre base de données en vérifiant qu'elles sont correctement stockées et récupérées.

Les collections créées pour stocker les informations relatives à notre application sont les suivantes :

The screenshot shows the Redux DevTools state inspection. The state is an object with 'users' and 'chats' arrays. The 'users' array contains one user object, which is expanded to show its properties: Birth, Comments (an array with one comment object), ProfilePicture, Status, Trip, and hasCar. The comment object is also expanded, showing its properties: Comment, Date, FamilyName, Name, CreatedAt, and Email.

```
{
  "users": [
    {
      "Birth": "4-5-2023",
      "Comments": [
        {
          "Comment": "ZOUBIR Mohamed Akran",
          "Date": "12-12-1212",
          "FamilyName": "ZOUBIR",
          "Name": "Zoubir",
          "CreatedAt": "1 mai 2023 à 13:36:47 UTC+1",
          "Email": "Im_zoubir@esi.dz"
        }
      ],
      "ProfilePicture": "https://firebasestorage.googleapis.com/v0/way-34f28.appspot.com/o/images%2FHe9ne8pNjuX7ixQB80X6gZn%2Ftoken=9ba4d8d4-a8cc-4816-8cb5-11e0495...",
      "Status": "Student",
      "Trip": ["Xql5i7F1cPAu0KQVHi9", "..."],
      "hasCar": true
    }
  ],
  "chats": []
}
```

Figure 24: Collection Users

esi-way-34f28

+ Commencer une collection

Cars

Notifications

Trips >

Users

chats

+ Ajouter un champ

Arrivee: "Dwawda, Douaouda, Algeria"

Arrivee_LatLng: "lat: 36.6770431 / longitude: 2.7955922"

Conducteur: "He9ne8pNjuX7ixQB80X6gZmkjjw1"

Date: "28 - 5 - 2023"

Depart: "TGCTP, El Biar, Algeria"

Depart_LatLng: "lat: 36.7602887 / longitude: 3.0268768"

Heure: "22:48"

Places: "4"

Price: ""

methode: ""

Figure 25: Collection Trips

```
esi-way-34f28
+ Commencer une collection

Cars >
Notifications
Trips
Users
chats

CarPicture: "https://firebasestorage.googleapis.com/v1/b/esi-way-34f28.appspot.com/o/Cars%2FHe9ne8pNjuX7ixQ8t206.png?alt=media&token=9a825ce7-d1f4-47d0-b080-900000000000"
Policy: "https://firebasestorage.googleapis.com/v1/b/esi-way-34f28.appspot.com/o/Policy%2FHe9ne8pNjuX7ixQ8t206.png?alt=media&token=8a574d54-7b0a-430b-9420-b080-900000000000"
brand: "Peugeot"
model: "206"
registrationNumber: "12014-112-31"
```

Figure 26: Collection Car

Figure 27:Collection Notification

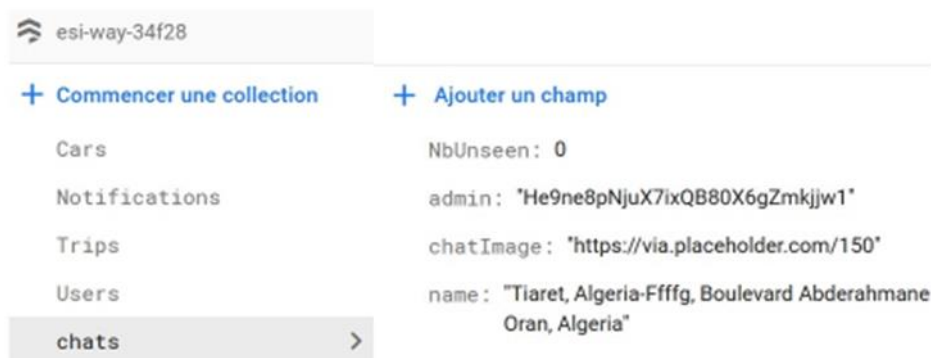


Figure 28:Collection chats

4.2 Les APIs utilisées

4.2.1 Maps SDK for Android API

Il s'agit d'une bibliothèque logicielle développée par Google qui permet de créer des applications Android intégrant les cartes interactives et de les personnaliser en ajoutant des marqueurs, des polygones, des polygones et d'autres éléments graphiques.

4.2.2 Maps SDK for iOS API

L'API fournie par le SDK permet aux développeurs d'ajouter une carte à leurs applications iOS et d'ajouter des marqueurs et d'autres éléments.

