



République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université des Sciences et de la Technologie Houari Boumediene



Faculté d'Informatique

Mémoire de Licence

Filière : Informatique

Spécialité :

ACAD

Thème :

**Conception et réalisation d'une application mobile pour
L'optimisation des trajets de livraison**

Sujet proposé par :

M.Rassoul Hamza

Réalisé et présenté par :

M.Elaidat Mohamed Redha

M.Azzi Adel

Promotion :2023 /2024

Remerciements

Ce travail est l'aboutissement d'un semestre d'efforts acharnés et d'agréables défis vécus en une période particulièrement difficile. Il n'aurait sans doute pu être concluant sans l'incalculable sollicitude de nombreuses personnes pour lesquelles nous voulons exprimer notre sincère gratitude.

Avant tout remerciement, louange à Dieu. Nous remercions ALLAH, gloire à lui pour nous avoir accordé lucidité, vigueur et constance durant notre projet. De nous avoir donné le courage, la volonté et la patience pour mener à terme le présent travail. Au premier rang, nous adressons nos plus sincères remerciements pour notre encadrante M. Rassoul Hamza pour avoir accepté de nous encadrer, de nous avoir proposé un sujet, de sa disponibilité, son aide et ses précieux conseils tout au long du travail.

Nous exprimons notre reconnaissance envers les membres du jury pour avoir accepté d'examiner notre travail et de l'enrichir par leurs remarques et propositions.

Nous ne pourrions omettre d'exprimer notre gratitude infinie envers nos parents qui ont été une source de soutien et de courage et qui ont montré une grande patience face à nos fréquentes absences et notre ardeur au travail.

Et pour finir, un grand merci à toutes les personnes qui nous ont soutenues, de près ou de loin, et qui ont participé à la réalisation de ce mémoire.

Table des matières :

1. Chapitre 01 : Etude de l'existant

Introduction	10
1.1. La livraison	10
1.2. Évolution du secteur de la livraison	10
1.2.1. Méthodes traditionnelles de livraison	10
1.2.2. Développement du transport ferroviaire et maritime	11
1.2.3. Introduction des services postaux	11
1.2.4. Émergence des entreprises de messagerie	11
1.2.5. Numérisation du secteur de la livraison	11
1.2.6. Émergence de la livraison de repas et de colis à la demande	12
1.3. Les applications mobiles dans la livraison	12
1.3.1. Fonctionnalités des applications mobiles de livraison	12
1.3.2. Avantages pour les entreprises de livraison	12
1.3.3. Avantages pour les consommateurs	13
1.3.4. Tendances émergentes dans le domaine de la livraison mobile	13
1.4. Présentation des application existantes	14
1.4.1. FedEx	14
1.4.2. DHL	16
1.4.3. Uber Delivery	17
1.5. Fonctionnalités et avantages et inconvénients des application existantes	18
Conclusion	19

2. Chapitre 02 : Conception

2.1. Introduction	20
2.2. Le langage de modélisation « UML »	20
2.3. Diagramme de cas d'utilisation	21
2.3.1. Définition	21
2.3.2. Modélisation	22
2.4. Diagramme de séquence	25
2.4.1. Définition	25
2.4.2. Modélisation	25
2.4.3.	30

Table des matières :

2.5 Diagramme de classe	30
2.5.1 Définition	30
2.5.2 Modélisation	30
Conclusion	31
3 Chapitre 03 : Implémentation	
Introduction	32
3.5 Environnement de travail	32
3.5.1 Outil de développement	32
3.5.2 Les langages utilisés	33
3.6 Présentation de l'application mobile	34
Conclusion	45
3. Annexe :	
3.1. Diagramme de cas d'utilisation	50
3.2. Diagramme de séquence	56

Liste des figures :

Figure 1.1 fonctionnalité de l'app FedEx	14
Figure 1.2 fonctionnalité de l'app FedEx	15
Figure 2 fonctionnalité de l'app DHL Express	16
Figure 3 fonctionnalité de l'app Uber Delivery	17
Figure 4 cas d'utilisation (en général)	22
Figure 5 cas d'utilisation (cas de gestion livreur)	23
Figure 6 cas d'utilisation (cas de gestion client)	24
Figure 7 diagramme de séquence (cas de sign up Client)	25
Figure 8 diagramme de séquence (cas de sign up Livreur).	26
Figure 9 diagramme de séquence (cas de log in).	27
Figure 10 diagramme de séquence (cas de création une commande).	28
Figure 11 diagramme de séquence (cas de modification d'une commande).	29
Figure 12 diagramme de séquence (cas de supprimer d'une commande).	29
Figure 13 diagramme de classe (gestion de livraison).	30
Figure 14 page d'accueil de l'application mobile	36
Figure 15 page d'inscription de l'application mobile (cas d'inscrit comme client).	37
Figure 16 page d'inscription de l'application mobile (cas livreur)	38
Figure 17 page d'inscription de l'application mobile (cas d'inscrit comme un livreur étape 2 entre les informations de véhicule de livreur)	39
Figure 18 page de création d'une commande de l'application mobile.....	40
Figure 19 page de gestion les chemins de l'application mobile	41
Figure 20 page de gestion commande de l'application mobile	42
Figure 21 page de profil d'utilisateur de l'application mobile	43

L'informatique, de nos jours, est bien plus qu'une simple technologie. Elle représente un pilier fondamental de notre société moderne, façonnant non seulement nos interactions quotidiennes, mais aussi nos modes de travail, de divertissement, et même notre manière d'envisager le monde qui nous entoure. À travers son évolution constante, l'informatique a profondément transformé notre façon de vivre, de communiquer et d'interagir avec notre environnement.

Depuis ses débuts modestes avec les premiers ordinateurs, l'informatique a parcouru un chemin extraordinaire. Des machines massives et encombrantes des premières générations, nous sommes passés à des appareils de plus en plus compacts, puissants et omniprésents. Cette évolution a été alimentée par une combinaison de progrès technologiques, d'innovations matérielles et logicielles, ainsi que par une demande croissante de solutions informatiques adaptées à nos besoins quotidiens.

Dans ce paysage technologique en constante mutation, les applications mobiles ont émergé comme des outils indispensables, redéfinissant la manière dont nous interagissons avec la technologie. Les smartphones et les tablettes, équipés d'une multitude d'applications, offrent une porte d'entrée pratique vers un monde d'informations, de divertissements et de services à portée de main, littéralement.

Les applications mobiles, qu'elles soient dédiées aux réseaux sociaux, à la productivité, au divertissement, à la santé ou à d'autres domaines, simplifient et facilitent la vie des utilisateurs de plusieurs manières :

- 1. Accessibilité accrue :** Les applications mobiles permettent un accès instantané à une gamme diversifiée de services et d'informations, n'importe où et à tout moment, tant que l'utilisateur dispose d'une connexion internet.
- 2. Utilisation intuitive :** Les interfaces conviviales et les fonctionnalités intuitives des applications mobiles rendent leur utilisation accessible même aux utilisateurs novices en informatique.
- 3. Personnalisation :** Les applications mobiles offrent souvent des fonctionnalités de personnalisation, permettant aux utilisateurs de configurer les applications en fonction de leurs préférences et de leurs besoins spécifiques.
- 4. Optimisation du temps :** De nombreuses applications mobiles proposent des fonctionnalités conçues pour optimiser la productivité et économiser du temps, que ce soit pour la gestion des tâches, la communication ou l'accès à des informations pertinentes.

5. Mobilité et flexibilité : Grâce à leur nature mobile, les applications permettent aux utilisateurs de rester connectés et de profiter de leurs fonctionnalités quel que soit leur emplacement, ce qui renforce leur efficacité dans notre monde de plus en plus mobile.

En somme, les applications mobiles ont profondément simplifié et amélioré la vie quotidienne des utilisateurs en offrant un accès rapide, intuitif et personnalisé à une gamme étendue de services et d'informations. Leur rôle dans l'évolution de l'informatique est indéniable, et leur impact continu sur notre façon de vivre et d'interagir avec la technologie est appelé à se renforcer dans les années à venir.

L'avènement de **l'informatique** et **des applications mobiles** a considérablement révolutionné divers secteurs d'activité, y compris celui de la livraison. Les applications mobiles dédiées **à la gestion des livraisons** ont transformé la manière dont les entreprises opèrent dans ce domaine, offrant des avantages tant pour les fournisseurs de services que pour les clients. Cet essor technologique a permis d'optimiser les processus de livraison, de réduire les délais, et d'améliorer la satisfaction client. Toutefois, malgré ces avancées, l'adoption de ces technologies a été freinée dans certaines régions, notamment en Algérie, en raison de divers facteurs. Dans cette optique, il est essentiel d'explorer l'impact de l'informatique, en particulier les applications mobiles, sur le secteur de la livraison, ainsi que les défis rencontrés dans leur implémentation et les moyens d'optimiser les processus pour améliorer l'efficacité des livraisons, réduire les délais et garantir la satisfaction des clients.

Les applications mobiles dédiées à la gestion des livraisons ont révolutionné l'ensemble du processus, de la commande à la livraison finale. Elles offrent une plateforme centralisée où les clients peuvent passer des commandes, suivre en temps réel le statut de leur livraison et fournir des commentaires instantanés. Pour les livreurs, ces applications offrent une navigation simplifiée vers les destinations, des itinéraires optimisés en fonction du trafic en temps réel, et une communication directe avec les clients. Cette technologie a également permis aux entreprises de livraison de mieux gérer leur flotte de véhicules, d'optimiser les itinéraires pour réduire les kilomètres parcourus et les coûts opérationnels, tout en garantissant une plus grande transparence et traçabilité des colis.

Malgré les avantages évidents des applications mobiles dans le secteur de la livraison, leur adoption a été entravée dans certaines régions, notamment en Algérie. Plusieurs facteurs contribuent à ce retard, notamment les infrastructures de télécommunications sous-développées, les réglementations gouvernementales restrictives, et le manque de sensibilisation et de formation des acteurs du secteur. De plus, des défis liés à la sécurité des données et aux paiements en ligne peuvent également dissuader les entreprises et les clients d'adopter ces technologies. Ces obstacles ont ralenti l'essor des applications mobiles de livraison en Algérie, limitant ainsi les possibilités d'optimisation des processus et d'amélioration de la satisfaction client.

Malgré les défis, l'intégration des applications mobiles dans le secteur de la livraison offre un potentiel considérable pour optimiser les processus et améliorer l'expérience globale. En facilitant la gestion des commandes, la planification des itinéraires et la communication avec les clients, ces applications permettent aux livreurs d'optimiser leur temps et de réduire les retards. De plus, en offrant des fonctionnalités telles que le suivi en temps réel des livraisons et la possibilité pour les clients de fournir des commentaires instantanés, ces applications améliorent la transparence et renforcent la confiance des clients dans le service de livraison. En fin de compte, en optimisant les processus, en réduisant les délais et en garantissant la satisfaction client, les applications mobiles contribuent à moderniser le secteur de la livraison et à répondre aux besoins changeants des consommateurs dans un monde de plus en plus connecté.

Cela peut affecter et endommager la relation entre un client et livreur. Par conséquent livreur doit fournir une Plateforme accessible via Internet.

Notre travail consiste à englober et contrôler tous ces problèmes dans un seul environnement simple et efficace (Application). Ce mémoire est divisé en trois chapitres :

- 1- Le premier chapitre va s'articuler sur la définition de quelques termes concernant les applications mobile et la gestion de livraison et l'étude des applications mobile existantes, en les critiquant et définissant leurs avantages et inconvénients .
- 2- Le deuxième chapitre sera consacré à l'étude conceptuelle, en utilisant le langage de Modélisation « UML » .
- 3- Dans le troisième chapitre, on parle des différents outils utilisés lors de la réalisation de notre projet, avec des captures d'écran et des explications détaillées .

Introduction

Avant de proposer toute solution dans n'importe quel domaine, une étude préalable qui touche les différents concepts des domaines visés et qui s'intéresse aux solutions déjà proposées pour la problématique traitée est indispensable. Cette étude nous permet de nous introduire au domaine et de connaître ce qui existe comme solution pour qu'elle ne soit pas répétée et pour qu'on se différencie. Dans ce premier chapitre nous allons prendre en connaissance les termes nécessaires pour la compréhension du sujet ainsi que des généralités sur la location, les agences et les véhicules. Nous discuterons aussi sur les solutions informatiques déjà présentes

1.1 La livraison :

La livraison est le processus de transfert de biens ou de services d'un expéditeur à un destinataire, généralement dans un contexte commercial. Cela peut impliquer la distribution de marchandises physiques, de produits alimentaires, de colis, de courrier, voire de services à domicile tels que la restauration ou le transport. Les moyens de livraison peuvent varier, allant des transporteurs traditionnels aux services de messagerie spécialisés ou à la livraison entre particuliers [1]

1.2 Évolution du secteur de la livraison :

L'évolution du secteur de la livraison a été marquée par des changements significatifs au fil du temps, reflétant les avancées technologiques, les évolutions sociales et les demandes croissantes des consommateurs. Voici une illustration simplifiée de cette évolution :

3.6.1 Méthodes traditionnelles de livraison :

- Historiquement, la livraison était effectuée principalement de manière manuelle et locale. Les commerçants livraient souvent leurs produits à pied, en charrette ou à dos d'animal, répondant aux besoins de proximité des clients.

3.6.2 Développement du transport ferroviaire et maritime :

Avec l'avènement du transport ferroviaire et maritime au cours du XIXe siècle, les entreprises ont pu étendre leurs activités de livraison sur de plus longues distances. Les marchandises pouvaient être transportées plus rapidement et à moindre coût, facilitant ainsi le commerce interrégional et international.

3.6.3 Introduction des services postaux :

Les services postaux ont joué un rôle crucial dans l'acheminement du courrier et des colis, offrant une solution centralisée et réglementée pour la livraison. Les systèmes postaux ont évolué pour inclure des services de suivi et des options de livraison express pour répondre aux besoins des entreprises et des consommateurs.

3.6.4 Émergence des entreprises de messagerie :

Au XXe siècle, des entreprises de messagerie privées ont commencé à émerger, offrant des services de livraison rapides et spécialisés pour les entreprises et les particuliers. Ces entreprises ont introduit des innovations telles que la livraison le jour même et les options de suivi en temps réel, répondant ainsi à la demande croissante de rapidité et de commodité.

3.6.5 Numérisation du secteur de la livraison :

Avec l'avènement de l'internet et des technologies mobiles au tournant du XXIe siècle, le secteur de la livraison a connu une transformation majeure. Les entreprises ont commencé à utiliser des systèmes informatisés pour gérer leurs opérations de livraison, tandis que les applications mobiles ont été introduites pour offrir des options de livraison à la demande et des fonctionnalités de suivi en temps réel.

3.6.6 Émergence de la livraison de repas et de colis à la demande :

Plus récemment, l'émergence de plateformes de livraison de repas et de colis à la demande a révolutionné l'industrie de la livraison. Des entreprises comme Uber Eats, Deliveroo et Amazon et FedEx ont introduit des modèles de livraison basés sur la gig economy, offrant des options de livraison rapides et flexibles pour répondre aux besoins changeants des consommateurs.

