



École Supérieure Multinationale des Télécommunications



BÉNIN



BURKINA FASO



GUINÉE



MALI



MAURITANIE



NIGER



SÉNÉGAL

MÉMOIRE FIN DE FORMATION POUR L'OBTENTION DU DIPLÔME DE LICENCE PROFESSIONNELLE



OPTION : TELECOMMUNICATIONS ET INFORMATIQUE
SPÉCIALITÉ : DEVELOPPEMENT D'APPLICATIONS REPARTIES

THÈME

Mise en place d'une plateforme de Gestion de lavage automobile

Sous la direction de

M. Jean-Marie PREIRA,
Enseignant - Chercheur à l'ESMT

Présenté et soutenu par

M. Abibou Falilou NGOM

Et

M. Assima Geovani TAGBA

Promotion 2017 -2020

Août 2020

DEDICACE



Nous dédions tout ce travail fourni à :

Nos parents pour leurs sacrifices dans le but de nous mettre dans les meilleures conditions de réussite possibles, leurs prières et leurs précieux conseils,

A nos amis envers qui nous témoignons toute notre gratitude, qui n'ont jamais cessé de nous encourager et de croire en nous,

L'ensemble des étudiants de DAR, nos compagnons de tous les jours, nous avons compris que l'union fait la force,

Merci pour votre soutien.

Que Dieu le tout puissant veille sur vous et continue de vous bénir !

REMERCIEMENTS



A l'issue de ce travail, nous remercions, en premier lieu, le bon Dieu de nous avoir donné la force et le courage de le mener à terme en cette période difficile.

Nous tenons, également, à exprimer notre sincère reconnaissance et notre profonde gratitude à nos parents sans qui rien de tout cela n'aurait été possible et qui par la droiture de leur éducation nous permettent de délivrer ce travail aujourd'hui.

Notre professeur titulaire M. PREIRA pour sa tolérance, sa disponibilité et le suivi tout au long de notre cursus,

L'ensemble du corps professoral de l'ESMT, pour leur disponibilité, leur patience, leurs conseils, leur intérêt incontestable et leur soutien tout au long de nos 3 années de formation. La promotion de LPTI se souviendra toujours de vous.

TABLE DES FIGURES

Figure 1: Diagramme de cas d'utilisation	14
Figure 2: Diagramme de communication d'authentification	15
Figure 3: Diagramme de communication d'ajout.....	16
Figure 4: Diagramme de communication de modification	16
Figure 5 Diagramme de communication de suppression.....	17
Figure 6: Diagramme de communication de recherche	18
Figure 7: Diagramme de communication de notification	18
Figure 8: Diagramme de communication de consultation	19
Figure 9: Diagramme de séquence de cas d'utilisation d'authentification	20
Figure 10: Diagramme de séquence de cas d'utilisation d'ajout	21
Figure 11: Diagramme de séquence du cas d'utilisation de modification.....	22
Figure 12: Diagramme de séquence du cas d'utilisation de consultation	22
Figure 13: Diagramme de séquence du cas d'utilisation de recherche	23
Figure 14: Diagramme de classe.....	25
Figure 15: Logo Symfony.....	28
Figure 16: Logo Django.....	28
Figure 17: Logo Laravel	28
Figure 18: Logo ReactJS.....	28
Figure 19: Logo Angular	29
Figure 20: Logo Ionic	29
Figure 21: Logo Flutter.....	29
Figure 22: Logo React Native.....	30
Figure 23: Logo Firebase.....	31
Figure 24: Application Shazam	32
Figure 25: Application Le Figaro – capture d'écran 1.....	32
Figure 26: Application Le Figaro – capture d'écran 2.....	32
Figure 27: Application Skyscanner.....	33
Figure 28: Logo Firebase Realtime Database.....	33
Figure 29: Capture d'écran - SplashScreen	36
Figure 30: Capture d'écran - Presentation.....	36
Figure 31: Capture d'écran – SignInScreen	37
Figure 32: Capture d'écran - StatusCheck.....	38
Figure 33: Capture d'écran - Status.....	39
Figure 34: Capture d'écran - Search.....	40
Figure 35: Capture d'écran - PlaceCheck.....	41
Figure 36: Capture d'écran - PlaceSet.....	42
Figure 37: Capture d'écran - AddPlace	43
Figure 38: Capture d'écran - AddWash.....	44
Figure 39: Capture d'écran - ListWash	45
Figure 40: Capture d'écran - WashDetail.....	46

GLOSSAIRE

Sigle ou abréviation	Signification
Android	Est le système d'exploitation mobile créé par Google qui équipe la majorité des smartphones et tablettes tout comme iOS
DOM	Document Object Model : est une interface de programmation normalisée par le W3C, qui permet à des scripts d'examiner et de modifier le contenu du navigateur web
iOS	Est le système d'exploitation d'Apple qui équipe actuellement les smartphones iPhone et les tablettes iPad
JSON	JavaScript Object Notation, est un format de données textuelles dérivé de la notation des objets du langage JavaScript.
JSX	C'est une extension syntaxique de JavaScript
PHP	Pour Hypertext Preprocessor, désigne un langage informatique, ou un langage de script, utilisé principalement pour la conception de sites web dynamiques
Sérialisation	dans le cadre du stockage de données, on la définit comme étant le processus de traduction de structures de données ou objet état dans un format qui peut être stocké ou transmis reconstruite plus tard.
SQL	Structure Query Language : est un langage informatique normalisé servant à exploiter des bases de données relationnelles
UML	Unified Modeling Language, est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système

AVANT-PROPOS



L'Ecole Supérieure Multinationale des Télécommunications (ESMT) a pour vocation de former des diplômés (Techniciens supérieurs, Licences Professionnelles, Ingénieurs, Masters, Masters spécialisés) dans les domaines techniques et managériaux des télécommunications/TIC. Elle accueille en formation initiale ou continue des stagiaires qui proviennent de l'ensemble des pays francophones d'Afrique, recrutés au niveau des écoles, des universités ou directement chez les opérateurs de télécommunications.

SOMMAIRE

AVANT-PROPOS	6
SOMMAIRE	7
INTRODUCTION	8
PRESENTATION GENERALE.....	9
A. Les espaces de lavages de automobiles à Dakar.....	10
B. Présentation du sujet.....	10
C. Cahier de charge.....	11
ANALYSE ET CONCEPTION.....	12
A. Diagramme de cas d'utilisation :.....	13
B. Diagramme de communication :.....	14
C. Diagramme de séquences :.....	19
D. Diagramme de classe.....	24
CHOIX DES TECHNOLOGIES.....	26
A. Généralité sur les frameworks	27
B. Cas du framework React Native.....	30
C. Gestion des données de l'application.....	31
REALISATION DE LA PLATEFORME	34
A. Mise en place de l'environnement de développement.....	35
B. Présentation des interfaces graphiques	35
CONCLUSION.....	47
WEBOGRAPHIE	48
ANNEXE	49
A. ANNEXE 1	50
B. ANNEXE 2	51
C. ANNEXE 3	52



INTRODUCTION

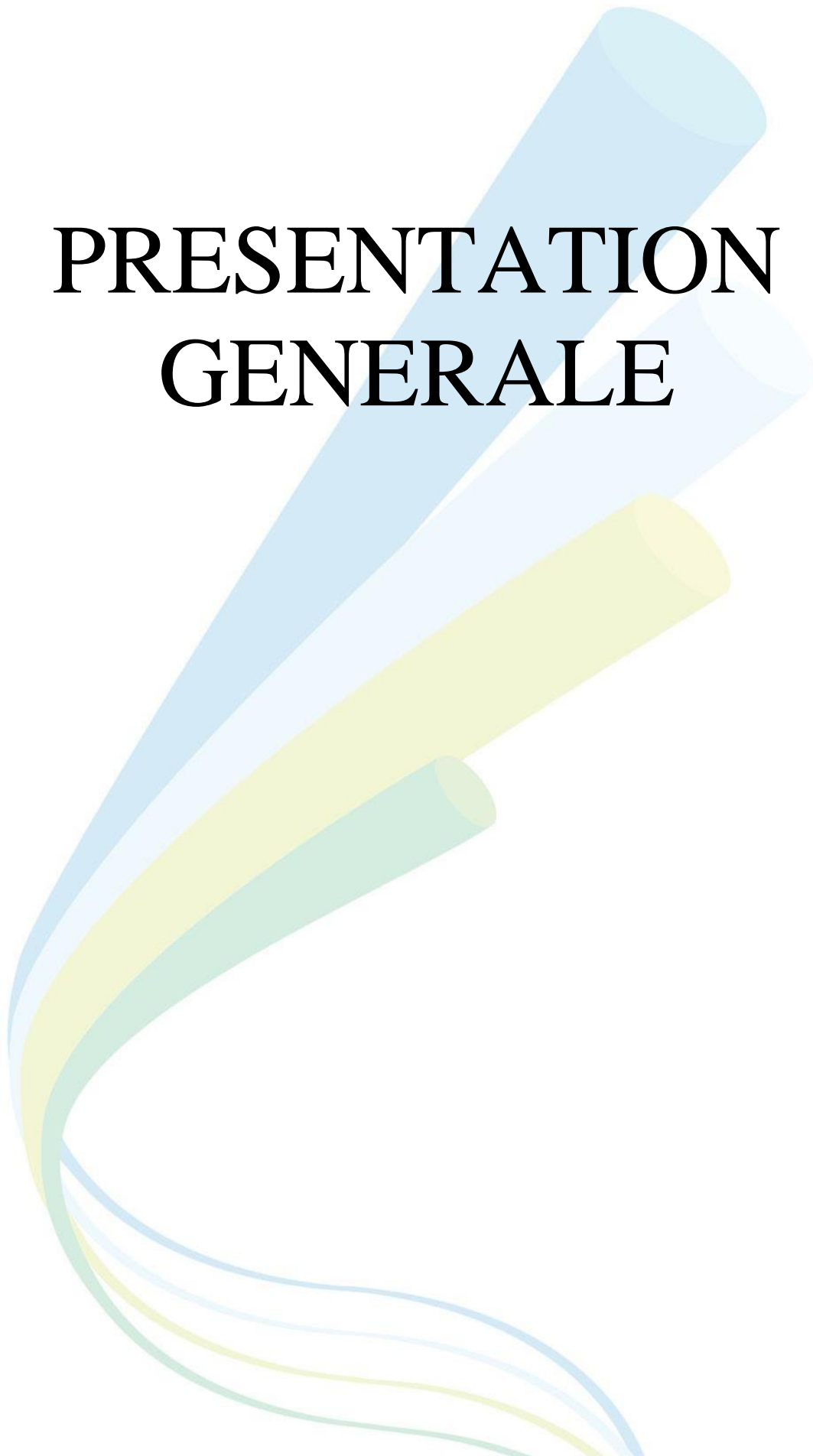
Le monde connaît actuellement une évolution technologique qui ne cesse de croître de façon exponentielle dans tous les secteurs et domaines d'activités, que ce soit dans le secteur formel tout comme dans le secteur informel. Ceci implique donc une révolution commerciale, qui bouleverse les rapports entre les producteurs, les distributeurs et les clients, aux profits de tous. Le facteur sociétal est désormais dominant à cause de la société qui devient une vraie partie prenante de l'entreprise, Internet et les réseaux sociaux obligent.