4.2.3 Places API

C'est une interface de programmation d'applications fournie par Google qui donne l'accès à des données de localisation et de places à partir de la base de données de Google Maps.

4.2.4 Geocoding API

Google propose à partir de cet API de convertir des adresses en coordonnées géographiques (latitude et longitude), et inversement. (Géocodage et le géocodage inversé).

4.2.5 Géolocalisation API

C'est une interface standardisée qui offre la possibilité d'obtenir la position géographique d'un utilisateur. (Position actuelle).

5. Intégration des fonctionnalités de l'application

Pour la mise en œuvre de notre application de covoiturage, nous avons pris soin d'intégrer avec précisions les différentes pages et fonctionnalités pour offrir aux utilisateurs une expérience fluide et homogène. Nous avons veillé à ce que tous les éléments de l'application soient compatibles les uns avec les autres afin d'assurer une performance optimale.

Nous avons également porté une attention particulière à la conception responsive pour garantir que l'application fonctionne de manière optimale sur toutes les tailles d'écrans, et cela en utilisant les différents outils proposés par Flutter (media query).

6. Tests des fonctionnalités de l'application

Une fois l'intégration terminée, nous avons effectué les tests complets pour nous assurer que l'application fonctionne de manière optimale. Tout d'abord, nous avons effectué les tests unitaires pour tester chaque fonctionnalité de l'application indépendamment des autres. Ces tests nous ont permis de détecter et de corriger rapidement les erreurs dans le code.

Ensuite, nous avons réalisé des tests d'intégration pour tester le fonctionnement des différentes pages de l'application, afin de vérifier qu'elles interagissent correctement les unes avec les autres.

Enfin, nous avons réalisé des tests de convivialité pour s'assurer que l'application est facile à utiliser pour tous les types d'utilisateurs, ce qui nous a permis de détecter les problèmes potentiels d'ergonomie et d'interface utilisateur, et d'y remédier pour garantir une expérience optimale à nos utilisateurs.

Les tests ont été réalisés sur plusieurs appareils pour garantir une compatibilité maximale et une expérience utilisateur optimale pour tous les utilisateurs.

7. Présentation des interfaces principales

7.1 Ecran de chargement

Au lancement de l'application, le logo de ESI Way est affiché, suivi d'une page contenant des slogans pour promouvoir le covoiturage, invitant l'utilisateur à s'inscrire ou à s'authentifier.



Figure 30: Interface Logo

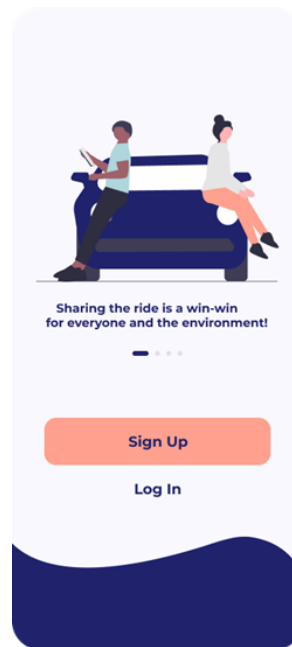


Figure 29: Interface Slogans

7.2 Pages du formulaire d'inscription

Tout visiteur désirant s'inscrire, doit remplir ce formulaire avec ses informations personnelles.

The image shows two mobile app screens for registration. The first screen, titled 'Sign up', has a 'Back' button at the top left. It contains four input fields: 'Email' (with a checkmark icon), 'Password' (with a lock icon), 'Password Confirmation' (with a lock icon), and 'Phone Number'. Below these is an orange 'Next' button with a right arrow. At the bottom, there is a link: 'Already Have an Account? Log In'. The second screen, titled 'Create Profile', also has a 'Back' button. It contains three input fields: 'Name', 'Last Name', and 'Gender' (with 'Male' and 'Female' radio buttons). Below these is a 'Status' section with three buttons: 'Teacher', 'Student', and 'Stuff'. At the bottom is an orange 'Next' button with a right arrow. Both screens have a decorative header with a road and car illustration.

Figure 31:Formulaire d'inscription

7.3

Page des informations de la voiture

Si l'utilisateur possède une voiture et souhaite créer de futurs trajets, un formulaire lui est affiché pour saisir les différentes informations relatives à son véhicule. Dans le cas où il ne possède pas de voiture, il peut accéder à page suivante en cliquant sur "Skip" en dessous du bouton "Next".