Cette évolution du secteur de la livraison démontre l'impact profond des progrès technologiques et des changements socio-économiques sur la manière dont les biens sont transportés et livrés, et souligne l'importance de l'adaptabilité et de l'innovation pour les entreprises opérant dans ce domaine.

3.7 Les applications mobiles dans la livraison :

Les applications mobiles dans le secteur de la livraison a révolutionné la façon dont les entreprises gèrent leurs opérations et interagissent avec leurs clients. Ces applications ont apporté un changement radical dans la manière dont les livraisons sont planifiées, exécutées et suivies, offrant une efficacité accrue, une transparence améliorée et une expérience utilisateur optimisée. Dans cette section, nous explorerons en profondeur l'impact des applications mobiles sur la livraison, en mettant en évidence leurs fonctionnalités clés, leurs avantages pour les entreprises et les consommateurs, ainsi que les tendances émergentes dans ce domaine.

1.3.1 Fonctionnalités des applications mobiles de livraison

Nous examinerons les fonctionnalités clés offertes par les applications mobiles de livraison, telles que la gestion des commandes, la planification des itinéraires, le suivi en temps réel des livraisons, la communication avec les clients, les paiements en ligne, etc.

1.3.2 Avantages pour les entreprises de livraison

Nous analyserons les avantages qu'apportent les applications mobiles aux entreprises de livraison, notamment en termes d'efficacité opérationnelle, de réduction des coûts, d'optimisation des ressources et d'amélioration de la satisfaction client.

1.3.3. Avantages pour les consommateurs

Nous explorerons les avantages que les applications mobiles de livraison offrent aux consommateurs, tels que la commodité, la transparence, le suivi en temps réel des livraisons, la personnalisation des commandes, etc.

1.3.4 Tendances émergentes dans le domaine de la livraison mobile

Nous identifierons les tendances émergentes dans le domaine de la livraison mobile, telles que l'utilisation de l'intelligence artificielle, de la réalité augmentée, des drones et des véhicules autonomes pour améliorer les processus de livraison.

En explorant ces aspects, nous serons en mesure de comprendre pleinement l'impact des applications mobiles sur la livraison et d'identifier les opportunités et les défis associés à leur intégration dans le secteur de la logistique et de la livraison.

3.8 Présentation des application existantes :

Il existe déjà beaucoup d'applications de ce type, nous allons en citer quelques-unes :

. FedEx :

L'application mobile FedEx® vous permet de suivre le statut de vos envois, de trouver des emplacements FedEx à proximité, de personnaliser et de programmer des envois, d'accéder à l'assistance clientèle et bien plus encore. Peu importe où vous êtes, l'application mobile FedEx vous donne accès aux fonctionnalités dont vous avez besoin pour suivre, gérer et sécuriser vos envois, tout en déplacement. [2]

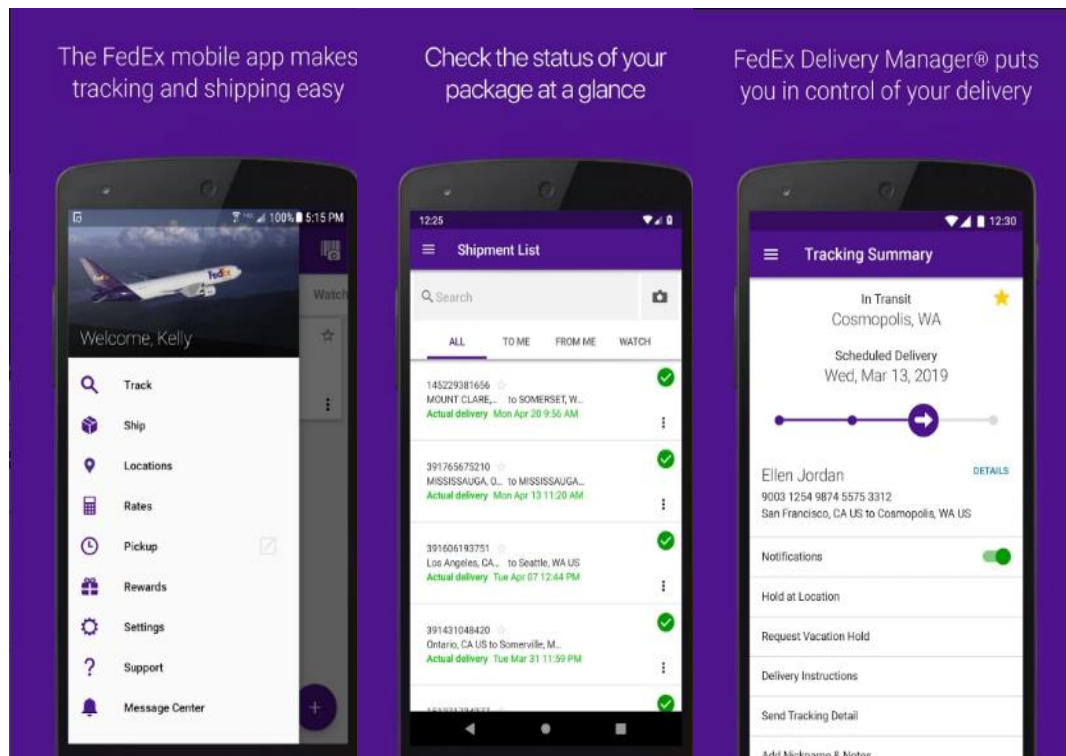


Figure 1.1 fonctionnalité de l'app FedEx .

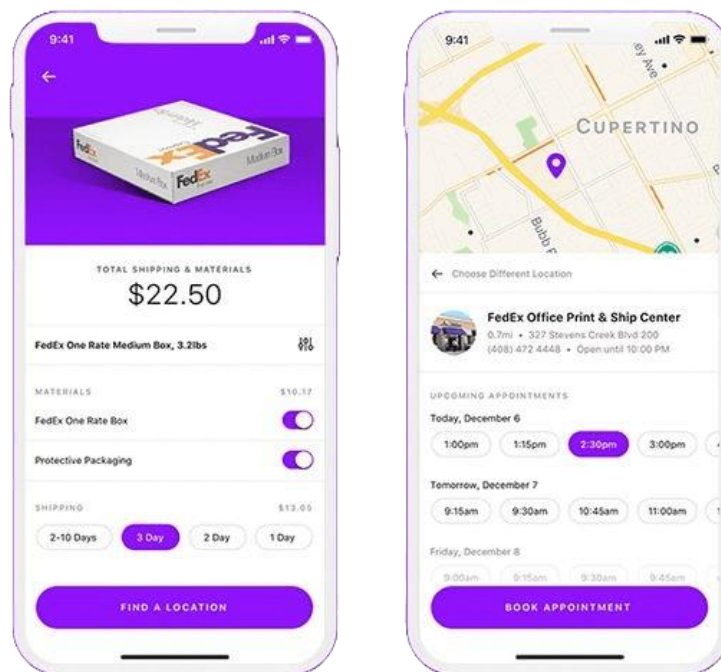


Figure 1.2 fonctionnalité de l'app FedEx.

. DHL Express :

L'application DHL Express est une plateforme mobile développée par DHL, une entreprise de logistique internationale de premier plan. Cette application permet aux utilisateurs d'accéder à une gamme de services de livraison, de suivi des colis et de gestion des expéditions directement depuis leur appareil mobile. Elle offre des fonctionnalités telles que la création et la planification d'expéditions, le suivi en temps réel des colis, la localisation des points de service DHL, la gestion des comptes clients, et bien plus encore. En résumé, l'application DHL Express offre aux utilisateurs une solution pratique et efficace pour gérer leurs envois et leurs livraisons avec DHL, en leur fournissant une expérience utilisateur transparente et pratique. [3]

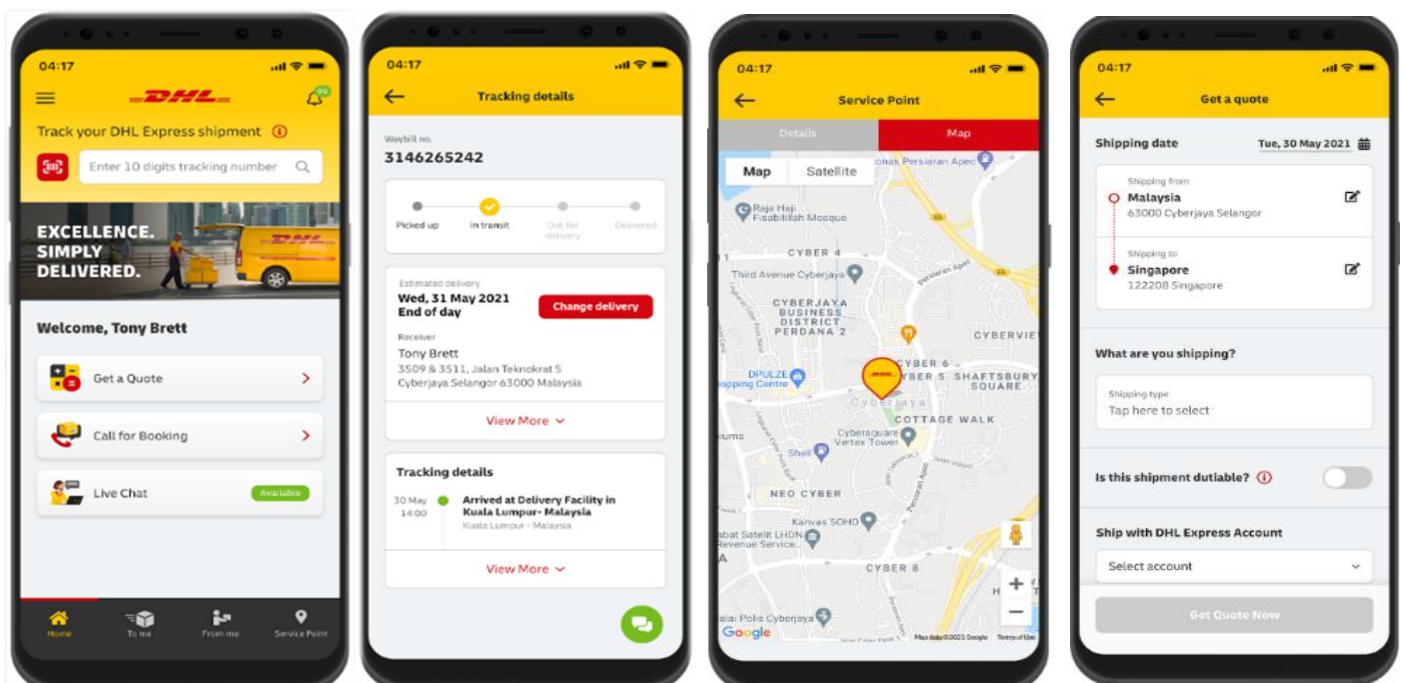


Figure 2 fonctionnalité de l'app DHL Express .

. Uber Delivery :

L'application Uber Delivery est une plateforme mobile développée par Uber Technologies Inc., qui permet aux entreprises de livraison de repas et de colis de gérer efficacement leurs opérations de livraison. Cette application offre aux utilisateurs une interface conviviale pour recevoir, traiter et suivre les commandes de livraison en temps réel. Elle permet également aux livreurs de visualiser les itinéraires, de communiquer avec les clients et de fournir des mises à jour sur l'état des livraisons. En résumé, l'application Uber Delivery offre aux entreprises de livraison une solution efficace pour optimiser leurs processus de livraison et offrir une expérience client fluide. [4]

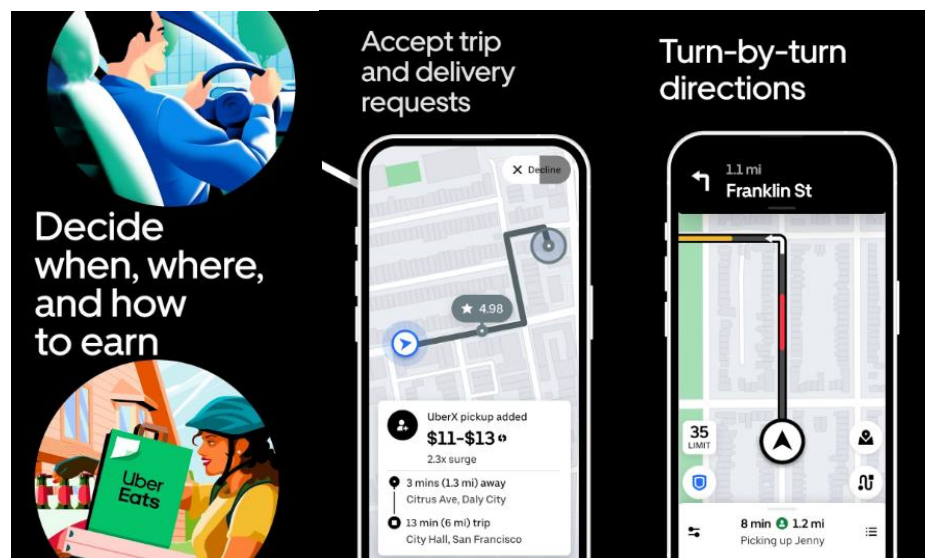


Figure 3 fonctionnalité de l'app Uber Delivery.

3.9 Fonctionnalités et avantages et inconvénients des application existantes :

Table 1 Fonctionnalités et avantages et inconvénients

Les Applications	Fonctionnalités	Avantage	Inconvénients
FedEx	<ol style="list-style-type: none"> 1. Suivi en temps réel des colis 2. Estimation des délais de livraison 3. Gestion des envois et des retours 4. Notifications de statut des livraisons 	<ol style="list-style-type: none"> 1. Suivi précis des colis. 2. Options de programmation flexibles. 3. Prise en charge des envois internationaux.. 	<ol style="list-style-type: none"> 1. Manque d'optimisation en temps réel pour les livraisons simultanées. 2. Absence de fonctionnalité de cartographie pour choisir le meilleur itinéraire. 3. Communication limitée entre le client et le livreur.
DHL Express	<ol style="list-style-type: none"> 1. Suivi précis des colis avec des mises à jour en temps réel. 2. Flexibilité dans la programmation des envois. 3. Expertise dans les envois internationaux. 	<ol style="list-style-type: none"> 1. Manque d'optimisation en temps réel pour les livraisons simultanées. 2. Absence de fonctionnalité de cartographie pour choisir le meilleur itinéraire. 3. Communication limitée entre le client et le livreur. 	<ol style="list-style-type: none"> 1. Gestion inefficace des livraisons simultanées dans un même espace. 2. Manque d'intégration de la cartographie pour optimiser les itinéraires. 3. Communication limitée entre le client et le livreur.
Uber Delivery	<ol style="list-style-type: none"> 1. Suivi en temps réel des livraisons. 2. Options de programmation des livraisons. 3. Intégration avec le système Uber pour la gestion des commandes. 	<ol style="list-style-type: none"> 1. Suivi précis des livraisons avec des mises à jour en temps réel. 2. Facilité d'utilisation grâce à l'intégration avec l'application Uber. 3. Flexibilité dans la programmation des livraisons. 	<ol style="list-style-type: none"> 1. Gestion inefficace des livraisons simultanées dans un même espace. 2. Manque d'intégration de la cartographie pour optimiser les itinéraires. 3. Communication limitée entre le client et le livreur.