L'informatique qui est en constante développement, propose un vaste étendu de technologies qui peuvent s'utiliser dans presque tous les domaines possibles. Elle joue un rôle important dans le développement des entreprises et d'autres entités. Les méthodes de gestions employées jusqu'ici pour la plupart des entreprises ont prouvé leur efficacité, mais elles présentent néanmoins certaines insuffisances telles que le manque d'utilisation des solutions digitales qui ne cessent de montrer leur importance dans chaque domaine d'activité, le manque des services en ligne pour permettre aux consommateurs d'en bénéficier sans le moindre effort à travers des plateforme web, des applications mobiles, etc ; le manque de suivi, d'assistance et de gérance des activités des clients, de la qualité des services fournis par ces tiers sont des contraintes qui se soulèvent fréquemment pour la plupart.

C'est au vu de toutes ces contraintes citées plus haut que s'inscrit notre sujet qui vise à développer une application de gestion de Carwash étant un business très en vogue de nos jours. Elle cherche à faciliter la gestion des lavages et le suivi des processus qui ont lieu par les clients de ces derniers.

Tout au long de notre rédaction, nous tenterons d'abord de présenter la structure Carwash, de faire dans un premier temps une étude théorique plus approfondie sur le sujet qui justifiera le choix des technologies utilisées pour la réalisation de la solution ; dans un second temps nous aborderons l'étude pratique en ce qui concerne la mise en œuvre de la solution qui est la conception d'une application de gestion qui pourra répondre aux problèmes posés par le sujet.

PRESENTATION GENERALE



I. Les espaces de lavages de automobiles à Dakar

Le business/domaine du lavage automobile est en plein essor au Sénégal du fait de la nécessité des personnes à maintenir leur véhicule présentable et en état. C'est un domaine appartenant au secteur informel du fait de leur utilisation d'un nombre réduit de travailleurs ayant des qualifications acquises hors du système scolaire, du caractère familial de l'entreprise mise en place et du faible niveau d'organisation.

Les lavages sur lesquels porteront notre étude se trouvent aux Maristes et à Grand Yoff appartiennent au même propriétaire. Comme la plupart des lavages ils sont aménagés sur des terrains appartenant soit au propriétaire de l'entreprise soit à une tierce personne avec qui elles ont une entente pour s'implanter. Placés en bordure de route et aménagés avec une espace de rangement pour le matériel et un espace couvert (03 places) pour le lavage de véhicules, le reste de la surface sert de parking pour les véhicules prêts ou en attentes.

Pour lancer une telle activité il faut un investissement assez conséquent en ce qui concerne non seulement l'aménagement mais aussi la fourniture du matériel de lavage et des produits d'entretien. Cependant il y a un risque de perte en cas de faillite de l'activité, car il s'inscrit dans le système de fonctionnement du secteur informel avec un nombre réduit d'employé :

- Le gérant
- 3 à 4 employés chargés du travail manuel

En effet il s'agit d'une activité familiale qui a été lancée et se gère en interne. Sur le plan organisationnel, la comptabilité et toutes les activités administratives se font sur supports papiers, ce qui implique des risques d'erreurs, d'oubli ou même de détournement des ressources mises à la disposition des employés.

II. Présentation du sujet

1. Problématique

Dans le cadre de la réalisation de notre mémoire nous sommes allés sur le terrain à la rencontre des acteurs de ce secteur. Après avoir eu à discuter avec un certain nombre de propriétaires de lavages automobiles dans la zone des **Maristes**, de **Grand Yoff** et de **Sacré-cœur**, nous avons pu faire ressortir certaines contraintes que les artisans de ce domaine rencontrent. Ces derniers sont pour le plus souvent confrontés aux mêmes problèmes ; ils sont souvent d'ordre organisationnel :

- L'impossibilité de contrôler l'arrivée des véhicules,
- Les problèmes de gestion de leur espace de lavage,
- La durée variable des processus de lavage,
- L'utilisation du support papier pour la comptabilité et le suivi de l'activité,
- Le besoin de fidéliser certains clients en rapport aux services récurrents.

Il nous a aussi été donné l'occasion de discuter avec les clients qui sont les principaux utilisateurs de ces espaces, auprès de qui nous avons pu recueillir aussi leurs préoccupations qui sont du même ordre que ceux des gérants des lavages :

- Le manque d'informations sur la disponibilité de place,
- Le manque de suivi sur son véhicule lorsqu'il est laissé sur place,
- La difficulté à trouver les différents lavages.

2. Solution proposé

Suite à toutes ses contraintes qui ont été évoquées plus haut, nous proposons de mettre en place une plateforme qui pourra répondre aux besoins de la structure ainsi que de ses clients. Les démarches à suivre sont nombreuses à cause des étapes par lesquelles il faut passer pour atteindre les objectifs fixés. Elle sera développée de façon à ce qu'elle puisse prendre en charge les demandes des utilisateurs de la plateforme et d'offrir la possibilité de suivre en temps réel l'état d'avancement du service demandé ; elle permettra de fournir certaines informations aux utilisateurs ; la structure pourra donc avoir une meilleure vue sur la qualité de leurs services. Il s'agit d'une application mobile, qui sera développée avec un framework de développement mobile multiplateforme pour permettre aux utilisateurs d'avoir l'application sur iOS et Android.

III.Cahier de charge

L'application que nous devons implémenter pour cette structure de lavage automobile devra répondre aux exigences suivantes :

- La fourniture d'informations concernant les places disponibles dans les différents espaces de lavage.
- Le renseignement sur l'état d'avancement des services demandés et historique de passage.
- La visualisation générale des différents lavages sur une carte géographique de la zone.
- La notification aux clients en fin de processus d'un service.
- La mise en place d'une partie d'administration permettant à la structure une vue globale sur les entrées des véhicules et leur gestion en leur sein.
- La possibilité de faire une réservation auprès de la structure à partir de l'application à titre d'extension



ANALYSE ET CONCEPTION

Cette partie sera consacrée aux étapes fondamentales pour le développement de notre application de gestion qui répondra aux attentes de la structure. Avant de se lancer dans cette étape de conception et de réalisation, il est important de cibler les tâches à accomplir. Dans le cas de notre application nous devons pouvoir, à partir de cette dernière, offrir les informations sur la structure et leurs emplacements géographiques sur une carte, renseigner les clients sur l'état leurs véhicules tout au long du processus lorsqu'ils sont pris en charge, mettre en place un espace administrateur pour la vue globale de la structure.

A cet effet pour la conception et la réalisation de notre application, nous avons choisis de modéliser avec le formalisme UML qui offre une flexibilité exprimée par l'utilisation des diagrammes. Pour la modélisation des besoins, nous utilisons les diagrammes UML suivant : Diagramme de cas d'utilisation, diagramme de séquence, diagramme de collaboration et diagramme d'activité.

I.Diagramme de cas d'utilisation :

Un diagramme de cas d'utilisation est un graphe, un ensemble de cas d'utilisation englobés par la limite du système, des associations de communication entre les acteurs et les cas d'utilisation, et des généralisations entre cas d'utilisation.

Il est destiné à représenter les besoins des utilisateurs par rapport au système.

- Identification des acteurs

Les acteurs d'un système sont les entités externes à ce système qui interagissent avec lui. Dans notre application, les acteurs qui interagissent avec le système sont le gestionnaire du lavage qui saisit les données et l'utilisateur final (client) qui visualise ces dernières en plus d'utiliser les autres fonctionnalités.

- Identification des cas d'utilisations

Un cas d'utilisation permet de décrire l'interaction entre l'acteur et le système. Chaque cas d'utilisation spécifie une séquence d'action, y compris des variantes, que l'entité réalise, en interagissant avec les acteurs de l'entité. Le cas d'utilisation est une description des interactions qui vont permettre à l'acteur d'atteindre son objectif en utilisant le système

Voici les cas d'utilisation de notre application :

- Authentification : l'application vérifie que l'utilisateur est bien ce qu'il prétend être et lui donne ensuite l'autorisation d'accès à la partie qui lui est réservé.
- Ajout : pouvoir ajouter les données relatives aux véhicules : plaque, formule choisie, tarif, date et heure. Mais aussi celles concernant les places.
- Modification : sert à modifier l'information dans la base de données
- Recherche : rechercher les informations relatives à son véhicule.
- Consultation : Visualiser sur une carte de géolocalisation les différents lavages qui se trouvent sur une zone.