The image shows a mobile app screen titled 'Car Information'. It has a 'Back' button at the top left. The form contains two input fields for 'Brand' and 'Model', each with a car icon and the placeholder text 'Example'. Below these is a 'Registration Number' input field with a car icon and the placeholder text 'Enter your car's registration'. The next section is 'Car's Picture', which has two buttons: 'Upload' (with a camera icon) and 'Take' (with a camera icon). Below that is an 'Insurance policy' section with an 'Upload' button (with a camera icon). At the bottom is an orange 'Next' button with a right arrow. Below the 'Next' button is a link: 'Not ready now? Skip'. The screen has a decorative header with a road and car illustration.

Figure 32:Interface informations de la voiture

7.4 Page d'accueil

Une fois inscrit ou connecté à son compte, l'utilisateur aura accès à toutes les fonctionnalités de l'application. Il pourra ainsi créer ou rechercher des trajets et avoir accès à ses informations personnelles.



Figure 33: Interface page d'accueil

7.5 Page de création d'un trajet

Tout utilisateur possédant une voiture est en mesure de créer des annonces de trajets. Si le formulaire d'informations du véhicule n'a pas été rempli, l'utilisateur ne pourra pas créer de trajets. Dans le cas où il possède un véhicule, il devra remplir le formulaire affiche en saisissant le point de départ, la destination, la date et heure de départ, le mode de paiement, ses préférences et enfin le nombre de places disponibles dans sa voiture. Une fois le formulaire validé par le système, le trajet sera créé et publié.

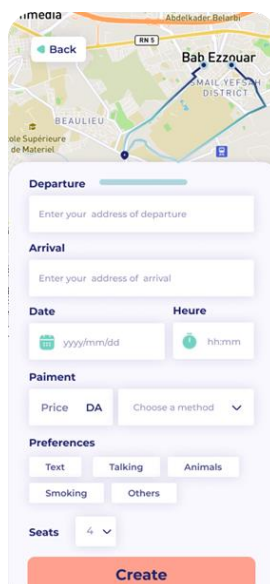


Figure 34: Interface créer un trajet

7.6 Pages recherche d'un trajet

Afin d'effectuer une recherche d'un trajet, un pop-up surgit contenant un formulaire où le potentiel passager devra saisir le lieu, la date et l'heure de départ ainsi que sa destination. Ensuite, une liste des personnes pouvant effectuer au moins 40% de son trajet lui est affichée, avec les informations de chaque trajet et du conducteur.

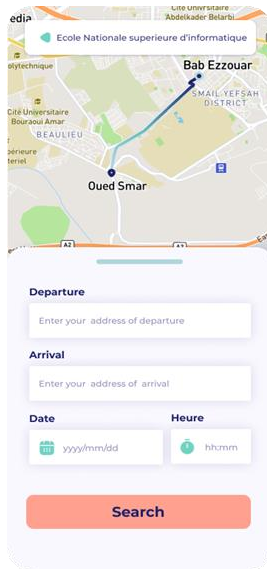


Figure 36: Interface de formulaire de recherche

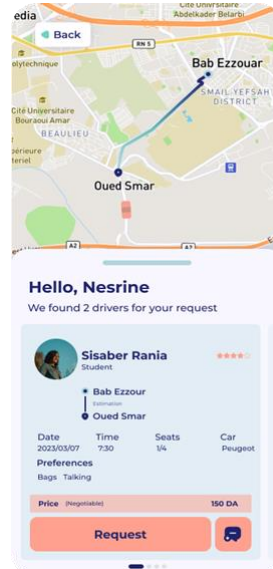


Figure 35: Interface des résultats de recherche

7.7 Page messagerie

La particularité de la messagerie qu'offre notre application est une discussion de groupe. Lorsqu'un conducteur accepte la demande de covoiturage d'un utilisateur, ce dernier est ajouté au groupe de discussion dont les membres sont les passagers et le conducteur. Pour distinguer le conducteur des passagers, ses messages apparaissent dans des bulles roses, tandis que les messages des autres utilisateurs seront de couleur bleu.

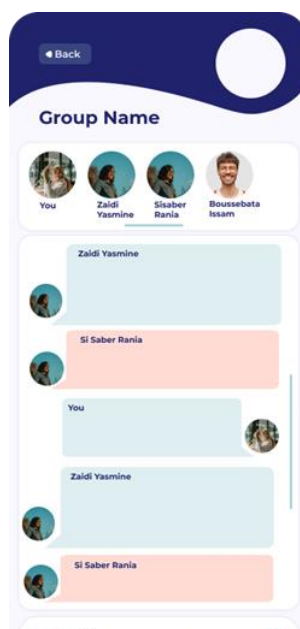


Figure 37: Interface Chat

7.8 Page profil

Tout utilisateur disposant d'un compte ESI Way, est en mesure d'accéder à ses informations personnelles et de les modifier.