Conclusion :

Dans ce chapitre, nous avons présenté les généralités, quelques applications pour gestion de livraison, cela nous a permis d'identifier les fonctionnalités principales que doit avoir une application pour gérer les livraisons pour minimiser le temps réel.

Introduction :

Dans ce chapitre, on présente et développe de manière élaborée les concepts relatifs au thème. Exploité, il s'agit non seulement de donner la définition générique des concepts, mais aussi et surtout d'indiquer de leur usage et leur mode d'application par rapport au thème.

2.1 le langage de modélisation «UML» :**2.1.1 Définition :**

UML (Unified Modeling Language) est un langage visuel standardisé utilisé pour décrire la structure et le comportement des systèmes logiciels. Il fournit une syntaxe et une sémantique riches pour représenter graphiquement les concepts et les interactions dans un logiciel. UML facilite la communication entre les développeurs, les architectes et les parties prenantes tout au long du processus de développement logiciel, réduisant ainsi les risques d'ambiguïté et d'erreur. Son utilisation permet de modéliser une grande variété de systèmes, des petites applications aux grandes infrastructures informatiques, aidant ainsi à concevoir des solutions logicielles robustes et efficaces. [5]

2.2 Diagramme de cas d'utilisation :**2.2.1 Définition :**

Les diagrammes de cas d'utilisation (DCU) sont une composante essentielle de la modélisation UML, offrant une vue globale du comportement fonctionnel d'un système logiciel. Ils représentent les différentes interactions entre les utilisateurs (ou acteurs) et le système lui-même. Il est important de noter que les acteurs peuvent être des utilisateurs humains ou des systèmes externes interagissant avec le système. Ainsi, les DCU offrent une perspective holistique des fonctionnalités du système en mettant l'accent sur les interactions entre les acteurs et le système plutôt que sur les détails internes de son fonctionnement. Ces diagrammes sont un outil puissant pour comprendre et communiquer les besoins et les fonctionnalités du système de manière claire et précise. [6]

2.2.2. Modélisation :**1. Cas d'utilisation : Gérer les Clients, les Livreurs, les Véhicules de Livreur, les Commandes et les Routes**

Acteur Principal : Gestionnaire de Livraison

Préconditions : Le gestionnaire ouvre une session dans le système.

1. Authentification (Include) :

- Le gestionnaire saisit ses identifiants (nom d'utilisateur et mot de passe).
- Le système vérifie les informations d'identification.
- Si l'authentification réussit, le flux continue vers les autres cas d'utilisation. Sinon, un message d'erreur est affiché.

2. Gérer les Clients (Extend) :

- Le gestionnaire peut ajouter, supprimer, modifier, consulter et rechercher des clients.

3. Gérer les Livreurs (Extend) :

- Le gestionnaire peut ajouter, supprimer, modifier, consulter et rechercher des livreurs.

4. Gérer les Véhicules de Livreur (Extend) :

- Le gestionnaire peut ajouter, supprimer, modifier, consulter et rechercher des véhicules de livreur.

5. Gérer les Commandes (Extend) :

- Le gestionnaire peut ajouter, supprimer, modifier, consulter et rechercher des commandes.

6. Gérer les Routes (Extend) :

- Le gestionnaire peut planifier et optimiser les routes de livraison.

Dans ce scénario, l'authentification est une étape essentielle (incluse) avant que le gestionnaire ne puisse accéder aux fonctionnalités spécifiques de gestion des clients, des livreurs, etc. Cette approche

garantit la sécurité et l'intégrité du système en s'assurant que seules les personnes autorisées peuvent effectuer des actions de gestion.

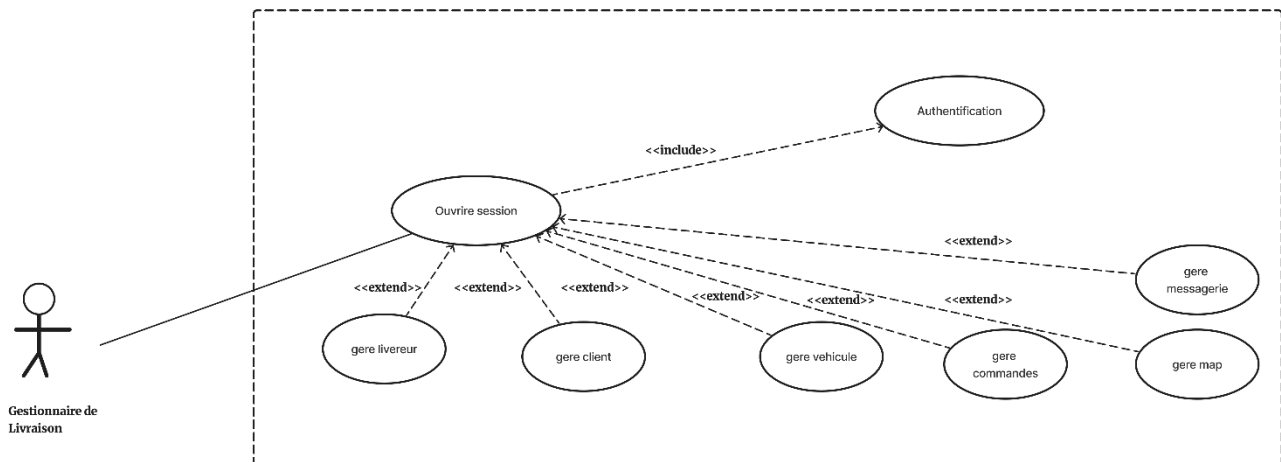


Figure 4 cas d'utilisation (en général)

2. Cas d'utilisation : Gérer les livreurs.

Acteur Principal : Gestionnaire de Livraison

Préconditions : Le gestionnaire est connecté au système.

1. Ajouter un Livreur :

Le gestionnaire saisit les informations du nouveau livreur (nom, numéro de téléphone, véhicule, etc.).

Le système enregistre les informations du livreur dans la base de données.

2. Supprimer un Livreur :

Le gestionnaire sélectionne le livreur à supprimer.

Le système confirme la suppression et retire le livreur de la base de données.

3. Modifier les Informations d'un Livreur :

Le gestionnaire recherche et sélectionne le livreur dont il souhaite modifier les informations.

Il modifie les détails nécessaires (numéro de téléphone, véhicule attribué, etc.).

Le système enregistre les modifications dans la base de données.

4. Consulter les Détails d'un Livreur :

Le gestionnaire recherche et sélectionne le livreur dont il souhaite consulter les informations.
Le système affiche les détails du livreur (nom, numéro de téléphone, véhicule attribué, etc.).

5. Rechercher un Livreur :

Le gestionnaire saisit les critères de recherche (nom, numéro de téléphone, etc.).
Le système affiche une liste de livreurs correspondant aux critères de recherche.

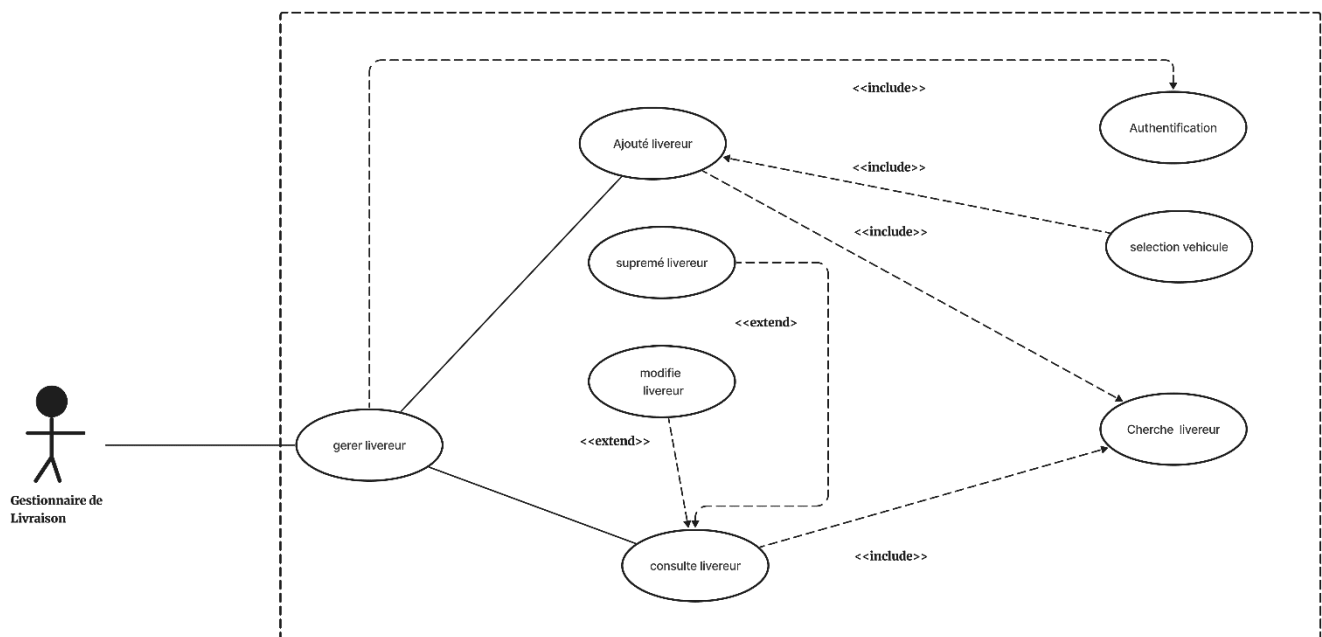


Figure 5 cas d'utilisation (cas de gestion livreur)

3. Cas d'utilisation : Gérer Client

Acteur Principal : Gestionnaire de Livraison

Préconditions : Le gestionnaire est connecté au système.

Ajouter un Client :

Le gestionnaire saisit les informations du nouveau client (nom, adresse, numéro de téléphone, etc.).
Le système enregistre les informations du client dans la base de données.

Supprimer un Client :

Le gestionnaire sélectionne le client à supprimer.

Le système confirme la suppression et retire le client de la base de données.

Modifier les Informations d'un Client :

Le gestionnaire recherche et sélectionne le client dont il souhaite modifier les informations.

Il modifie les détails nécessaires (adresse, numéro de téléphone, etc.).

Le système enregistre les modifications dans la base de données.

Consulter les Détails d'un Client :

Le gestionnaire recherche et sélectionne le client dont il souhaite consulter les informations.

Le système affiche les détails du client (nom, numéro de Téléphone , historique des commandes, etc.).

Rechercher un Client :

Le gestionnaire saisit les critères de recherche (nom, numéro de téléphone, etc.).

Le système affiche une liste de clients correspondant aux critères de recherche.

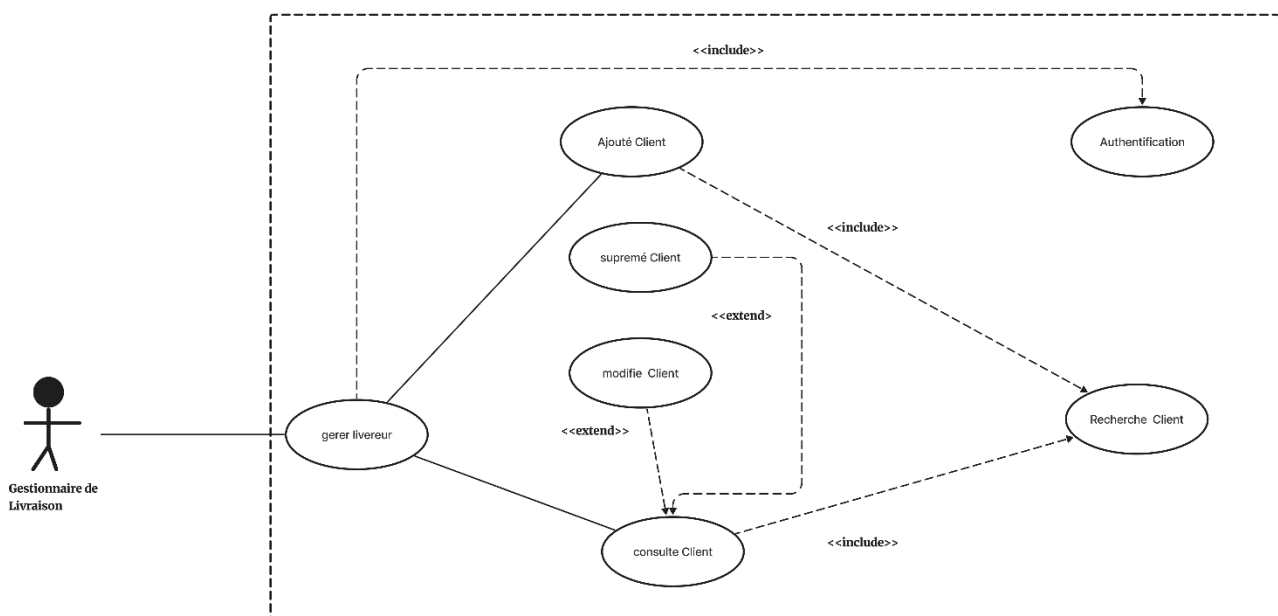


Figure 6 cas d'utilisation (cas de gestion client).