- Notification : informer le client lorsque les procédures effectuées sur son véhicule arrivent à terme

D'où la présentation de notre diagramme de cas d'utilisation :

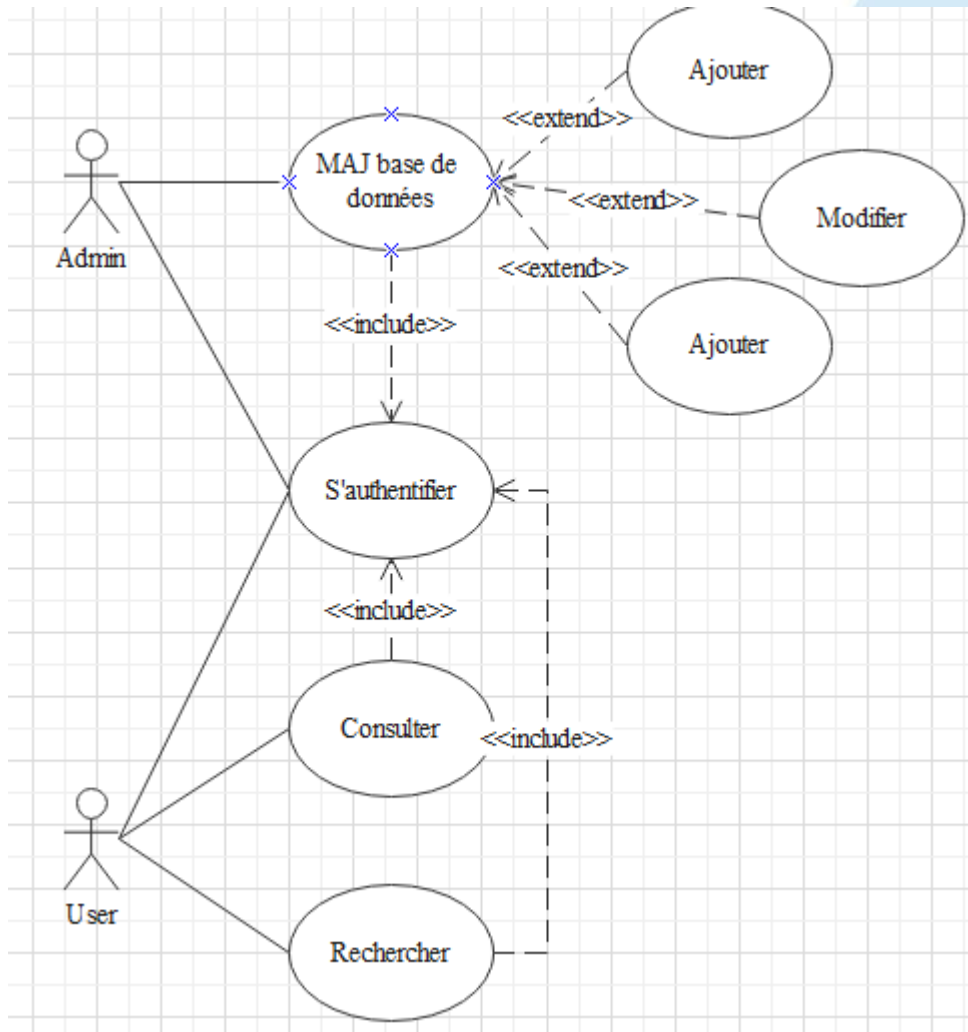


Figure 1: Diagramme de cas d'utilisation

II. Diagramme de communication :

Un diagramme de communication montre une interaction organisée autour d'un ensemble d'objets et de leurs liens. En revanche, un diagramme de collaboration ne montre pas le temps dans une dimension séparée ; ainsi la séquence des messages et les fils concurrents doivent être déterminés en utilisant les numéros de séquence.

C'est une autre représentation des scénarios des cas d'utilisation qui met plus l'accent sur les objets et les messages échangés.

1. Diagramme de communication d'authentification :

Ce diagramme décrit les messages échangés entre les différents objets pour montrer le fonctionnement de l'opération d'authentification : l'utilisateur/l'admin saisit l'identifiant et le mot de

passé puis le système vérifie sa validité, ensuite c'est le système qui retourne la page d'accueil de l'application à l'utilisateur/l'admin.

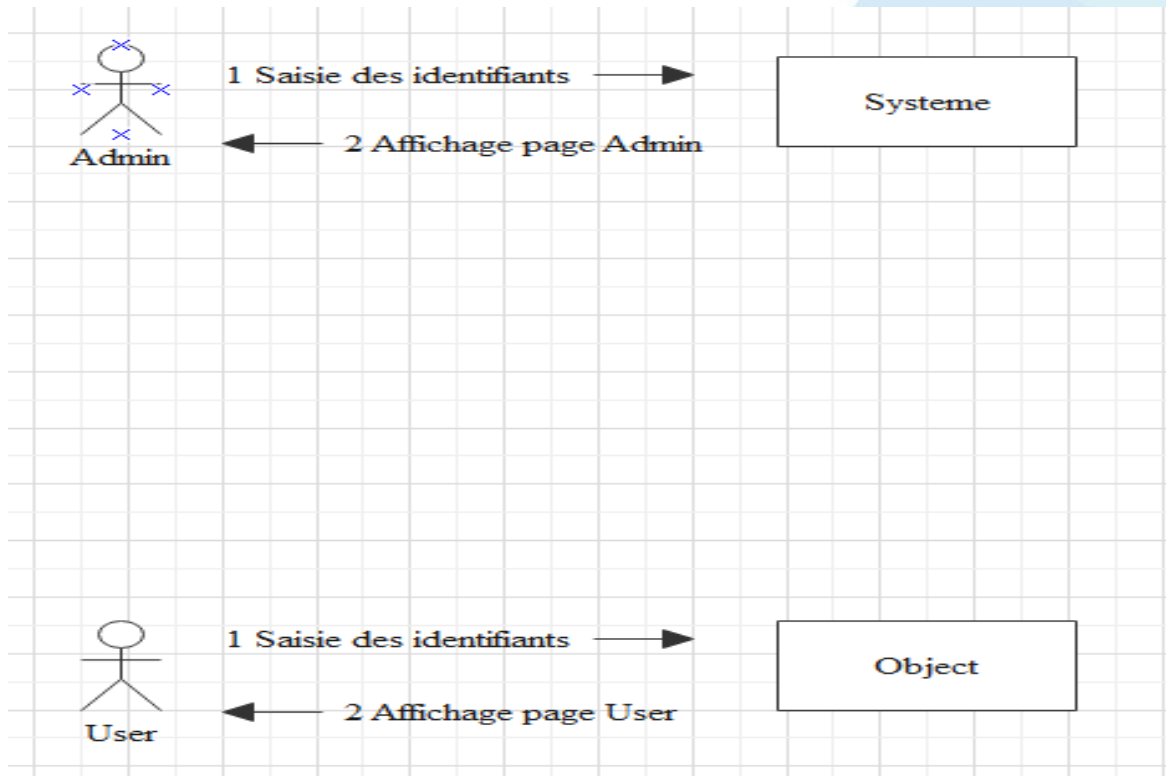


Figure 2: Diagramme de communication d'authentification

2. Diagramme de communication d'Ajout

Ce diagramme illustre la façon avec laquelle l'opération d'ajout d'un véhicule (matricule, formule, tarif, etc.) s'effectue.

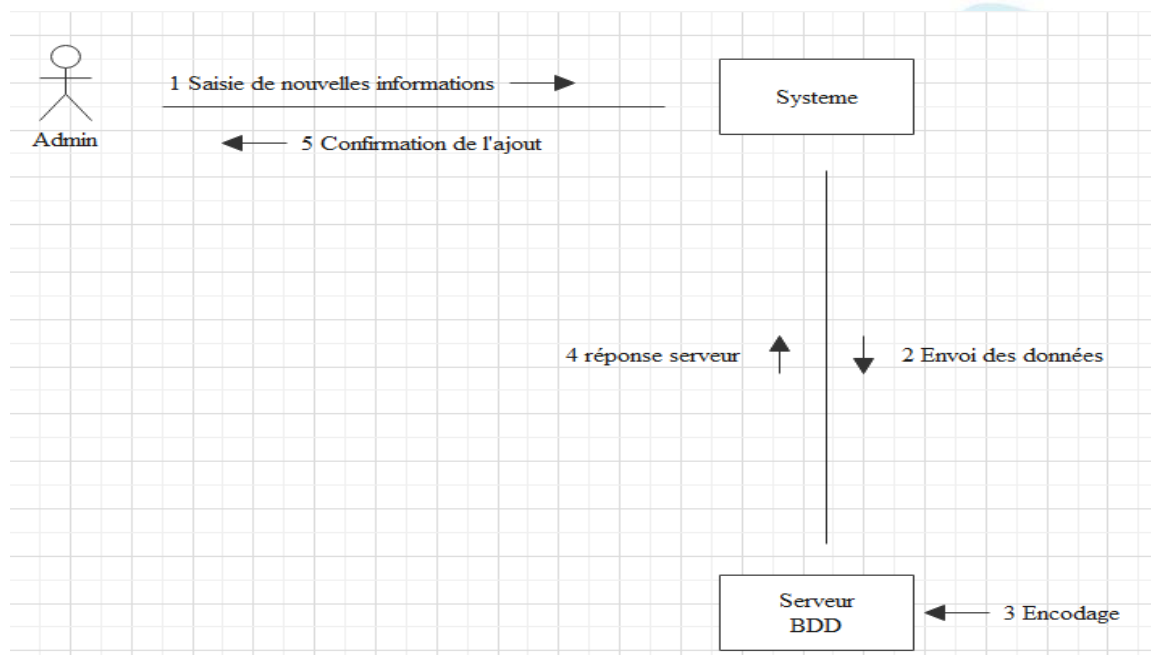


Figure 3: Diagramme de communication d'ajout

3. Diagramme de communication de Modification

Ce diagramme montre comment modifier une donnée : l'administrateur demande la modification en saisissant le matricule de la donnée et le système recherche cette dernière dans la base de données et l'affiche à l'administrateur qui va la modifier et l'enregistrer, le système la stocke ensuite dans la base de données.