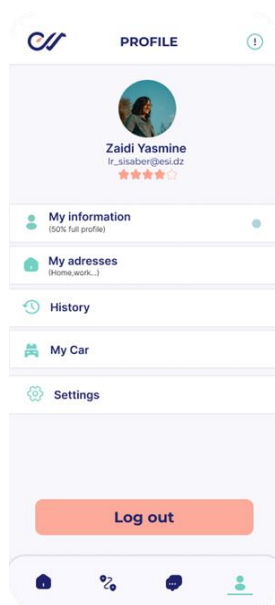


Figure 38: Interface Profil

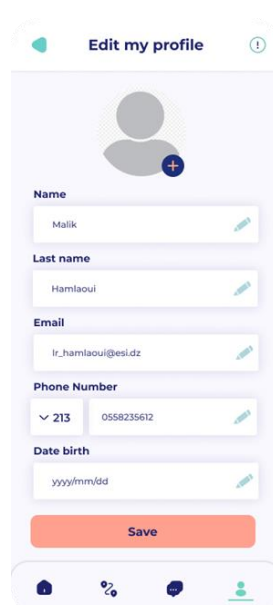


Figure 39: Interface modifier profil

7.9 Page historique

L'utilisateur peut consulter la liste des trajets qu'il a effectués dans l'historique, accompagnés des informations relatives à ces derniers. Les trajets effectués en tant que passager seront affichés en bleu, tandis que les trajets effectués en tant que conducteur seront affichés en rose pour mieux organiser l'historique.



Figure 40: Interface historique

7.10 Page notifications

L'utilisateur est notifié par le système de ESI Way instantanément lors de la modification d'un trajet, validation ou refus d'une demande de covoiturage et toute mise à jour relative aux trajets.

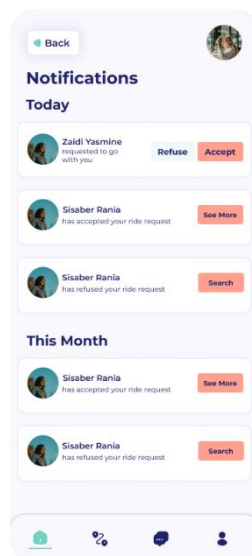


Figure 41: Interface notifications

7.11 Page Mes trajets

A travers cette page, l'utilisateur pourra connaître les trajets en cours et les trajets réservés. En tant que conducteur, le système lui affichera également les annonces de trajets qu'il a proposé (trajets créés).

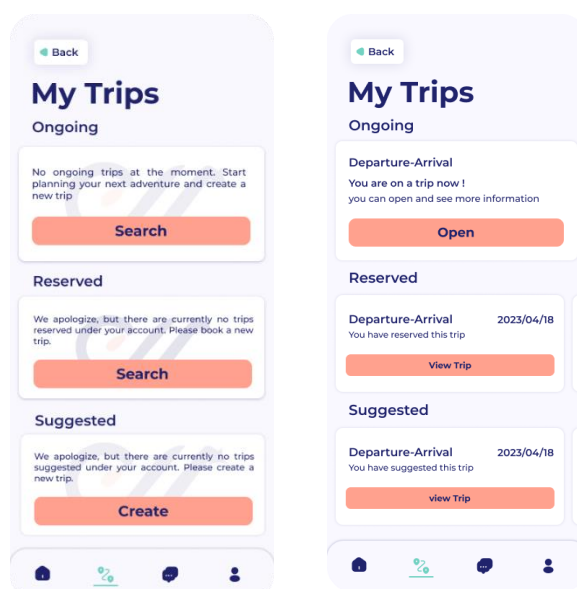


Figure 42: Interface Mes trajets

7.12 Page évaluation et commentaires

Un pop-up apparaît automatiquement à la fin de chaque voyage, permettant à l'utilisateur, qu'il soit passager ou conducteur, d'évaluer l'autre utilisateur et laisser un commentaire s'il le souhaite. Cette fonctionnalité est importante car elle permet d'éliminer les utilisateurs non fiables et de garantir la sécurité de nos clients.