2.3 Diagramme de séquence :

2.3.1 Définition :

Un diagramme de séquence est une représentation visuelle utilisée dans le langage UML pour décrire les interactions entre les objets d'un système dans un ordre chronologique. Il illustre la séquence des messages échangés entre les objets au fil du temps, mettant en évidence les actions et les transitions qui se produisent lors de l'exécution d'un scénario spécifique. Les diagrammes de séquence sont particulièrement utiles pour modéliser le flux de contrôle dans les systèmes logiciels et pour comprendre les interactions entre les différents composants d'une application. [7]

2.3.2 Modélisation :

- **Inscription (comme client) :**

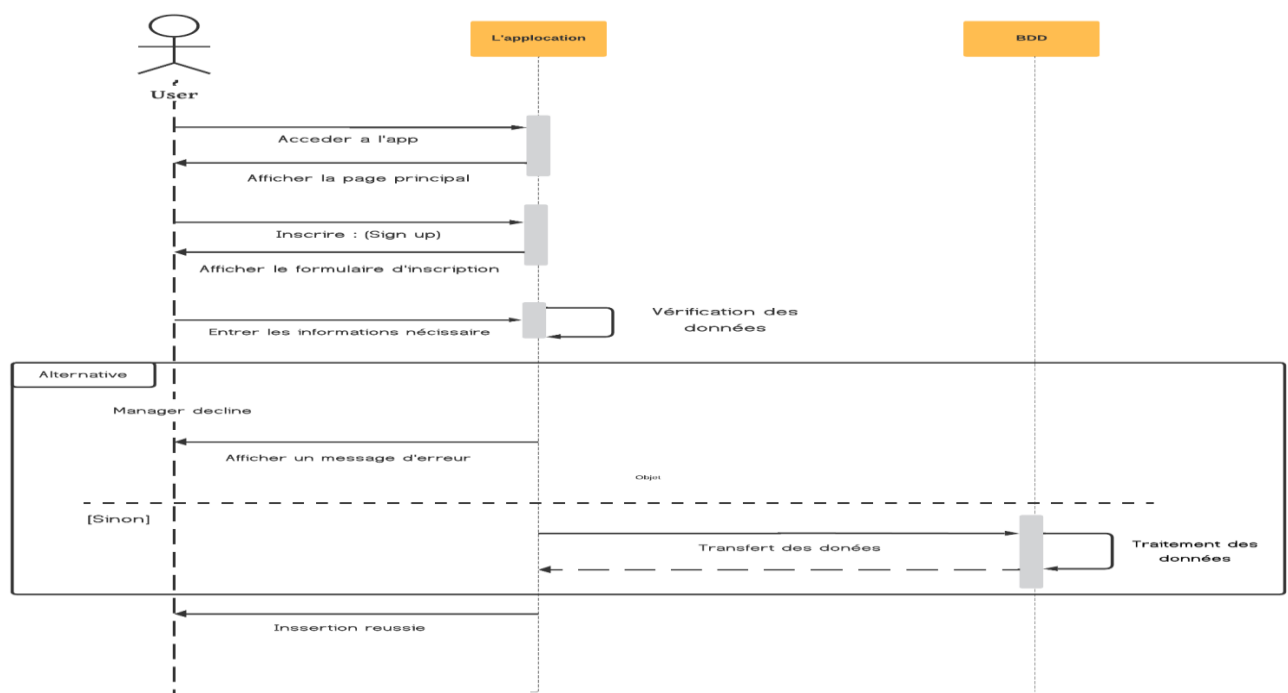


Figure 7 diagramme de séquence (cas de sign up Client).

■ Inscription (comme Livreur) :

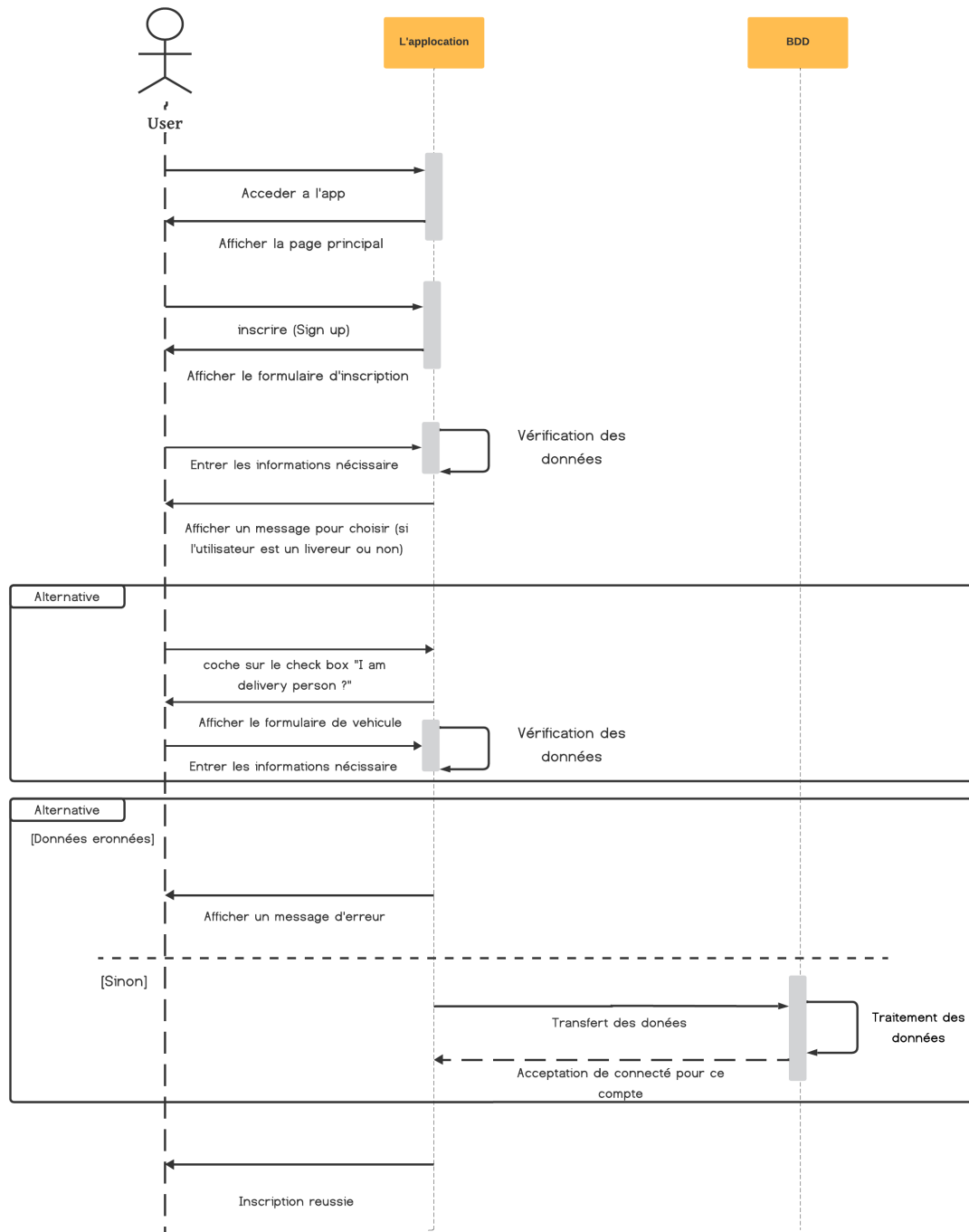
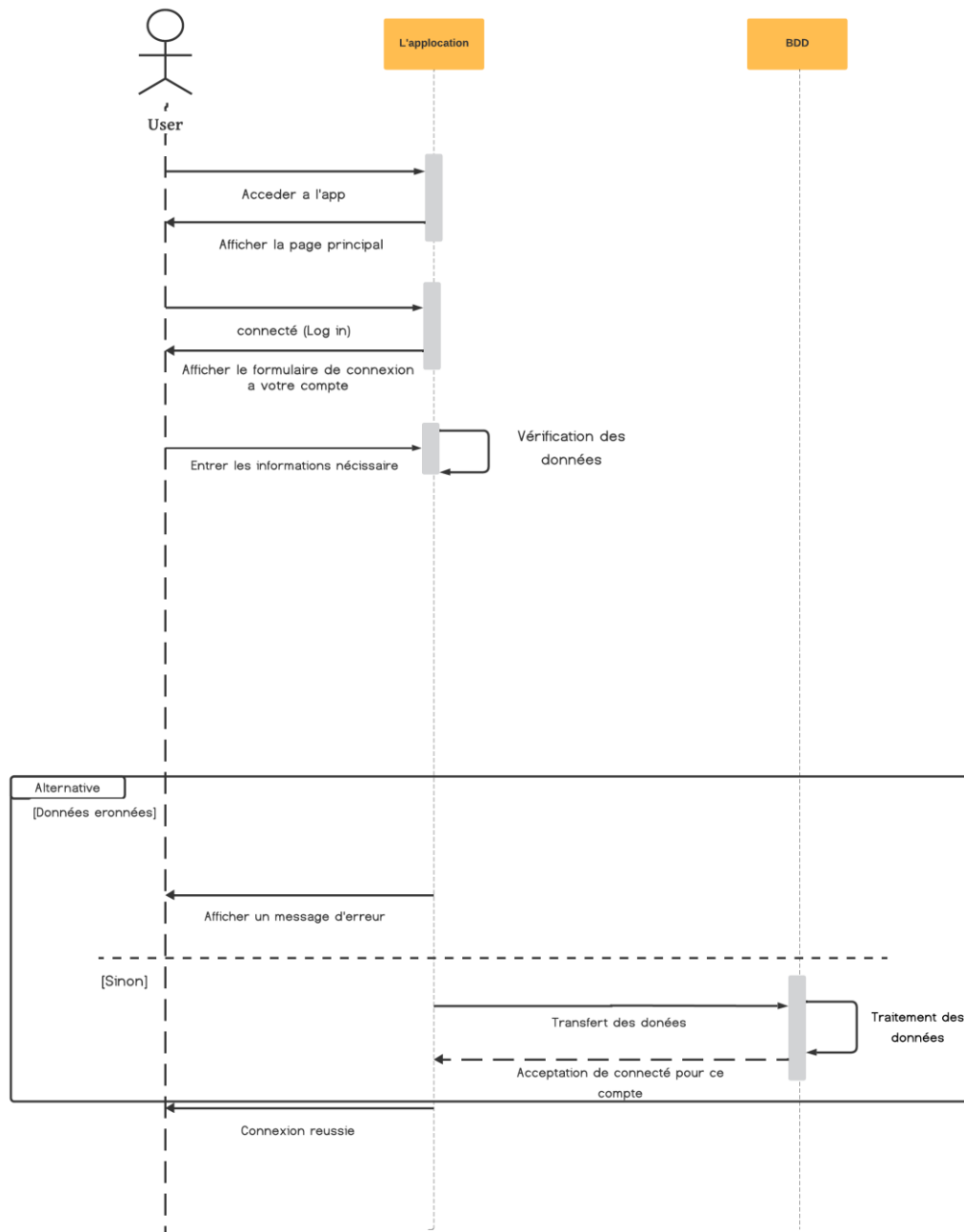
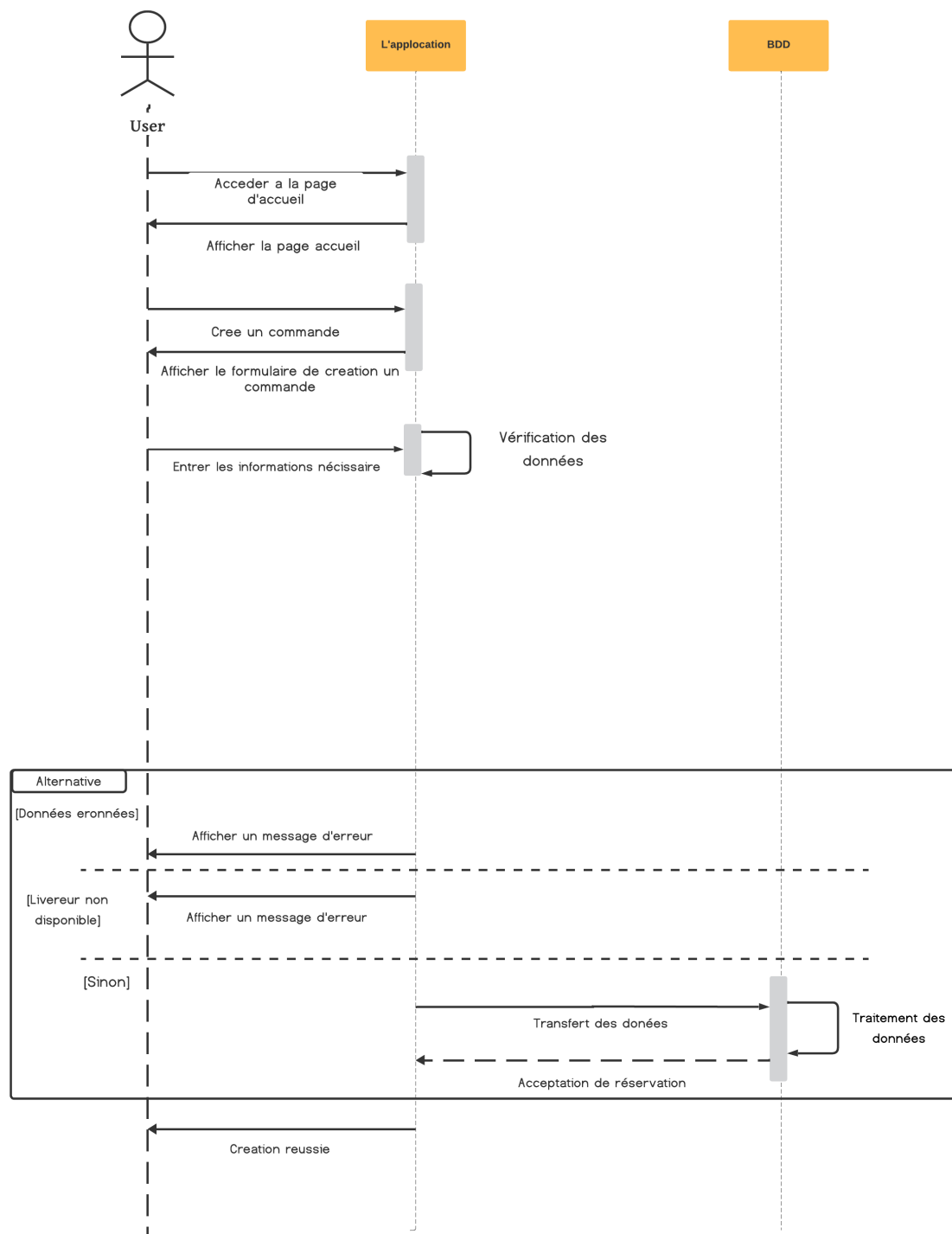


Figure 8 diagramme de séquence (cas de sign up Livreur).

■ Authentification

*Figure 9 diagramme de séquence (cas de log in).*

■ Création d'une commande :

*Figure 10 diagramme de séquence (cas de création une commande).*

■ Gestion Commande :

Modification commande :

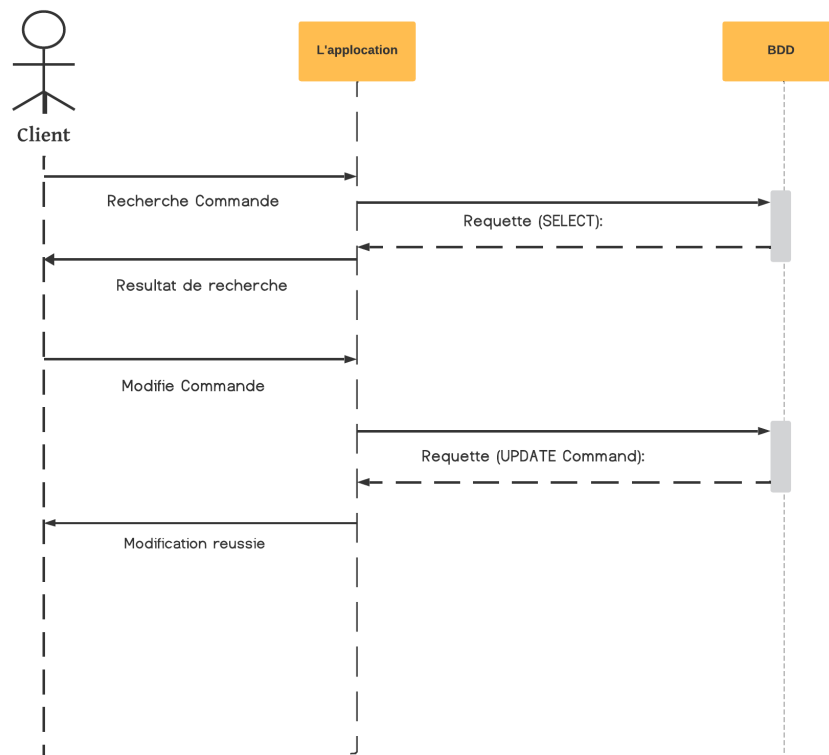


Figure 11 diagramme de séquence (cas de modification d'une commande).