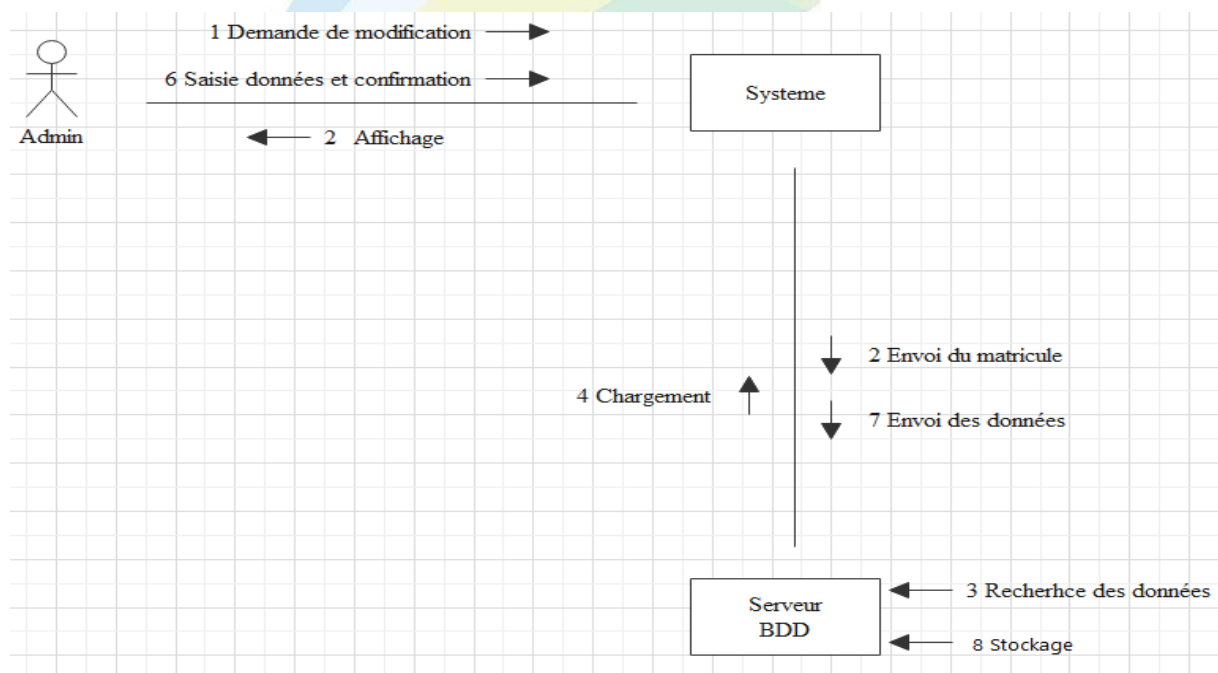


Figure 4: Diagramme de communication de modification

4. Diagramme de communication de suppression

Dans le cas de la suppression d'éléments, l'administrateur fait une requête de suppression d'une donnée précise. Cette requête est prise par le système et transféré au serveur qui la traite en supprimant la donnée.

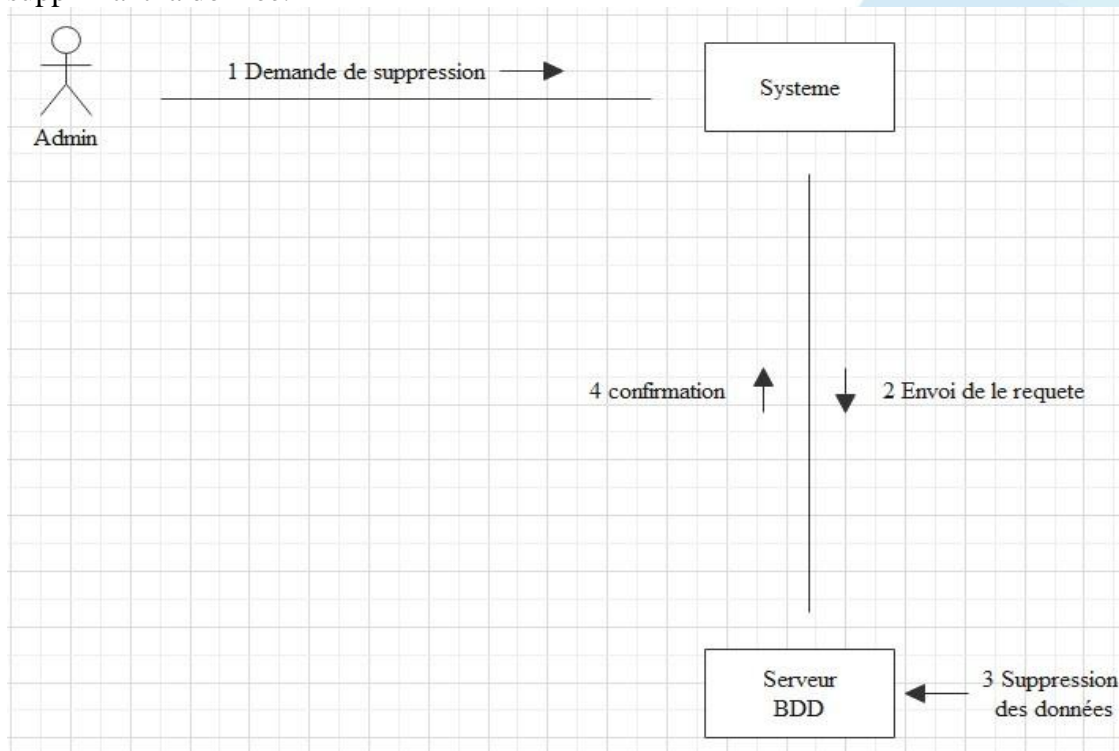


Figure 5 Diagramme de communication de suppression

5. Diagramme de communication de Recherche

Pour la recherche (Carwash au niveau de la carte de géolocalisation), il suffit que l'utilisateur ouvre la page concernée et le système effectue une requête au niveau de l'API de google pour lui afficher le résultat.

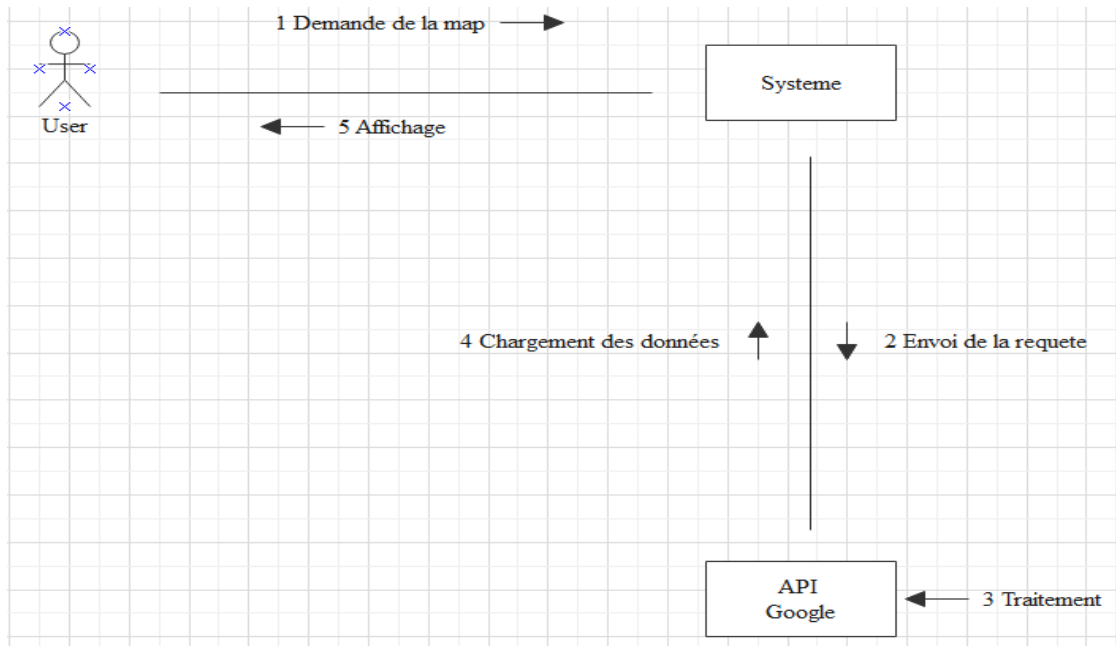


Figure 6: Diagramme de communication de recherche

6. Diagramme de communication de Notification

Ce diagramme illustre la façon dont l'utilisateur est notifié : lorsque son véhicule est prêt, l'administrateur fait une demande de notification au système à qui il fournit le message et le matricule. Le système entre en relation avec le serveur de messagerie qui lui se charge de l'envoi du message.

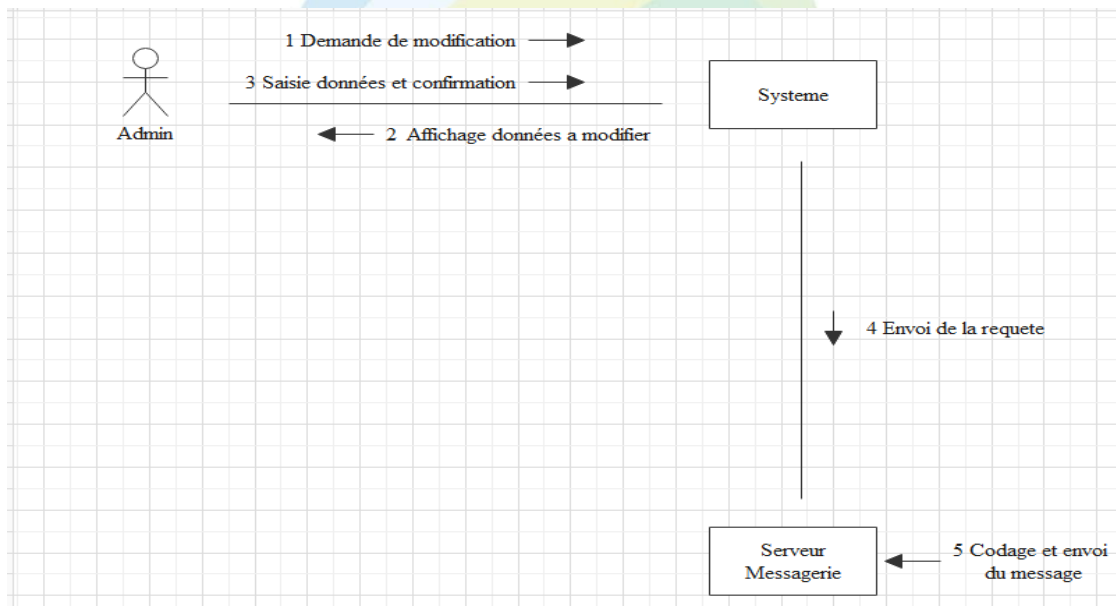


Figure 7: Diagramme de communication de notification

7. Diagramme de communication de Consultation

Pour la consultation l'utilisateur saisit le matricule de son véhicule, le système envoie le matricule au serveur de BDD qui recherche les informations et les envoie au système qui par la suite les affiche à l'utilisateur.