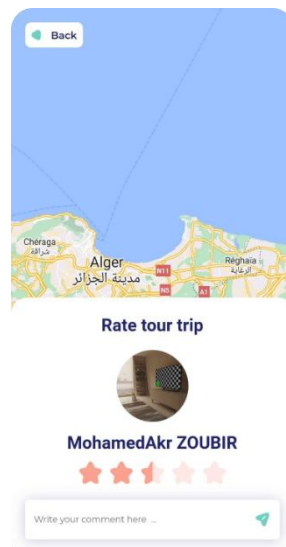


Figure 43: Interface évaluations et commentaires

7.13 Page Administrateur

L'administrateur peut bénéficier de ses droits en accédant à son espace par cliquer sur le bouton "Administrator", pour consulter les différentes informations sur les utilisateurs et les trajets.

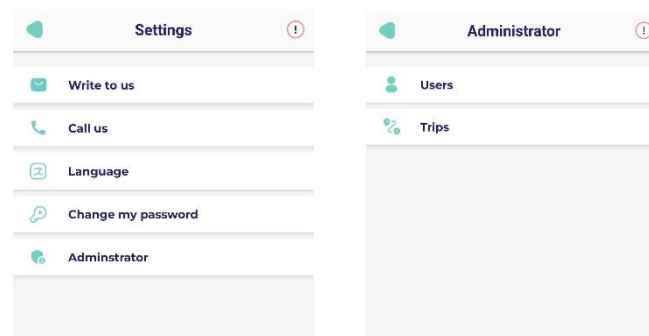


Figure 44: Compte administrateur

8 Conclusion

Dans ce chapitre, nous avons décrit les aspects pratiques liés à la réalisation de notre application ESI Way, à savoir l'environnement, les outils de développement, ainsi que les langages utilisés. Nous avons aussi présenté les principales interfaces qui constituent notre application.

Conclusion

1. Conclusion générale

En conclusion, notre projet d'application de covoiturage a été un véritable défi dans lequel nous avons travaillé en équipe pendant 10 semaines pour concevoir, développer et tester une application permettant de mettre en relation des personnes souhaitant avoir recours au covoiturage.

La réalisation de ce projet a été une expérience enrichissante et stimulante. Il nous a permis de nous immerger dans le rôle d'un ingénieur travaillant en équipe et d'acquérir des connaissances dans le domaine de la programmation, et de confronter nos connaissances en conception logicielle.

D'un point de vue organisationnel et relationnel, nous avons appris à suivre un planning bien défini, à diviser les tâches, à communiquer efficacement, à collaborer professionnellement et à gérer les imprévus qui peuvent survenir lors de la réalisation d'un projet.

Nous avons également acquis de l'expérience dans la rédaction de documents importants tels que les procès-verbaux, les rapports et les différents manuels, qui sont essentiels pour documenter le travail accompli.

Ce rapport vise à exposer en détail les étapes de conception et développement que nous avons suivies pour créer notre application.

Nous avons commencé par une étude approfondie du cahier des charges, afin de bien comprendre les attentes des encadreurs et les besoins de nos utilisateurs potentiels. Suivie par plusieurs recherches sur les solutions existantes pour collecter des informations pertinentes et ainsi souligner les objectifs de notre projet.

Nous avons ensuite enchaîné avec la conception du projet en réalisant la charte graphique de l'application, un premier prototype et un découpage modulaire de notre système.

La dernière partie a fait figure de présentation des outils utilisés pour la concrétisation de notre travail, suivie d'une présentation des interfaces principales de ESI Way.

Notre équipe espère que ESI Way sera accueillie favorablement par les utilisateurs et qu'elle contribuera à promouvoir l'utilisation du covoiturage et à offrir une solution pratique, économique et écologique aux problèmes de déplacement de la communauté de l'Ecole Nationale Supérieure de l'Informatique d'Alger.

2. Perspectives

Les perspectives de notre application de covoiturage sont prometteuses. En effet, on souhaiterait intégrer les fonctionnalités suivantes :

- L'utilisation de mécanismes de paiement en ligne.
- L'intégration d'autres types de transport.
- Inscription par Google.
- Une extension pour ESI-SBA et ESTIN a Bejaia.
- Une version en plusieurs langues.
- Intégration du suivi GPS.