Suppression commande :

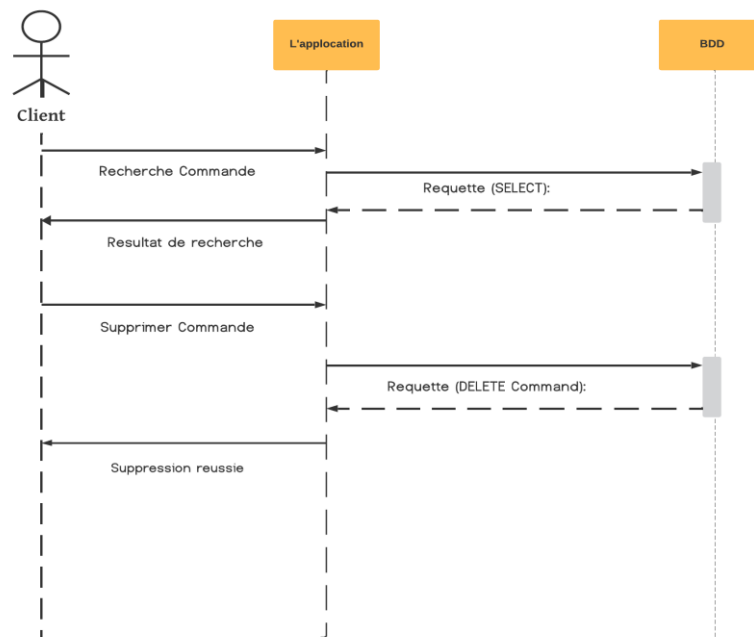


Figure 12 diagramme de séquence (cas de supprimer d'une commande).

2.3 Diagramme de classe :

2.3.1 Définition :

Un diagramme de classes est un type de diagramme utilisé en génie logiciel et en modélisation de données pour décrire la structure d'un système en représentant ses classes, leurs attributs, leurs méthodes et les relations entre elles. Il est souvent utilisé dans le cadre de la méthodologie UML (Unified Modeling Language) pour la conception et la documentation des architectures de systèmes logiciels. [8]

2.3.1 Modélisation :

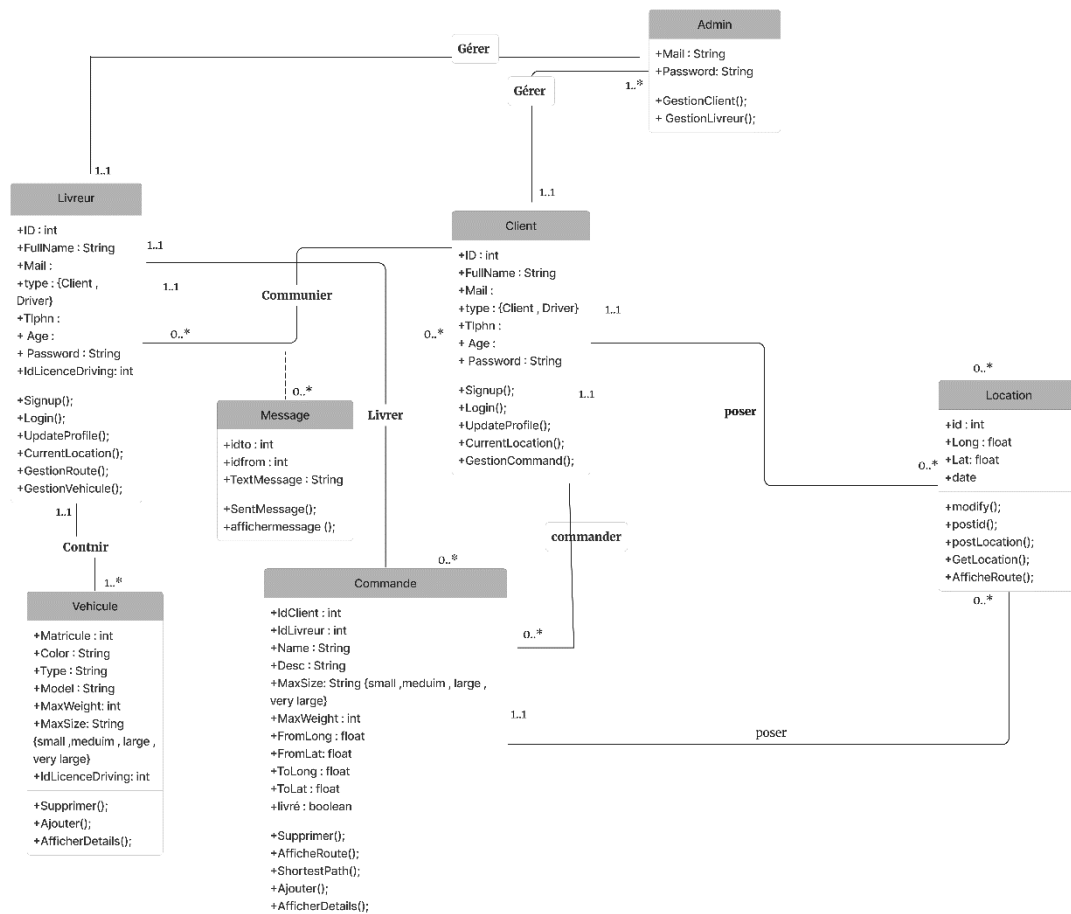


Figure 13 diagramme de classe (gestion de livraison).

Conclusion :

Après nos recherches et notre étude détaillée, nous avons pu évaluer un système de gestion de livraison en utilisant des outils de modélisation simples. Cette évaluation nous a permis de mieux comprendre les étapes de création d'un système efficace qui aide et facilite le travail des livreurs tout en optimisant les trajets de livraison. Les résultats de cette évaluation seront présentés dans le chapitre suivant.

Introduction :

Dans ce chapitre, nous passons de l'étude conceptuelle à la phase de réalisation et d'implémentation, où nous allons présenter l'état final de notre projet (application mobile). Ce chapitre sera partitionné en deux parties : la première présentera l'environnement de travail (les différents outils de développement utilisés dans notre réalisation), tandis que la deuxième partie détaillera les multiples fonctionnalités qu'offre le système à travers les différentes interfaces de l'application mobile.

3.1. Environnement du travail :

3.1.1. Outil de développement :



Android Studio est un environnement de développement intégré (IDE) conçu spécifiquement pour le développement d'applications Android. Il offre une suite d'outils puissants pour la conception, le débogage et le déploiement d'applications sur la plateforme Android. Android Studio inclut des fonctionnalités telles que l'émulateur Android, le support pour Kotlin, le langage de programmation officiel d'Android, ainsi que des outils de performance avancés. Il est largement utilisé par les développeurs Android pour créer des applications robustes et performantes. [9]



Flutter est un framework open-source développé par Google pour la création d'applications nativement compilées pour mobile, web et bureau à partir d'une seule base de code. Il utilise un moteur de rendu graphique performant pour afficher des interfaces utilisateur élégantes et réactives. Flutter permet un développement rapide grâce à son hot reload, qui permet de voir instantanément les modifications apportées au code. [10]



SQLite est un moteur de base de données relationnelle embarqué largement utilisé pour le stockage de données locales dans les applications logicielles. Il est léger, rapide et nécessite peu de configuration, ce qui en fait un choix populaire pour les applications mobiles, les navigateurs web et d'autres logiciels nécessitant une gestion de données locale. [11]



Une **API (Interface de Programmation d'Application)** est un ensemble de règles, de protocoles et de définitions de données qui permettent à différentes applications logicielles de communiquer entre elles. Elle définit les méthodes et les formats de données que les applications peuvent utiliser pour demander et échanger des informations. Les API sont couramment utilisées pour permettre l'intégration entre différents systèmes, services ou plateformes, facilitant ainsi le développement d'applications complexes et la collaboration entre développeurs. [12]



OpenStreetMap (OSM) est un projet collaboratif de cartographie en ligne qui vise à créer une carte du monde libre et éditée par tous. Il permet à quiconque de visualiser, modifier et utiliser les données cartographiques, contribuant ainsi à créer une carte détaillée et précise de la planète. Les utilisateurs peuvent ajouter des routes, des bâtiments, des points d'intérêt et d'autres informations géographiques, ce qui en fait une ressource précieuse pour de nombreux projets, notamment la navigation, l'urbanisme, l'humanitaire et la planification de voyages. [13]

3.1.2. Les Langages utilisés



Dart est un langage de programmation optimisé pour les applications client, également développé par Google. Il est utilisé comme langage principal pour le développement avec Flutter. Dart est un langage orienté objet et possède des caractéristiques modernes telles que le typage statique optionnel, une syntaxe simple et des outils robustes, facilitant ainsi le développement de performances élevées et d'applications maintenables. [10]



HTML (HyperText Markup Language) est le langage standard de balisage utilisé pour créer des pages web. Il permet de structurer le contenu sur le web en utilisant des éléments comme des titres, des paragraphes, des listes, des liens, des images, et plus encore. [14]



CSS (Cascading Style Sheets) est un langage de style utilisé pour décrire la présentation d'un document écrit en HTML ou XML. CSS permet de contrôler la mise en page, les couleurs, les polices et d'autres aspects visuels des pages web, séparant ainsi le contenu de sa présentation. [14]



JavaScript est un langage de programmation orienté objet principalement utilisé pour rendre les pages web interactives. Il est capable de modifier le contenu et le style des pages après leur chargement, gérer les événements, valider les formulaires, créer des animations, et bien plus encore. [15]



Django est un framework web open-source écrit en Python qui facilite le développement rapide de sites web complexes et sécurisés. Il suit le principe du modèle-vue-contrôleur (MVC) et offre une architecture modulaire, des fonctionnalités de sécurité intégrées, un ORM (Object-Relational Mapping) pour interagir avec la base de données, et un système de gestion d'administration automatique. Django favorise également la réutilisation du code et le respect des meilleures pratiques de développement web. [16]



Python est un langage de programmation interprété, polyvalent et convivial. Connu pour sa syntaxe claire et concise, il favorise la lisibilité du code et la productivité des développeurs. Python est utilisé dans une large gamme de domaines, y compris le développement web, la science des données, l'intelligence artificielle, l'automatisation des tâches, et bien d'autres. Sa grande communauté de développeurs et sa vaste bibliothèque standard en font l'un des langages les plus populaires et les plus appréciés au monde. [17]



JSON (JavaScript Object Notation) est un format d'échange de données léger, facile à lire et à écrire pour les humains, et facile à analyser et à générer pour les machines. Basé sur un sous-ensemble du langage de programmation JavaScript, il est largement utilisé pour la transmission de données entre un serveur et une application web en tant qu'alternative à XML. Les données JSON sont représentées sous forme de paires clé-valeur et prennent en charge les tableaux et les objets imbriqués. Les fichiers JSON sont couramment utilisés pour les fichiers de configuration, le stockage de données et la communication entre différents systèmes logiciels en raison de leur simplicité et de leur facilité d'utilisation. [18]

3.2. Communication entre Frontend et backend :

Le fonctionnement du backend implique la mise en place d'un serveur web utilisant Django, un framework web Python, pour traiter les requêtes entrantes du frontend, développé en Flutter et Dart. Lorsqu'une requête est émise depuis le frontend, elle est envoyée au backend via des appels d'API RESTful. Le backend utilise ensuite ces requêtes pour interagir avec la base de données, récupérer ou ajouter des données selon la méthode spécifiée dans la requête (GET pour la récupération, POST pour l'ajout, etc.). Une fois que le backend a récupéré ou ajouté les données, il les renvoie au frontend sous forme de fichiers JSON, qui sont ensuite interprétés et affichés dans l'interface utilisateur du frontend. Cette communication entre le frontend et le backend permet la synchronisation des données et le bon fonctionnement de l'application dans son ensemble.

Table account :

- **Attributs** : username, age, email, phone_number, id_number, id_driver_license, user_type, password, id
- **API** :
 1. register : Permet de créer un nouveau compte utilisateur.
 2. login : Permet de se connecter en utilisant les identifiants de l'utilisateur. (Un token aléatoire est créé pour renforcer le système de sécurité et donner à chaque utilisateur sa propre sécurité)
 3. update : Permet de mettre à jour les informations d'un utilisateur existant. (avec token)
 1. remove : Permet de supprimer un utilisateur. (avec token)
 4. list users : Permet de lister tous les utilisateurs.
 5. get user : Permet de récupérer les détails d'un utilisateur spécifique.

Table vehicule :

- **Attributs** : matricule, color, type, maxweight, maxsize, iddriver, model
- **API** :
 1. vehicule list : Permet de lister tous les véhicules.
 2. vehicule details : Permet de récupérer les détails d'un véhicule spécifique.
 3. vehicule details with id license : Permet de récupérer les détails d'un véhicule à partir d'identifiant du conducteur.
 4. create : Permet de créer un nouveau véhicule.
 5. update with matricule : Permet de mettre à jour un véhicule en utilisant le matricule.
 6. update with id license : Permet de mettre à jour un véhicule en utilisant l'identifiant du conducteur.
 7. delete with matricule : Permet de supprimer un véhicule en utilisant le matricule.
 8. delete with id license : Permet de supprimer un véhicule en utilisant l'identifiant du conducteur.

Table location :

- **Attributs** : id, lat, lon, geopoint, time
- **API** :
 1. create : Permet de créer une nouvelle localisation.
 2. list locations : Permet de lister toutes les localisations.
 3. get user locations : Permet de récupérer les localisations d'un utilisateur spécifique.
 4. update location : Permet de mettre à jour une localisation existante.

Table commande :

- **Attributs** : id, iddriver, idclient, description, fromlat, fromlon, tolat, tolon, size, weight, completed
- **API** :
 1. list commande : Permet de lister toutes les commandes.
 2. create commande : Permet de créer une nouvelle commande.
 3. get commande : Permet de récupérer les détails d'une commande spécifique.
 4. update commande : Permet de mettre à jour une commande existante.
 5. delete commande : Permet de supprimer une commande.
 6. complete commande : Permet de marquer une commande comme complétée.
 7. get commande : Permet de récupérer les détails d'une commande spécifique.