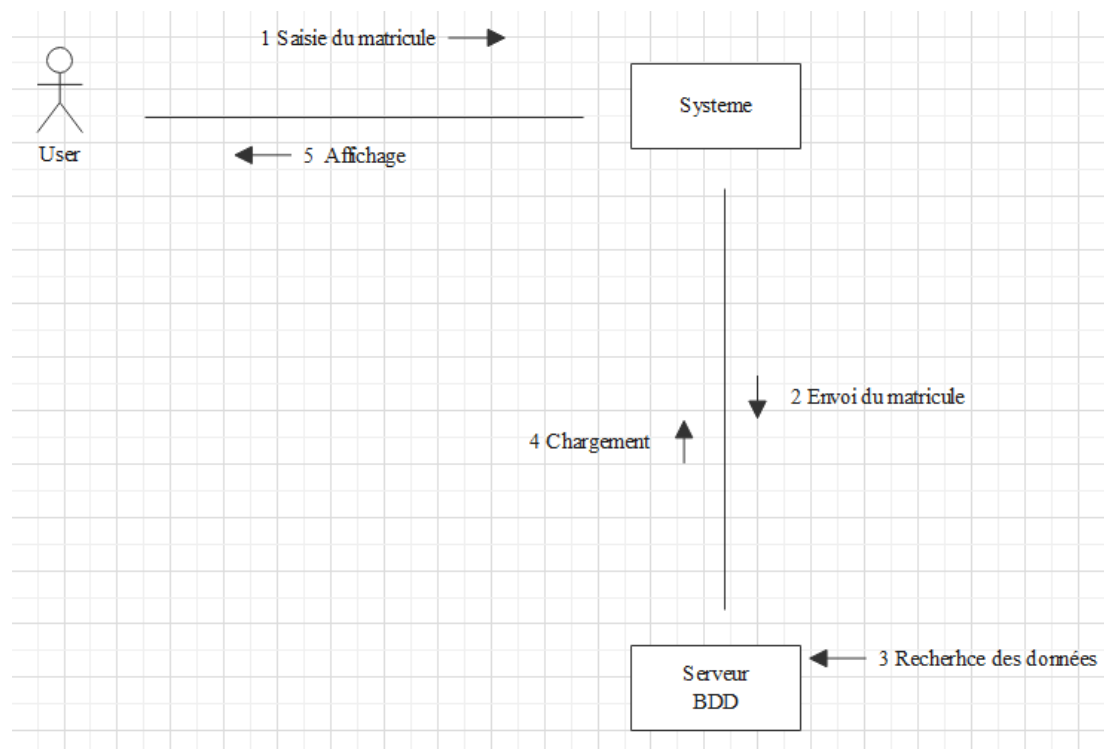


Figure 8: Diagramme de communication de consultation

III. Diagramme de séquences :

Il permet de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets. Un diagramme de séquence montre une interaction présentée en séquence dans le temps. En particulier, il montre aussi les objets qui participent à l'interaction par leur "ligne de vie" et les messages qu'ils échangent présentés en séquence dans le temps.

Voici quelques notions de base du diagramme :

- Scénario : une liste d'actions qui décrivent une interaction entre un acteur et le système.
- Interaction : un comportement qui comprend un ensemble de messages échangés par un ensemble d'objets dans un certain contexte pour accomplir une certaine tâche.
- Message : Un message représente une communication unidirectionnelle entre objets qui transporte de l'information avec l'intention de déclencher une réaction chez le récepteur.

1. Diagramme de séquence du cas d'utilisation "authentification"

- L'utilisateur/l'administrateur demande le formulaire d'authentification
- L'application affiche le formulaire d'authentification

- L'utilisateur/l'administrateur saisit l'identifiant et le mot de passe
- Le système en vérifie la validité
- L'application affiche la page d'accueil

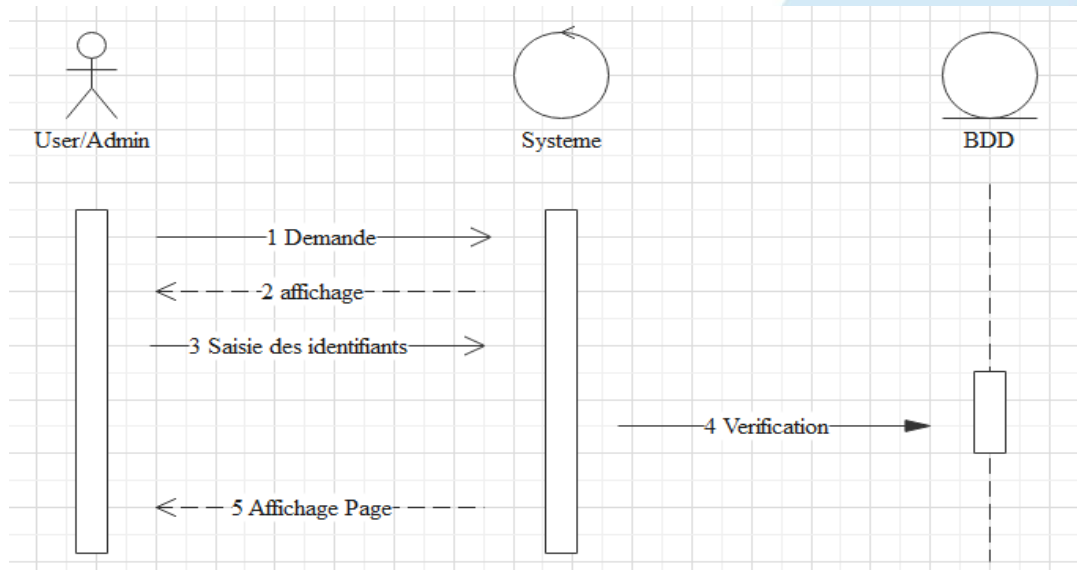


Figure 9: Diagramme de séquence de cas d'utilisation d'authentification

2. Diagramme de séquence du cas d'utilisation "Ajout"

- L'utilisateur demande le formulaire d'ajout.
- L'application affiche le formulaire d'ajout.
- L'utilisateur saisit les nouvelles données.
- L'application envoie la requête.
- L'application stocke les données au niveau de la base de données.
- L'application confirme l'enregistrement.

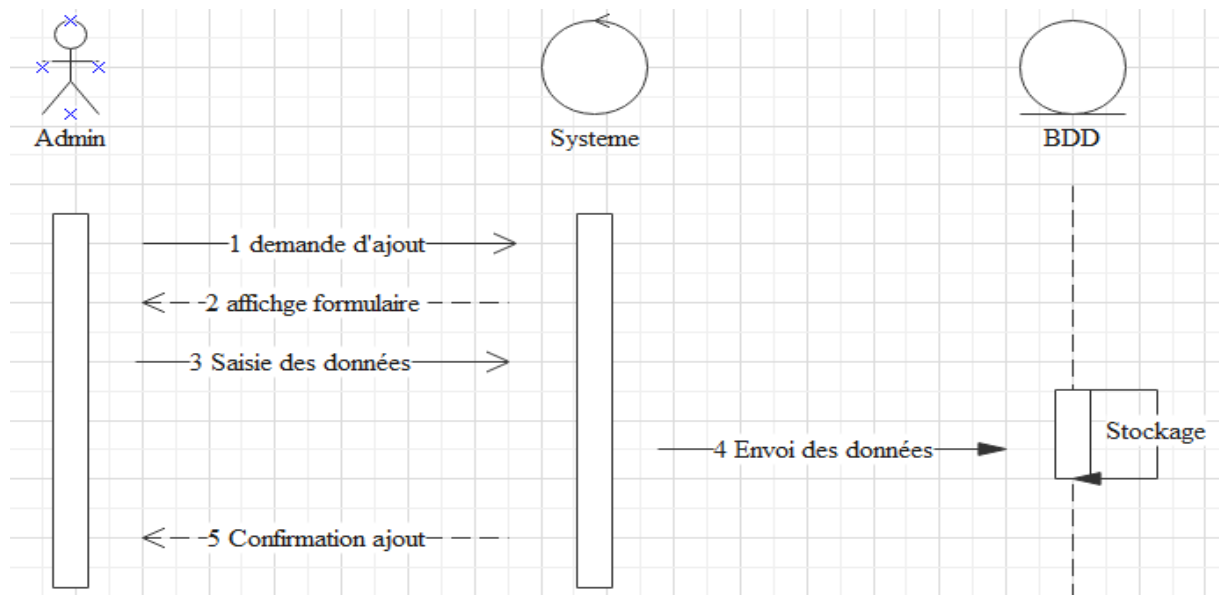


Figure 10: Diagramme de séquence de cas d'utilisation d'ajout

3. Diagramme de séquence du cas d'utilisation "Modification"

- L'utilisateur demande la modification.
- L'application affiche les données.
- L'utilisateur choisit les données à modifier.
- L'utilisateur saisit les nouvelles données.
- L'application envoie les nouvelles données à la BDD.
- Au niveau de la BDD se fait le stockage.
- Confirmation de la modification.