Annexe : Les défis rencontrés

Durant les dix semaines du projet, notre équipe a rencontré plusieurs défis, dépassés à l'aide des conseils précieux de nos encadreurs et le travail en groupe pour obtenir la solution finale. Grâce à ces défis nous avons vécu une bonne expérience d'ingénierie. Parmi ces défis on site :

1. Les applications modèles

Ce défi est l'un des plus difficiles si ce n'est pas le plus dur durant la conception de notre application, le manque d'exemple réel, et le peu d'application de covoiturage disponible, a rendu la tâche très dure pendant la conception de l'application et l'identification des différents cas d'utilisation, ce défi n'est pas juste un défi de codage ou réalisation, mais un défi d'idées. L'implémentation de telle application est un exploit vu le nombre de références et le manque des moyens de documentation. L'équipe ESI Way, à l'aide des conseils des encadreurs et l'analyse du besoin des différents membres de la communauté de l'école, on commencer le projet et le finir tout en respectant les délais.

2. API Google Map

Notre application utilise les cartes pour manipuler les trajets et les localisations, et cela est obtenu à l'aide de l'API Google Map, qui n'est accessible que par une carte VISA, créant ainsi un retard de réalisation liés à l'attente de la carte de paiement, c'est l'un des principaux défis de notre équipe, malgré l'opportunité d'avoir d'autre API, mais le manque de ressources et le peu de services proposés, nous a poussé à opter pour l'utilisation des services de Google Map.

3. Notifications

ESI Way, notre application, est basé sur le bon partage d'information entre les utilisateurs qu'ils soient conducteurs ou passagers, et notifier toute mise à jour concernant la création des trajets ou bien les demandes de covoiturage aux personnes concernées. Cette fonctionnalité dite notifications, est très utile, son importance le rendu inaccessible gratuitement, en d'autres termes on doit payer pour bénéficier de ce service en utilisant une carte VISA, indisponible pour le membre de l'équipe, qui ont dû retarder la réalisation du système de notifications, en attente de trouver une solution gratuite, efficace et légale. Cela à créer une nouvelle piste pour obtenir une application de covoiturage avec un système de notifications fonctionnel.

Bibliographie

- Mémoire (Conception et réalisation d'une application de géolocalisation) par : TATOULTE Hassen, LOMANI Jugurtha.
- Mémoire (Conception et réalisation d'une application web Java EE pour la gestion du covoiturage en Algérie) par : DAICH Hicham, BELAIDABI Ben Youcef.
- Mémoire (Développement d'une application ANDROID pour le partage des évènements sur Google Map) par : KEHLI Assia, BENNEZAIR Nadjat.
- Mémoire (Réalisation d'une application basée agents pour la réservation de taxi) par : Djakour Safa, Yessad Khatima.
- Mémoire (Conception et réalisation d'une application mobile multiplateformes pour le transport routier de personnes et de marchandises en temps réel) par : BOUTERFA Sarah, HADID Zahia.
- Livre (tutorials point: simply easy learning).
- Livre (Flutter for beginners).
- Livre (Learn google flutter fast 65 Exemples Apps).
- Livre (Practical Flutter Improve your Mobile Development with Google's Latest Open-Source SDK).
- Livre (Apprenez firebase).
- Livre (Flutter Succinctly).
- Rapport (Diagramme UML concernant le covoiturage).
- Chaine YouTube "The Net Ninja", playlist "[Flutter Tutorial for Beginners - YouTube](#)".
- Chaine YouTube "The Net Ninja", playlist "[Flutter & Firebase App Build - YouTube](#)".
- Site officiel de Firebase : "[Firebase Documentation \(google.com\)](#)".
- Site officiel de Flutter : "[Flutter documentation | Flutter](#)".
- Site officiel de Dart : "[Dart programming language | Dart](#)".
- Site officiel des Packages pour Flutter : "[Dart packages \(pub.dev\)](#)".
- Site officiel de BlaBlaCar : "[BlaBlaCar](#)".
- Site officiel de Karos : "[Karos, Court-voiturage](#)".
- "[Understanding NoSQL Data Modeling Techniques {15 Techniques} \(phoenixnap.com\)](#)".
- Google Map Flutter: "[Adding Google Maps to a Flutter app](#)".
- Flaticon: "<https://www.flaticon.com>".
- "[Le covoiturage en France, ses avantages et la réglementation en vigueur](#)".
- "[Ce qu'il faut savoir sur le covoiturage | economie.gouv.fr](#)".
- "[Vector Icons and Stickers - PNG, SVG, EPS, PSD and CSS \(flaticon.com\)](#)".
- "[Wikipedia](#)".