Table message :

- **Attributs** : id, iddriver, idclient, textmessage, time
- **API** :
 1. get message : Permet de récupérer les messages échangés entre le conducteur et le client.
 2. create message : Permet de créer un nouveau message.

Exemples

On a user1 avec :

- username: "reda_elaidate"
- age: 28
- email: "reda@gmail.com"
- phone_number: "0525148342";
- id_number: "23419051921";
- id_driver_license: "12415105383";
- user_type: "Driver";

- 1) Pour get vehicule de reda on a utilisé API (vehicule details with id license) avec id_driver_license et vehicule comme output .
- 2) Pour get messages avec user2 on a utilisé API (get message) avec id user1 et user2 comme input et messages comme output filtered avec temp
- 3) location driver avec (update location) qui exécuté après 2 min location de driver on a toujours garde.

3.3. Présentation de l'application mobile :

3.3.1. La page d'accueil :

Nous avons programmé ce page pour donne l'utilisateur le choix pour soit se connecte ou s'inscrit.
Nous avons choisi un design simple et efficace avec des couleurs élégantes, comme le montre la figure ci-dessous :

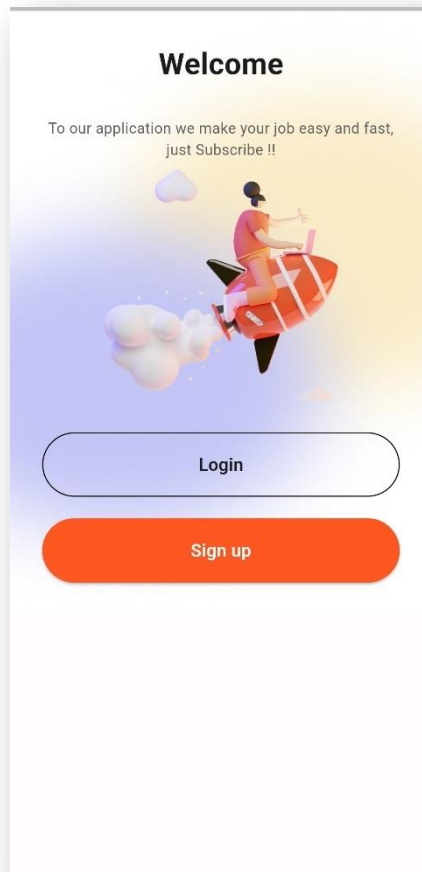


Figure 14 page d'accueil de l'application mobile

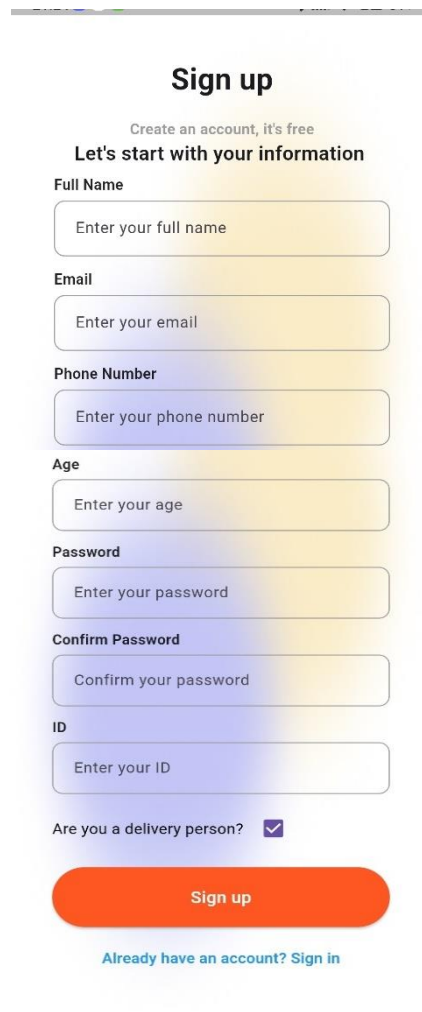
3.3.2. La page d'inscription :

Nous avons programmé cette page pour aider l'utilisateur à s'inscrire sur cette application , il doit remplir toutes les entrées du formulaire comme son nom, son e-mail, son numéro de téléphone, son âge, son mot de passe et confirmer le mot de passe .

Figure 15 page d'inscription de l'application mobile (cas d'inscrit comme client)

Remarque : Si l'utilisateur ne coché pas la case du "Are you a delivery person?" il sera accède à la page principal comme un client sinon il sera accède a la page suivant comme un livreur mais d'abord il faut remplir l'entrée de identifier.

Après de remplissage le formulaire s'il est livreur



The image shows a mobile application sign-up screen for a delivery person. The form is titled 'Sign up' and includes a sub-header 'Let's start with your information'. Below this, there are several input fields: 'Full Name', 'Email', 'Phone Number', 'Age', 'Password', 'Confirm Password', and 'ID'. Each field has a placeholder text indicating what to enter. At the bottom, there is a checkbox labeled 'Are you a delivery person?' which is checked. Below the checkbox is a large orange 'Sign up' button. At the very bottom, there is a link that says 'Already have an account? Sign in'.

Sign up

Create an account, it's free

Let's start with your information

Full Name

Enter your full name

Email

Enter your email

Phone Number

Enter your phone number

Age

Enter your age

Password

Enter your password

Confirm Password

Confirm your password

ID

Enter your ID

Are you a delivery person? ☒

Sign up

[Already have an account? Sign in](#)

Figure 16 page d'inscription de l'application mobile (cas livreur)

Il va à la page suivant spéciale pour les information de véhicule de livreur qui il faut remplir toutes les entrées de formulaire et télécharger des photos du véhicule et la plaque d'immatriculation et document d'enregistrement

The image shows a mobile application registration screen for a driver, specifically step 2 of entering vehicle information. The screen has a light yellow background with a blue gradient at the bottom. At the top left, there is a back arrow icon. Below it, the text "let's know your vehicle information" is displayed. The form consists of several input fields, each with a label and a placeholder text: "Matricule" (placeholder: "Enter your matricule of your vehicle"), "Color" (placeholder: "Enter color of your vehicle"), "Type" (placeholder: "Enter your type of your vehicule"), "Model" (placeholder: "Enter your model of your vehicle"), "Max weight" (placeholder: "Enter max weight of your vehicule"), "Max size" (placeholder: "Enter max size of your vehicule"), and "ID licence driving" (placeholder: "Enter your id licence of driving"). Below these fields are three upload buttons: "Upload Vehicle Picture", "Upload License Plate Picture", and "Upload Registration Document Picture". At the bottom, there is a large orange "Sign up" button.

Figure 17 page d'inscription de l'application mobile (cas d'inscrit comme un livreur étape 2 entre les information de véhicule de livreur)

Notre application a une page de profil de l'utilisateur cette page donne a l'utilisateur le droit de voir son profil et il a droit de modifier son profil par exemple son nom, son e-mail, et il a bouton de notification qui donne à l'utilisateur la possibilité pour voir toute la mise à jour comme nouveau information et les bons de transaction avec les livreurs.

Figure 18 page de création d'une commande de l'application mobile

Donc ici pour le client crée une commande il faut d'abord remplis ces trois attributs :

- ❖ Size (il faut entrer un ces quatre propositions 'Extra large' ou 'large' ou 'Medium' ou 'Small').
- ❖ Weight (il faut entrer le poids de sa commande mais le numéro il doit être naturelle sans virgule).
- ❖ Description (il faut entrer description sur son commande)
- ❖ Location (il faut click sur ce buton pour il accède à autre page pour sélectionné sa position actuelle et sa position future après il faut clique sur buton 'done' pour sauvegarde les deux positions et retourne a cette page pour clique sur buton 'create' pour crée la commande exactement postule la commande).

Nous avons créé cette page afin de lui faciliter le travail en utilisant une carte et en lui montrant l'itinéraire pour livrer toutes les commandes. Nous avons ajouté deux options pour lui : soit laisser l'itinéraire tel que, soit lui montrer un raccourci afin de l'afficher. Pour lui donner le raccourci, nous avons utilisé l'algorithme de Dijkstra, qui permet d'afficher le meilleur itinéraire.

Nous avons ajouté quelques boutons à cette page afin de rechercher, masquer des points, masquer le chemin.

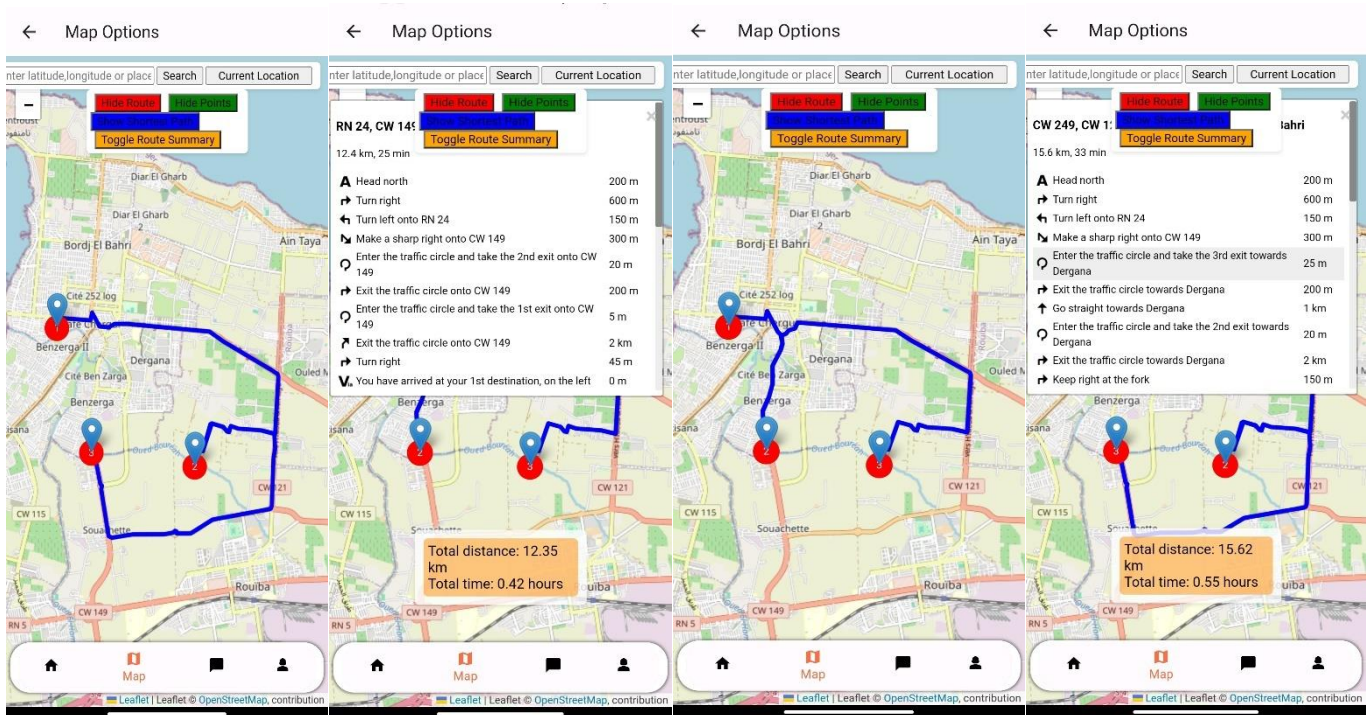


Figure 19 page de gestion des chemins de l'application mobile

Remarque :

Cet algorithme ça marche sur nombre des points limité si les points dépassants 50 points on va perde de temps donc on va utiliser une autre méthode :

Voyageurs de commerces résolut par : - l'optimisation par colonie de fourmis (ACO) et l'optimisation par essaim de particules (PSO).

Nous avons créé ce page pour affiche les commandes disponibles et après on a donner deux propositions soit accepter la commande soit rejeter.

S'il est accepté automatiquement emplacement de la commande va afficher sur la carte.

Sinon elle n'apparaîtra plus.

Et on a liste d'historique des transactions client avec livreur.

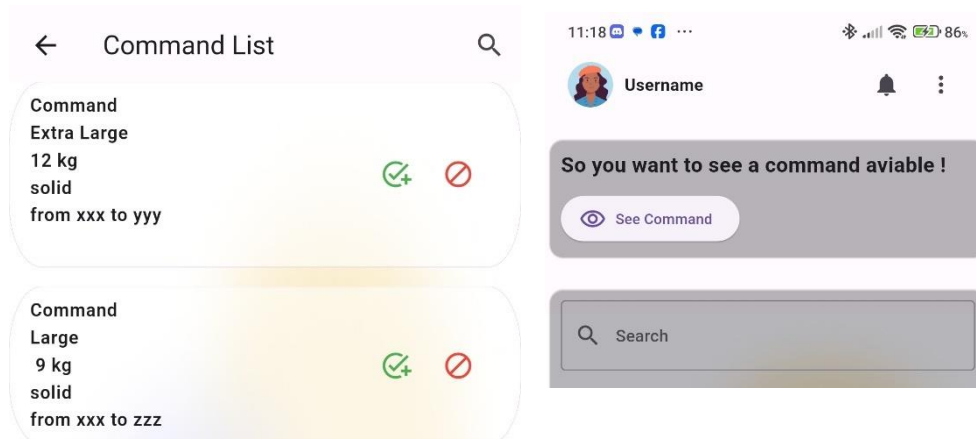


Figure 20 page de gestion commande de l'application mobile

Nous avons créé aussi cette page pour afficher le profil de l'utilisateur.

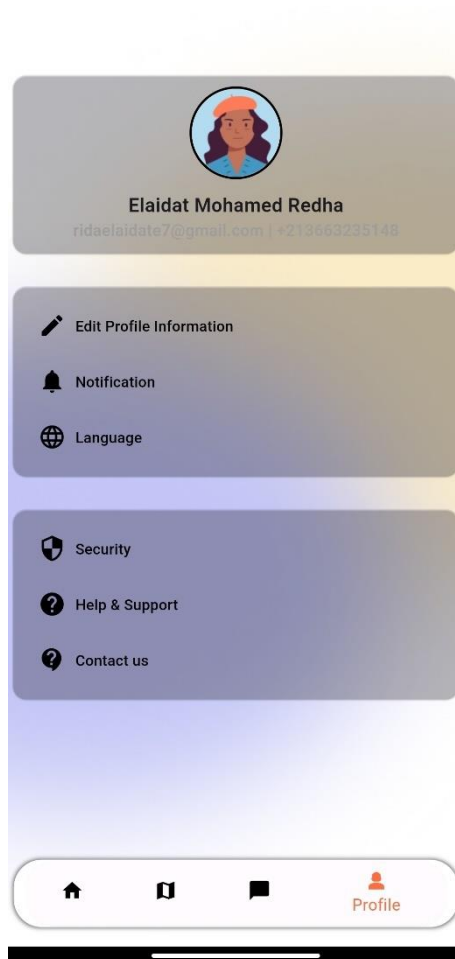


Figure 21 page de profil de utilisateur de l'application mobile

Conclusion :

Dans ce chapitre, nous avons présenté toutes les fonctionnalités de notre system par des captures et des explications détaillés.