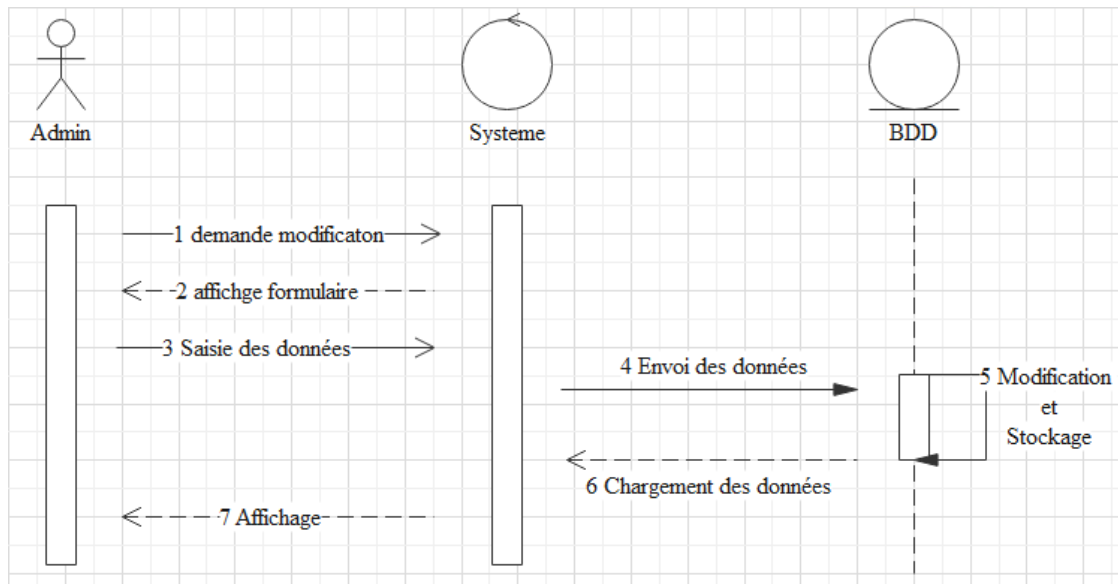


Figure 11: Diagramme de séquence du cas d'utilisation de modification

4. Diagramme de séquence du cas d'utilisation "Consultation"

- L'utilisateur saisit le matricule.
- L'application envoie le matricule
- La BDD recherche les données
- La BDD retourne les données à l'application.
- L'application affiche les informations.

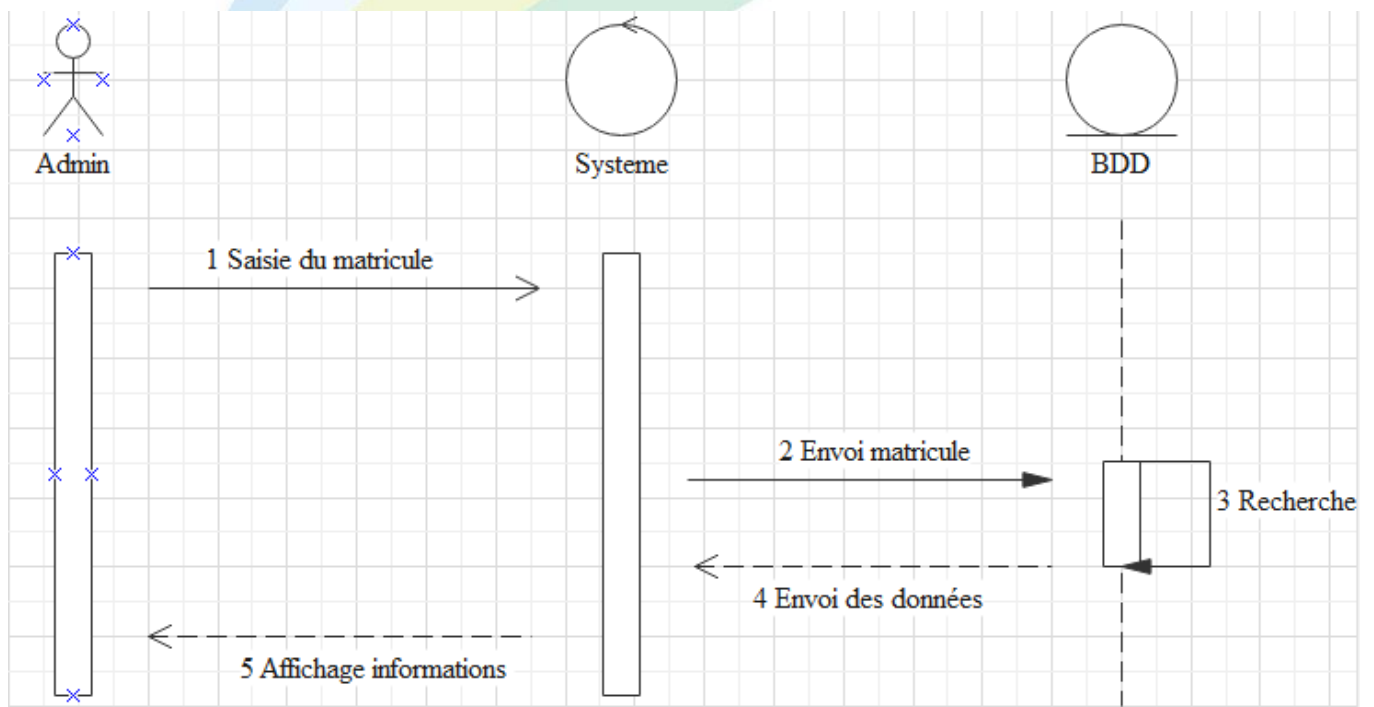


Figure 12: Diagramme de séquence du cas d'utilisation de consultation

5. Diagramme de séquence du cas d'utilisation "Recherche"

- L'utilisateur demande la carte de géolocalisation et les données.
- L'application envoie la requête.
- L'API charge les données
- L'API retourne les données à l'application.
- L'application affiche la carte de géolocalisation et les données.

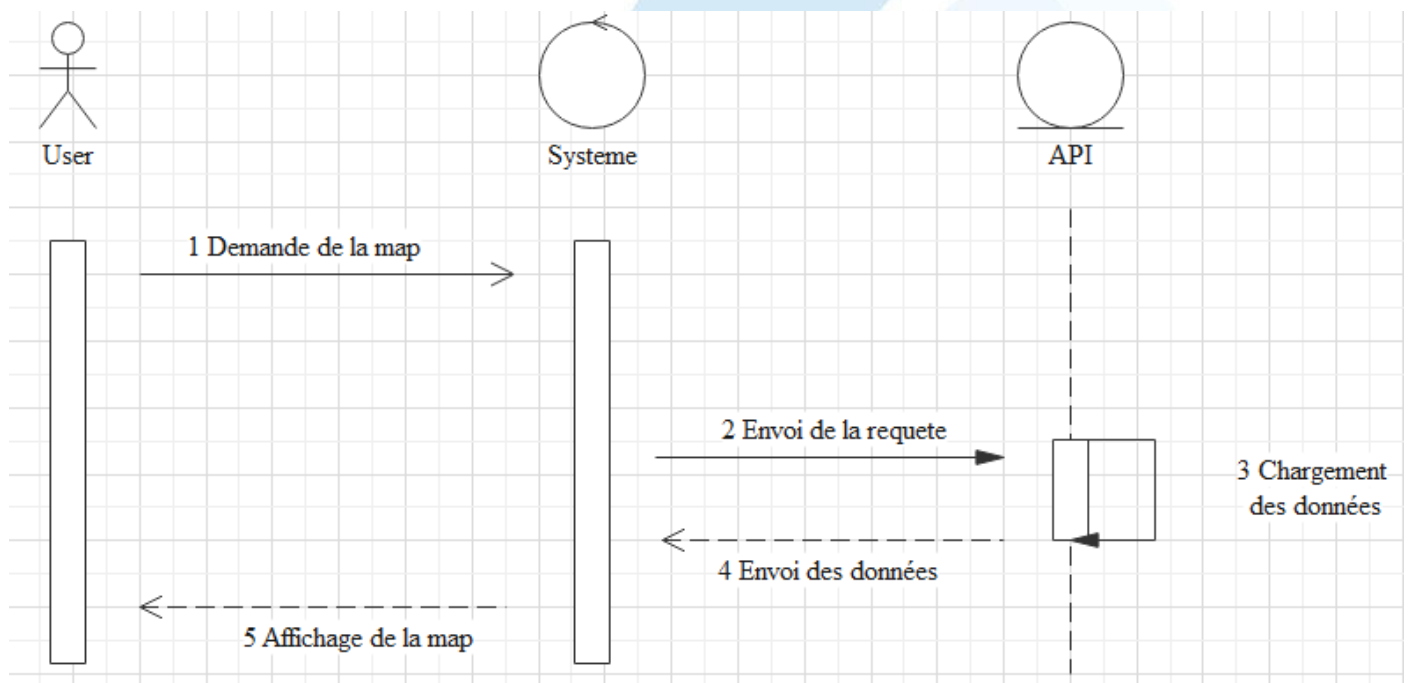


Figure 13: Diagramme de séquence du cas d'utilisation de recherche

IV. Diagramme de classe

C'est une collection d'éléments de modèle statique, tels que des classes, des interfaces et leurs relations, connectés entre eux comme un graphe.

Il représente la description statique du système en intégrant dans chaque classe la partie dédiée aux données et celle consacrée aux traitements. C'est le diagramme pivot de l'ensemble de la modélisation d'un système.

Identification des classes

Une classe est une description d'un groupe d'objets partageant un ensemble commun de propriétés (les attributs), de comportements (les opérations) et de relations avec d'autres objets (les associations et les agrégations).

Une classe contient :

Des attributs (ou champs, ou variables d'instances) : Les attributs d'une classe décrivent la structure de ses instances (les objets).

Des méthodes (ou opérations de la classe) : Les méthodes décrivent les opérations qui sont applicables aux instances de la classe.

Une agrégation est une association correspondant à une relation qui lorsqu'elle est lue dans un sens signifie "est une partie de" et lorsqu'elle est lue dans l'autre sens elle signifie "est composé de".