Conclusion Générale :

Cette expérience, nous a convaincu de l'importance de la modélisation dans le processus de réalisation des projets informatiques. En plus, ce projet nous a donné l'occasion d'appliquer et d'améliorer nos connaissances théoriques dans le langage UML et notamment dans la modélisation des applications mobile.

Notre application assure la gestion des livreurs et clients et véhicule et chemins et commandes... afin de garantir le bon fonctionnement pour entreprise de livraison.

Autrement dit ce travail peut être considéré comme une première étape pour une gestion complète d'application.

En outre, notre application a besoin de gérer d'autres fonctionnalités et améliorations comme les notifications pour les mises à jours et un réel serveur pour garantir la sécurité des données et bon fonctionnement de transfert des données et donne la possibilité pour l'administrateur peut modifier sur la base de données à distance.

Finalement, ce projet a été très enrichissant que ça soit au point de vue technique qu'humain. Néanmoins, Nous aimerons bien que ce travail sera apprécié par les responsables et en suite connaît son achèvement.

Bibliographie :

[1] "Delivery." Investopedia.

Consulté le 15 Avril 2024.

Disponible sur : <https://www.investopedia.com/terms/d/delivery.asp>

[2] "Mobile Application FedEx." FedEx.

Consulté le 11 Avril 2024.

Disponible sur : <https://www.fedex.com/en-us/tracking/mobile-apps.html>

[3] DHL Express.

Consulté le 12 Avril 2024.

Disponible sur : www.logistics.dhl/fr-fr/home/our-divisions/express.html

[4] Uber Technologies Inc.

Consulté le 13 Avril.

Disponible sur : <https://www.uber.com/us/en/business/delivery/>

[5] "Lucidchart - Unified Modeling Language (UML)."

Lucid Software Inc. Consulté le 16 Avril 2024.

Disponible sur : <https://www.lucidchart.com/pages/uml>

Bibliographie :

[6] Lucidchart - Unified Modeling Language (UML)."

Lucid Software Inc.

Consulté le 3 Mai 2024

Disponible sur : <https://www.lucidchart.com/pages/uml>

[7] "Lucidchart - Diagramme de séquence."

Lucid Software Inc. Consulté le 7 Mai 2024.

Disponible sur : <https://www.lucidchart.com/pages/fr/uml-diagramme-de-sequence>

[8] "Diagramme de classes (UML)."

Wikipedia. Consulté le 8 Mai 2024 . Disponible sur :

[https://fr.wikipedia.org/wiki/Diagramme_de_classes_\(UML\)](https://fr.wikipedia.org/wiki/Diagramme_de_classes_(UML))

[9] Murphy, Mark L., et al. Android Programming: The Big Nerd Ranch Guide (4th Edition). Big Nerd Ranch Guides, 2019.

[10] "Flutter Documentation."

Consulté le 30 Mars 2024.

Disponible sur : <https://flutter.dev/docs>

[11] "SQLite Documentation."

Consulté le 26 Mars 2024.

Disponible sur : <https://www.sqlite.org/docs.html>

Bibliographie :

[12] "Mozilla Developer Network (MDN) - Glossaire API."

Consulté le 18 Avril 2024.

Disponible sur : <https://developer.mozilla.org/fr/docs/Glossary/API>

[13] "OpenStreetMap - À propos."

Consulté le 22 Avril 2024.

Disponible sur : <https://www.openstreetmap.org/about>

[14] Duckett, Jon. **HTML and CSS: Design and Build Websites**. John Wiley & Sons, 2011.

[15] Flanagan, David. **JavaScript: The Definitive Guide**. O'Reilly Media, 2020.

[16] "Django - Overview."

Consulté le [date à vérifier].

Disponible sur : <https://www.djangoproject.com/start/overview/>

[17] "Python - Blurb."

Consulté le 6 Avril 2024.

Disponible sur : <https://www.python.org/doc/essays/blurb/>

[18] "JSON."

Consulté le 5 Mai 2024.

Disponible sur : <https://www.json.org/json-en.html>

Diagramme de cas d'utilisation :

1. Cas d'utilisation : Gérer les Commandes

Acteur Principal : Gestionnaire de Livraison

Préconditions : Le gestionnaire est connecté au système.

Ajouter une Commande :

Le gestionnaire saisit les détails de la nouvelle commande (client, produits, adresse de livraison, etc.).

Le système enregistre la commande dans la base de données.

Supprimer une Commande :

Le gestionnaire sélectionne la commande à supprimer.

Le système confirme la suppression et retire la commande de la base de données.

Modifier les Informations d'une Commande :

Le gestionnaire recherche et sélectionne la commande dont il souhaite modifier les informations.

Il modifie les détails nécessaires (produits, adresse de livraison, etc.).

Le système enregistre les modifications dans la base de données.

Consulter les Détails d'une Commande :

Le gestionnaire recherche et sélectionne la commande dont il souhaite consulter les informations.

Le système affiche les détails de la commande (client, produits, statut de livraison, etc.).

Rechercher une Commande :

Le gestionnaire saisit les critères de recherche (client, statut de livraison, etc.).

Le système affiche une liste de commandes correspondant aux critères de recherche.

Annexe :

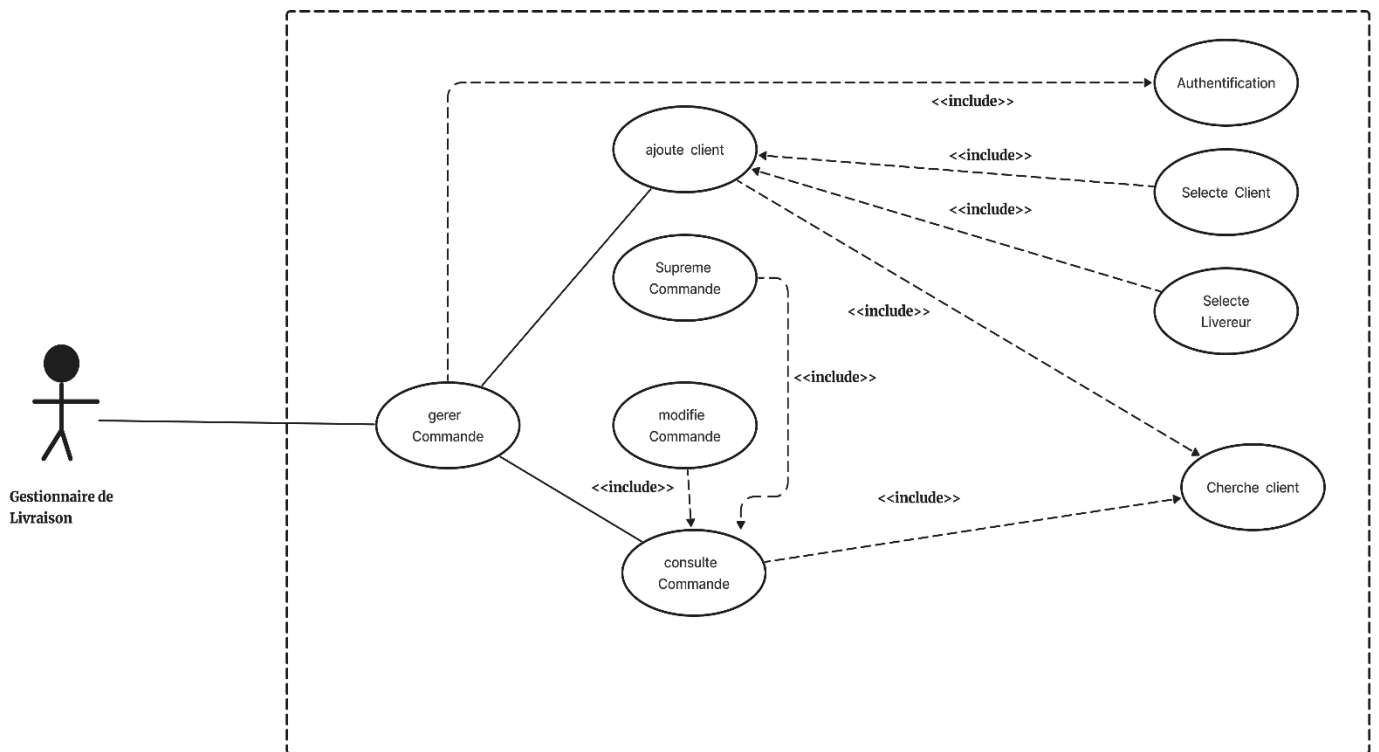


Figure 1 cas d'utilisation (cas gestion Commander).

2. Cas d'utilisation : Gérer les Véhicules des Livreurs

Acteur Principal : Gestionnaire de Livraison

Préconditions : Le gestionnaire est connecté au système.

Ajouter un Véhicule de Livreur :

Le gestionnaire saisit les informations du nouveau véhicule (modèle, immatriculation, capacité de charge, etc.).

Le système enregistre les informations du véhicule dans la base de données.

Annexe :

Supprimer un Véhicule de Livreur :

Le gestionnaire sélectionne le véhicule à supprimer.

Le système confirme la suppression et retire le véhicule de la base de données.

Modifier les Informations d'un Véhicule de Livreur :

Le gestionnaire recherche et sélectionne le véhicule dont il souhaite modifier les informations.

Il modifie les détails nécessaires (immatriculation, capacité de charge, etc.).

Le système enregistre les modifications dans la base de données.

Consulter les Détails d'un Véhicule de Livreur :

Le gestionnaire recherche et sélectionne le véhicule dont il souhaite consulter les informations.

Le système affiche les détails du véhicule (modèle, immatriculation, capacité de charge, etc.).

Rechercher un Véhicule de Livreur :

Le gestionnaire saisit les critères de recherche (modèle, immatriculation, etc.).

Le système affiche une liste de véhicules correspondant aux critères de recherche.

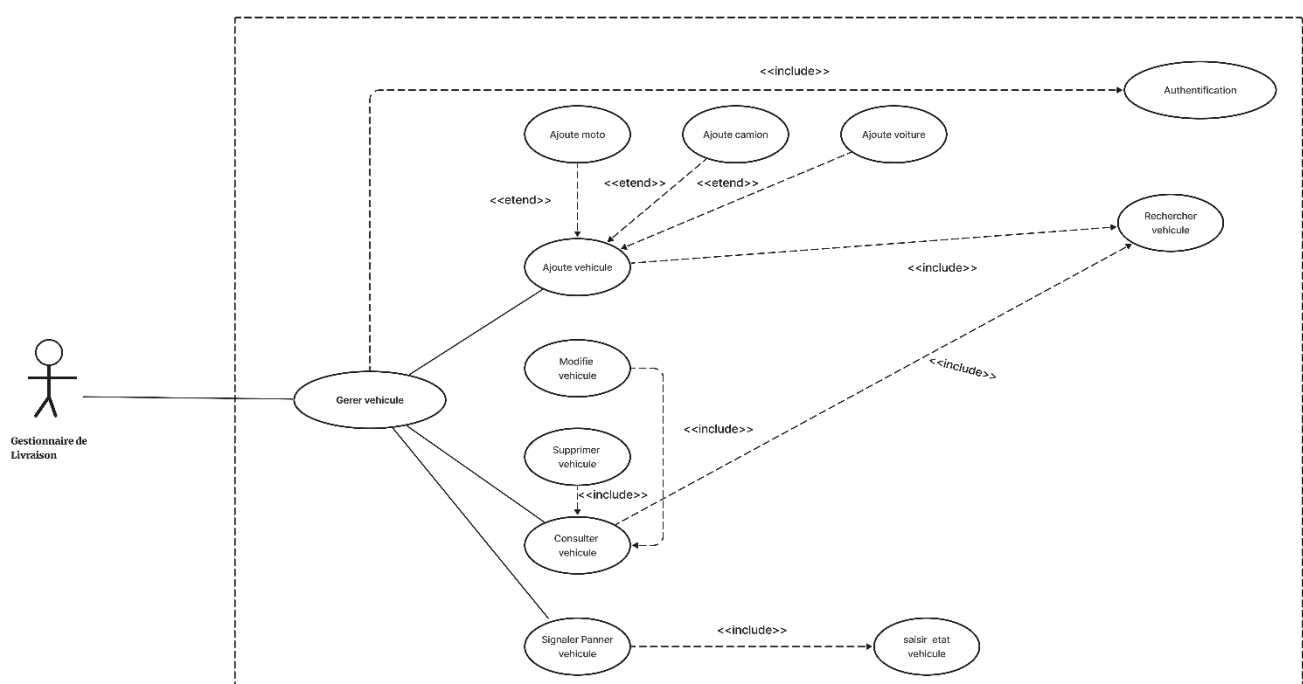


Figure 2 cas d'utilisation (cas de gestion véhicule de livreur).

Annexe :

3. Cas d'utilisation : Gérer Map .

Ajouter un nouveau chemin :

Le gestionnaire saisit les détails du nouveau chemin (points de départ et d'arrivée, distance, durée estimée, etc.).

Le système enregistre le chemin dans la base de données.

Supprimer un chemin :

Le gestionnaire sélectionne le chemin à supprimer.

Le système confirme la suppression et retire le chemin de la base de données.

Modifier les informations d'un chemin (changer) :

Le gestionnaire recherche et sélectionne le chemin dont il souhaite modifier les informations.

Il modifie les détails nécessaires (distance, durée estimée, etc.).

Le système enregistre les modifications dans la base de données.

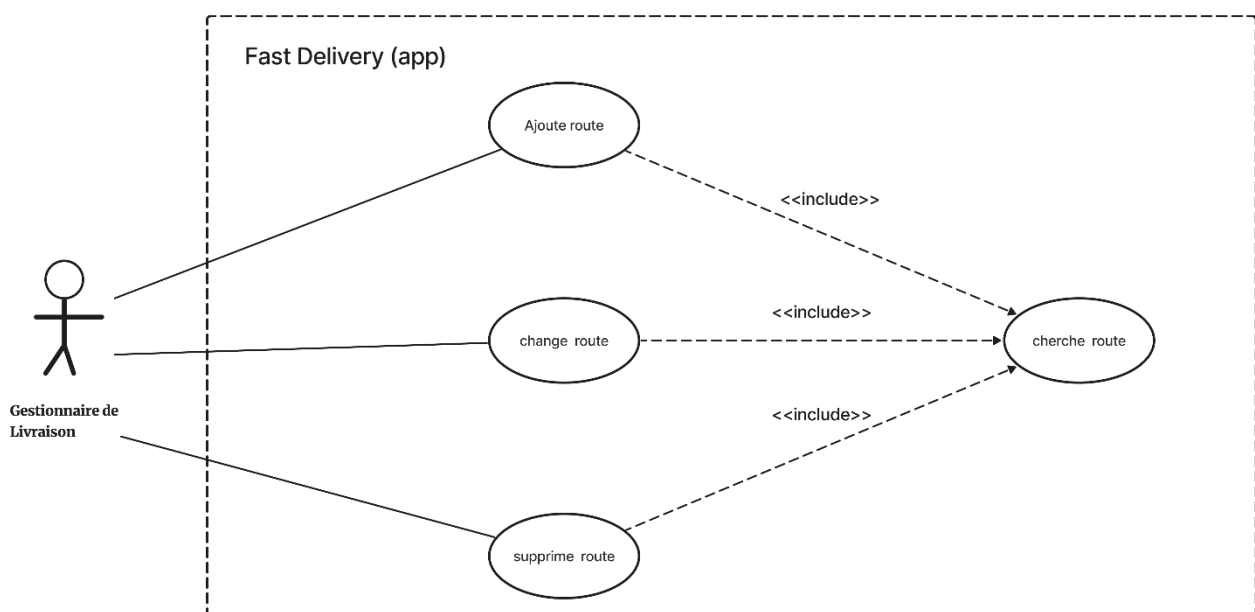


Figure 3 cas d'utilisation (cas gestion map).