Les classes sur lesquelles se porte notre application sont les suivantes :

- Utilisateur
- Gestionnaire
- Client
- Laveur
- Véhicule
- Service de lavage

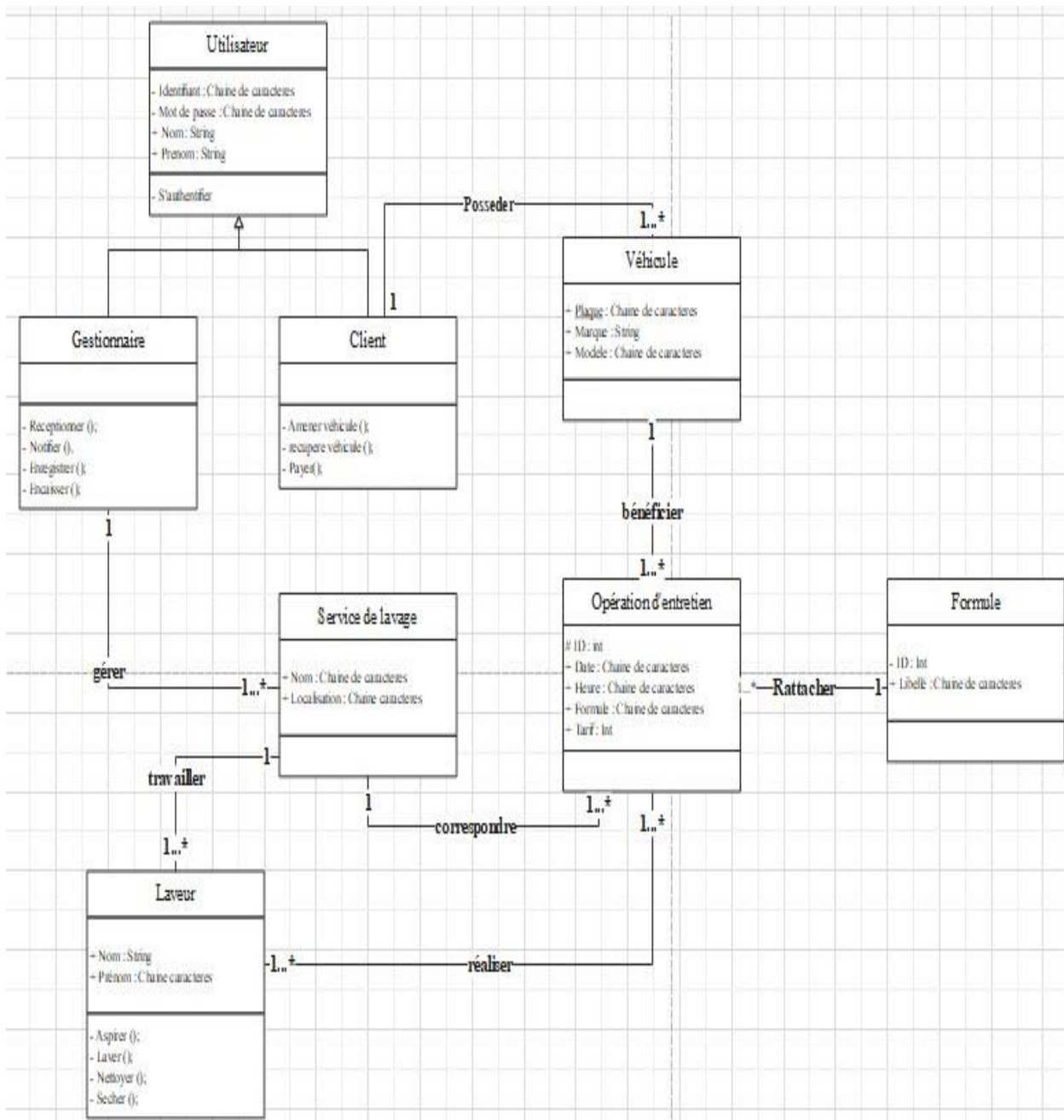


Figure 14: Diagramme de classe



CHOIX DES TECHNOLOGIES

I.Généralité sur les frameworks

1. Introduction générale aux frameworks

Dans le monde de l'informatique, un framework se définit comme une infrastructure logicielle. En d'autres termes c'est un ensemble logique de composants logiciels suivant une structuration bien définie. Il s'identifie par une architecture de base qui est déjà prêt à l'emploi. Il est généralement utilisé pour simplifier le travail des développeurs informatiques car il permet la réutilisation des codes, une standardisation de la programmation et la formalisation d'une architecture adéquate aux besoins des entreprises. Mais le concept de framework n'est pas vraiment une nouveauté. Son apparition remonte en Juillet 1984 avec le logiciel Framework de Ashton-Tate.

Il fut développé par Forefront et Ashton-Tate écrit en Assembleur, puis sera ensuite commercialisée par Ashton-Tate qui est une société américaine qui éditait des logiciels pour micro-ordinateurs. Framework n'était tout autre qu'un ensemble de programmes informatiques dédié spécialement pour les opérations basiques du travail de bureau à savoir la rédaction des lettres, des rapports, comptabiliser des factures, etc.

L'aspect innovateur de Framework était dans la structure de ses fichiers. Un composant Framework pouvait contenir une structure arborescente comme une table de matières, ou prenaient place les éléments provenant de tous les modules de la suite bureautique. Ainsi donc il pouvait réunir divers éléments relatifs à un même sujet mais de types différents dans un même fichier. Framework va être très vite concurrencé par Lotus Symphony de Lotus Software qui est un ensemble d'applications servant à créer, modifier et partager des documents bureautiques.

De nos jours la bureautique n'est pas le seul domaine où l'on note la présence des frameworks. L'univers du développement web et mobile, en est composé d'une multitude spécifique pour chaque langage de programmation. Ceci pour simplifier et booster le domaine du développement d'application. On note à cet effet des frameworks Front-end comme Angular, ReactJS, etc et Back-end tels que Symfony, Django. La contenance d'un framework est caractérisée par le type de programme et l'architecture cible pour lequel il est conçu.

2. Les différents types de frameworks

On peut donc noter plusieurs types de frameworks, à savoir :

- **Le framework d'infrastructure système** : utilisé pour développer des systèmes d'exploitation, des interfaces graphiques, des outils de communication.
- **Le framework d'intégration intergicielle** : utilisé pour mettre à disposition un groupe d'applications différents
- **Le framework d'entreprise** : utilisé pour développer des applications spécifiques au secteur d'activité de l'entreprise.
- **Le framework de gestion de contenu** : utilisé pour construire la base d'un système de gestion de contenu pour la création, la collecte, le classement, le stockage et la publication de biens numérisés.

Les frameworks se font de plus en plus présents et leurs utilisations est également fréquents à cause des divers avantages qu'ils présentent.

3. Exemples de frameworks

Les frameworks sont de plus en plus sollicités de nos jours dans plusieurs domaines où ils ont déjà prouvé leur efficacité.

a) Symfony

C'est un framework MVC open-source, il a été développé par une entreprise logiciel appelé SensioLabs et est écrit en PHP. C'est l'une des meilleures technologies pour développer des projets avec des exigences assez complexes. Il facilite le développement et offre une meilleure flexibilité.



Figure 15: Logo Symfony

b) Django

Ecrit en langage Python, c'est une technologie côté serveur. Il a été mis sur pieds par des développeurs expérimentés. Pour mettre en place des sites d'actualités, des systèmes de gestion de contenu, etc. La protection des sites est prise en charge de façon automatique par un moyen sécurisé et fiable pour la gestion des utilisateurs.



Figure 16: Logo Django

c) Laravel

Tout comme Symfony, il est écrit en PHP composé de plusieurs bibliothèques à l'image de d'autres frameworks. Il permet de gérer les systèmes de cache, l'envoi d'emails, les sessions utilisateurs, la pagination des sites et encore même créer des requêtes SQL.



Figure 17: Logo Laravel

d) ReactJS

C'est un framework qui a été développé par Facebook en 2013 et est basé sur le langage JSX et intègre la notion de composants. Il est aujourd'hui utilisé par certaines structures comme Instagram, Netflix, etc. Il est utilisé principalement pour le développement front-end c'est-à-dire pour la création d'interfaces utilisateurs.



Figure 18: Logo ReactJS

e) Angular

Créé par Google en 2009, c'est framework JavaScript open-source qui s'occupe de l'animation, la manipulation du DOM, le routage et le data binding. Il dispose d'une grande communauté active qui ne cesse de contribuer pour l'améliorer au fur et à mesure.



Figure 19: Logo Angular

f) Ionic

C'est une technologie open-source créée en 2013 par Max Lynch, Ben Sperry, et Adam Bradley, et est basée initialement sur AngularJS et Apache Cordova. Il permet de créer des codes multisupports en utilisant des outils web pour générer des applications iOS, Android et bien d'autres.



Figure 20: Logo Ionic

g) Flutter

Tout comme Ionic c'est un framework de développement mobile multiplateformes. Il a été créé par Google et écrit en langage Dart qui est aussi un langage de développement Web créé par Google également. Il est utilisé pour créer des applications iOS, Android, Windows, etc.



Figure 21: Logo Flutter

4. Les avantages des frameworks

Il y a beaucoup d'avantages qui se dégagent en ce qui concerne l'utilisation d'un framework. D'une part leur avènement ont en quelque sorte booster de façon considérable le monde du développement informatique. Ces avantages sont :

- **Le gain considérable en temps et en efficacité** : c'est-à-dire qu'il est question ici de partir sur une base structurelle préétablie offrant de nombreux outils qui permettent de réinventer l'exécution de toutes les fonctionnalités,
- **La meilleure organisation** : grâce à la séparation et la classification de chaque fichier, ce qui permet d'avoir une meilleure visibilité sur l'interface,
- **Le bon système de sécurité** : chaque framework offre la possibilité de configurer la partie sécurité.

Mais il s'avère que l'utilisation des frameworks présente bien des inconvénients qui ne passe pas inaperçu.

5. Les inconvénients

Bien que les frameworks ont prouvé jusqu'ici leurs efficacités, ils présentent tout de même des insuffisances dont il faut tenir compte. Ses inconvénients sont :

- **Les limites sur les fonctionnalités complexes** : en d'autres termes l'accès à certaines fonctionnalités du langage qui ne sont pas gérées par le framework est limité,
- **La couche de complexité supplémentaire** : en fonction du framework utilisé et de l'architecture appliqué, entrer sur la technologie peut s'avérer assez complexe pour toute personne ayant un niveau de connaissance moyen, il faut donc trouver une meilleure approche pour mieux appréhender l'utilisation d'un framework,
- **De plus un framework augmente l'incertitude** : sur la capacité à maintenir un projet dans le temps puisque le code sera en grande partie dépendant de la solvabilité de ce framework.

II.Cas du framework React Native

Actuellement le marché du mobile est en plein essor et cela depuis quelques années déjà. De ce fait fort est de constater une utilisation forte des téléphones portables et des tablettes que des ordinateurs. Qui dit téléphones portables et tablettes dit forcément application mobile, et de nos jours il existe toute une panoplie de solutions permettant de développer des applications mobiles.