Annexe :

Diagramme de séquence :

1. Gestion Véhicule :

i. Modification les informations d'un véhicule :

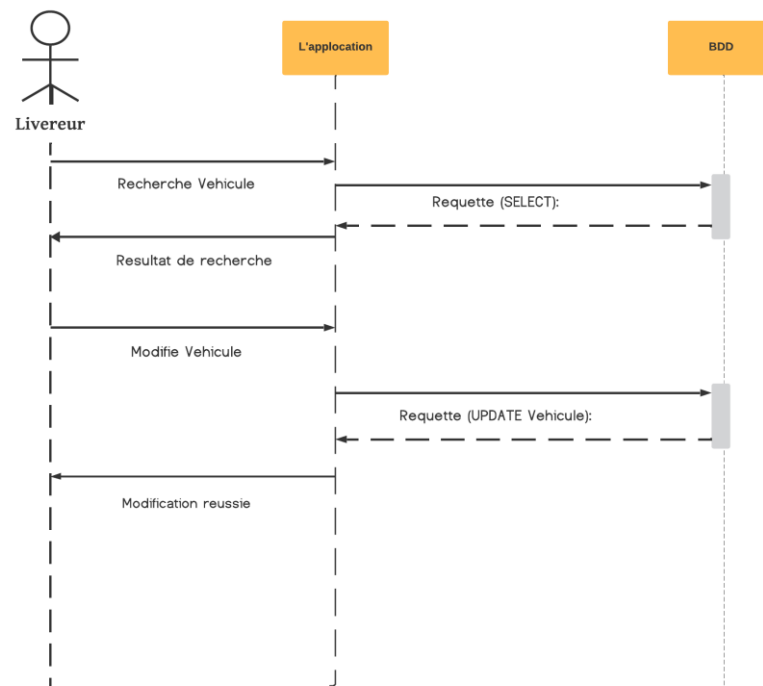


Figure 4 diagramme de séquence (cas de modification d'un véhicule de livreur).

ii. Suppression les informations d'un véhicule :

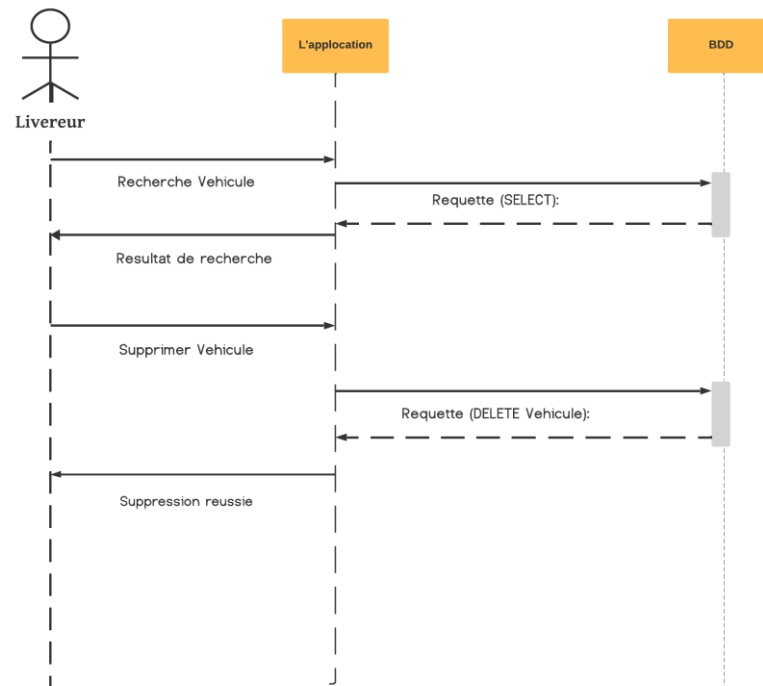


Figure 5 diagramme de séquence (cas de supprimer d'un véhicule de livreur).

Annexe :

2. Gestion Route :

i. Modification sur la route :

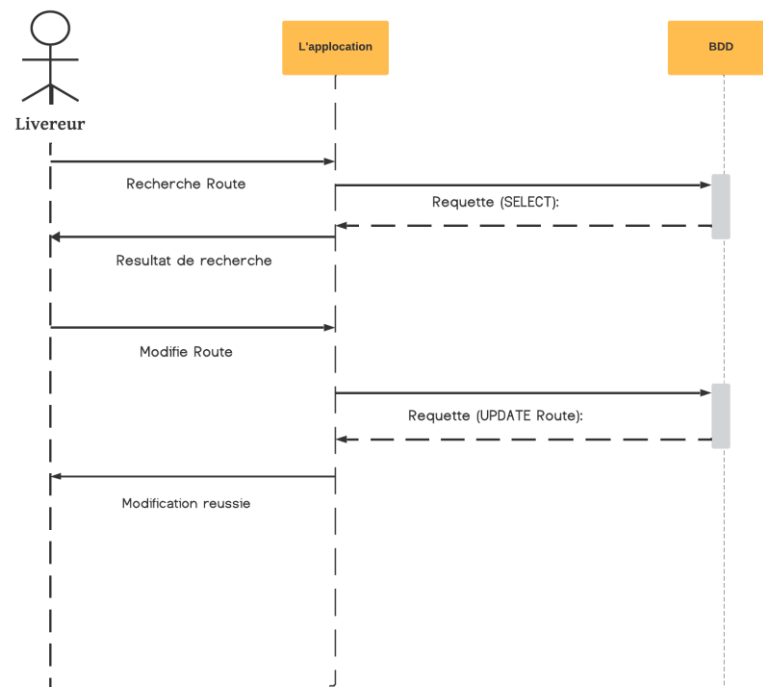


Figure 6 diagramme de séquence (cas de modifier la route du livreur).

Annexe :

ii. Suppression la route :

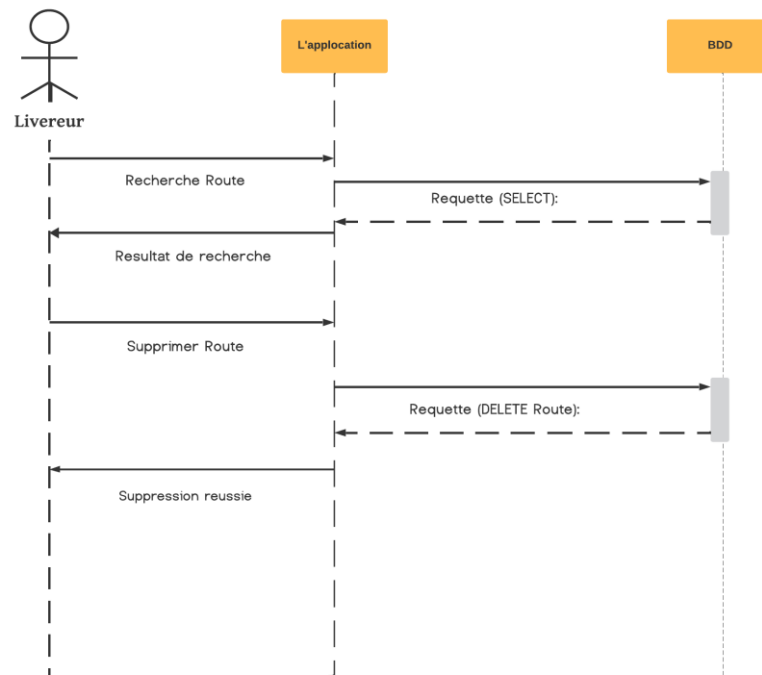


Figure 7 diagramme de séquence (cas de supprimer la route du livreur).

3. Gestion Client :

i. Modification les informations d'un client :

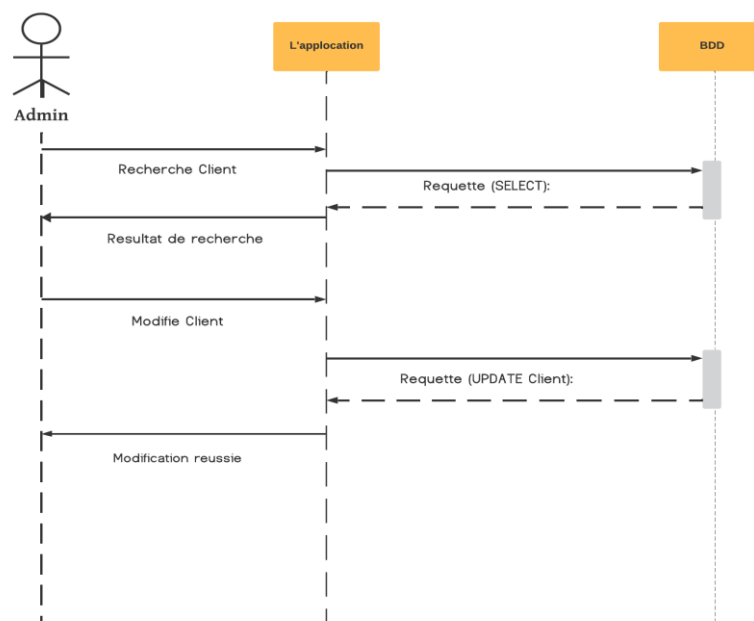


Figure 8 diagramme de séquence (cas de modifier un client).

ii. Suppression les informations d'un client :

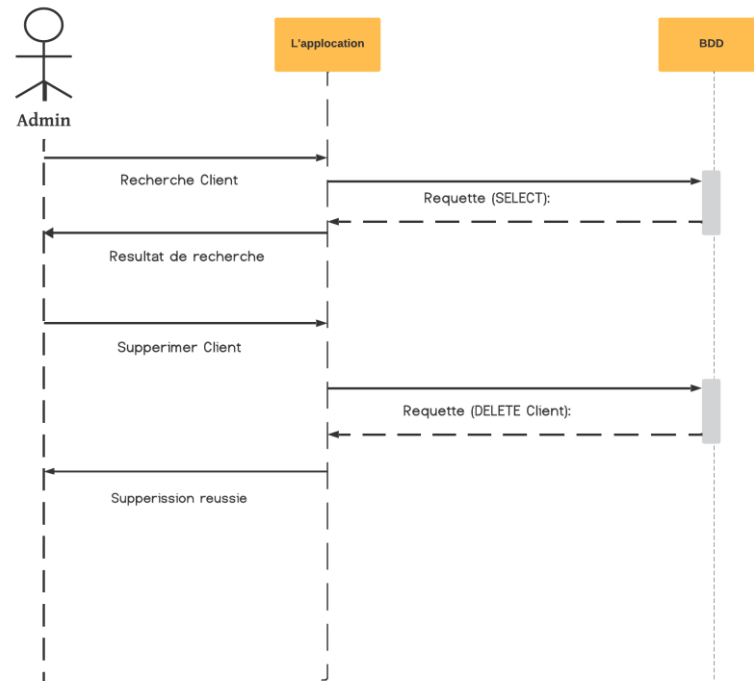


Figure 9 diagramme de séquence (cas de supprimer un client).

4. Gestion Livreur :

i. Modification les informations d'un livreur :

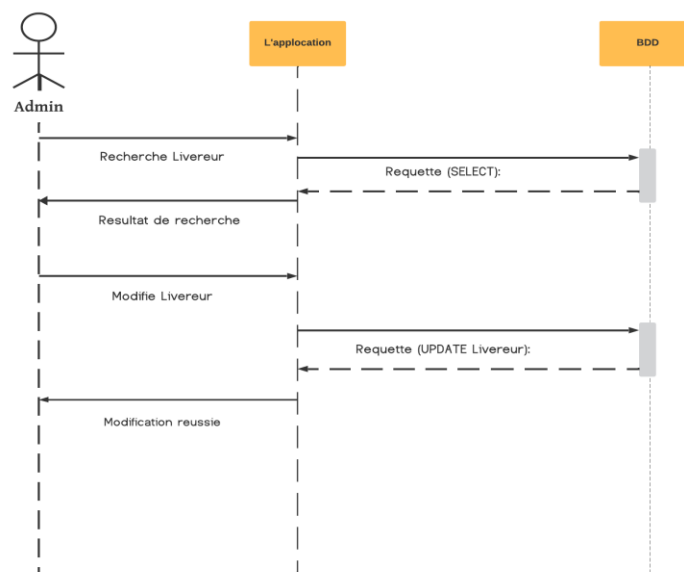


Figure 10 diagramme de séquence (cas de modifier un livreur).

ii. Suppression les informations d'un livreur:

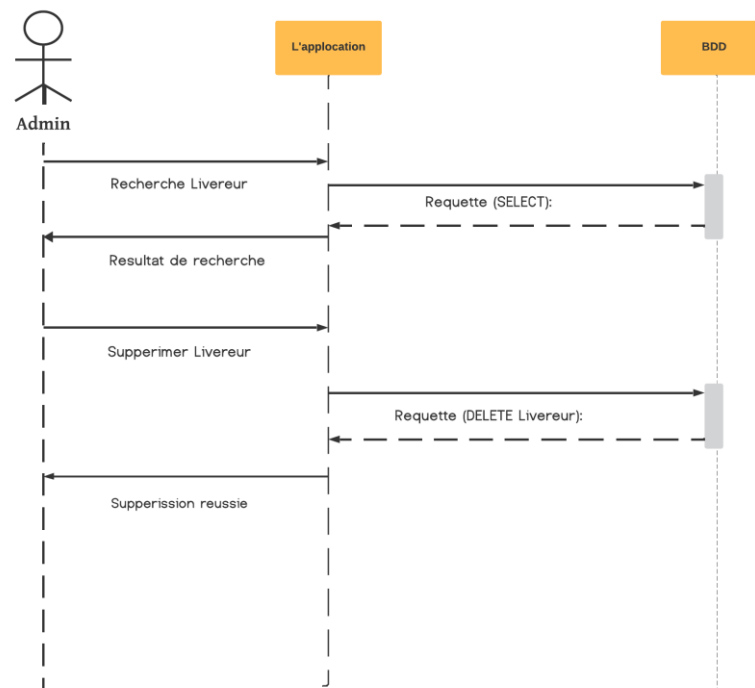


Figure 11 diagramme de séquence (cas de supprimer un livreur).