Les systèmes d'exploitation les plus connus pour le mobile sont Android et iOS, et il existe entre autres deux types d'applications mobiles : les applications natives et les applications cross-platforms. Les applications natives sont des applications développées pour un unique système d'exploitation tandis que les applications cross-platforms sont développées une seule et sont compatibles sur Android et iOS. Ce type de développement d'application se fait par des frameworks tels que : Ionic, Xamarin, etc.

Au vu des exigences que pose le cahier de charge, notre choix s'est donc porté sur le framework React Native, qui est une solution de développement d'application mobile multiplateforme. De plus, l'ayant abordé en classe lors du cours de Web mobile multiplateformes, sa prise en main s'annonçait plus facile pour nous.

1. Présentation de React Native

React Native est un framework de développement d'applications cross-platforms créé par Facebook. Il est essentiellement basé sur le langage JavaScript lui permettant d'utiliser React qui est une bibliothèque de ce langage également créé par Facebook dont le but principal est de faciliter la création d'application web monopage. Une première version de React Native fut publiée en 2015.

Le principe de fonctionnement de ce framework est presque similaire à celui de React, la différence est que React Native ne manipule pas le DOM. Il s'exécute donc par un processus en arrière-plan et interprète ensuite le code JavaScript ; il communique avec la plate-forme native par une passerelle de sérialisation non synchronisé.



Figure 22: Logo React Native

2. Particularités de React Native

Alors que d'autres frameworks cross-platforms utilisent des composants web, React Native utilise des composants mobiles natifs. Lorsqu'on définit des éléments dans le code en React Native, on remarque dans les applications iOS et Android que se sont les équivalents des composants mobiles natifs propre à chaque système d'exploitation et c'est le même fonctionnement pour tous les types d'éléments de React Native ; il convertit tous les éléments en leur équivalent natif. Cette particularité

rend ainsi les applications plus performantes, plus fluides.

D'autre part React Native s'est aussi démarqué grâce à certains points qui sont :

- Le fait qu'il est gratuit, depuis le développement jusqu'au déploiement des applications sur les stores.
- L'aspect Open Source, par lequel une grande communauté s'est formée autour du framework ce qui a permis son évolution et son amélioration.
- La possibilité de tester son application instantanément
- La flexibilité de React Native, qui permet son utilisation dans un projet natif qui existe déjà.

On ne peut pas parler d'une application sans parler du système de gestion des données de l'application.

III. Gestion des données de l'application

React Native propose des options de gestion de données à l'aide desquelles il est possible de traiter, stocker vos données localement ou dynamiquement. Pour mener à bien la bonne gestion des données de notre application notre choix s'est porté sur Firebase qui est un ensemble de services d'hébergements.

Pour la gestion des données de notre application nous utiliserons le service Realtime Database qui est une solution qui est la base de données en temps réel de Firebase

1. Présentation de Firebase

Firebase est une plateforme mobile créée par Google qui permet de développer rapidement des applications. Il propose un ensemble de services comme l'hébergement en NoSQL et en temps réel des bases de données, des services d'authentification sociale ou encore des services d'un serveur de communication en temps réel.



Figure 23: Logo Firebase

Certaines applications parmi lesquelles on retrouve les services fournis par Firebase sont :

- Shazam qui est un logiciel de reconnaissance musicale de chansons

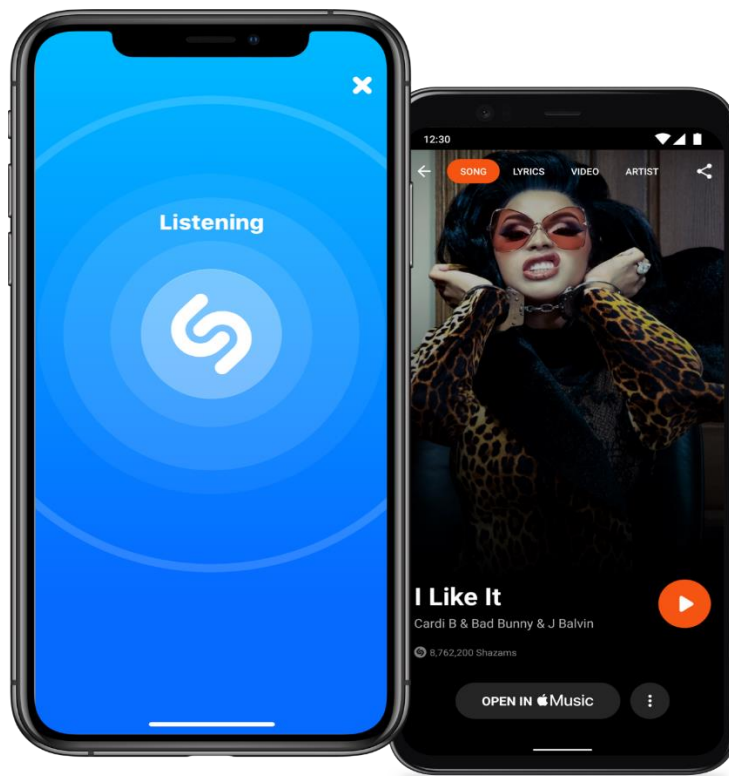


Figure 24: Application Shazam

- Le Figaro un journal quotidien français



Figure 25: Application Le Figaro – capture d'écran 1



Figure 26: Application Le Figaro – capture d'écran 2

- Skyscanner qui est un site web et aussi une application mobile proposant un moteur de comparaison de vols, hôtels, et localisations de voitures.

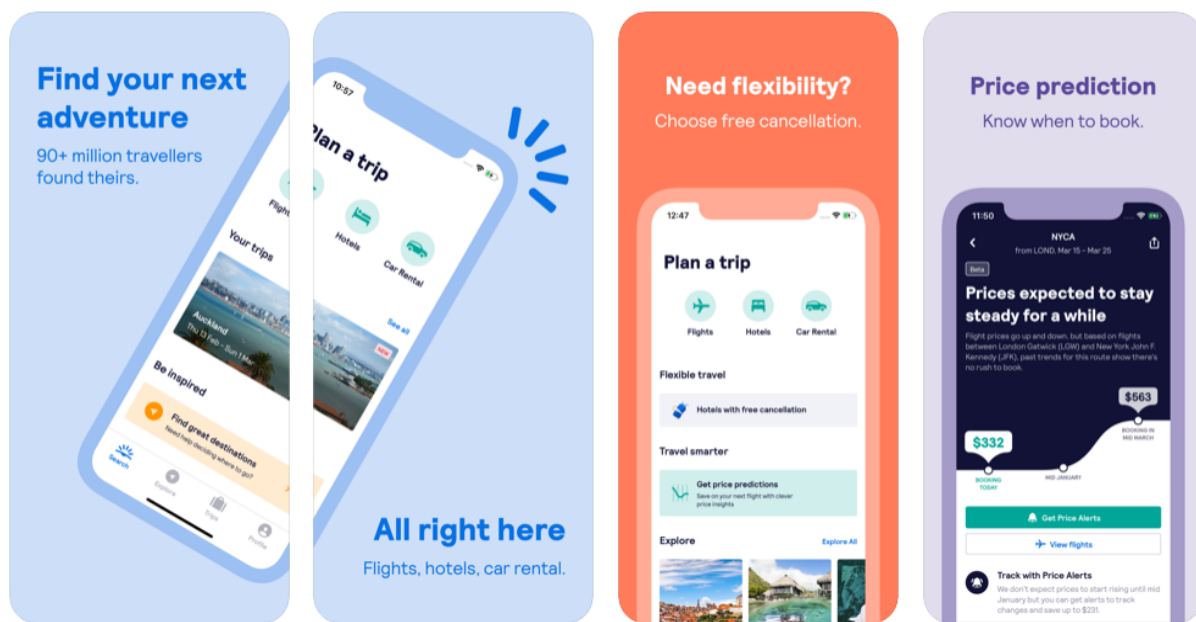


Figure 27: Application Skyscanner

2. Le service Firebase Realtime Database

Comme énoncé plus haut, Realtime Database est un service de base de données en temps réel tout comme son nom l'indique. Il permet de stocker et synchroniser les données. Il fournit un moyen sûr, fiable et rapide pour la synchronisation des données ; il garantit également la persistance des données en stockant les données localement.

Il utilise une base de données NoSQL pour le stockage des données ; elles sont stockées sous la forme d'un objet JSON qui se compose généralement d'une paire de clé/valeur. La clé identifie de façon unique l'objet dans la base de données et la valeur représente les données stockées.

Au vu du cahier de charge qui a été présenté plus haut, nous avons porté notre choix sur ce service car sa particularité est sa capacité de gérer les données en temps réels. Ce service vient donc répondre à certains points du cahier de charge.



Figure 28: Logo Firebase Realtime Database

REALISATION DE LA PLATEFORME



I. Mise en place de l'environnement de développement

Pour débuter la conception de l'application, on doit dans un premier temps mettre en place l'environnement de travail par lequel on pourra développer de bout en bout notre application. Elle consiste à créer un nouveau projet React Native dans lequel se trouve des fichiers de base qui sont indispensables pour démarrer tout projet.

1. Création et structuration de base du projet React Native

Ce dossier comporte différents fichiers et dossiers essentiels pour démarrer notre application, parmi eux, le fichier qui sert de point d'entrée de l'application, le fichier ou les configurations de l'application sont définis et qui sera modifié au fur et à mesure que l'application sera développée, le dossier contenant les modules dont l'application a besoin pour son bon fonctionnement.

2. Ajout de fichiers complémentaires au projet

Il est question ici d'ajouter des fichiers manuellement qui seront utilisés. Même si le dossier de base contient déjà des fichiers de base il est possible d'en rajouter au besoin sans pour autant modifier ceux de base.

II. Présentation des interfaces graphiques

L'aspect technique des fichiers que contient le dossier présenté plus haut sera abordé dans cette partie. On abordera donc le rôle de chaque composant créé et le traitement des données qui s'y effectue.

Certains composants sont communs aux utilisateurs c'est-à-dire les clients et l'administrateur de l'application

a) Les composants splashscreen et presentation

Ces composants sont les premiers qu'on renvoie à l'interface graphique qui affiche une description globale de l'application et ce pour quoi elle est mise en place.

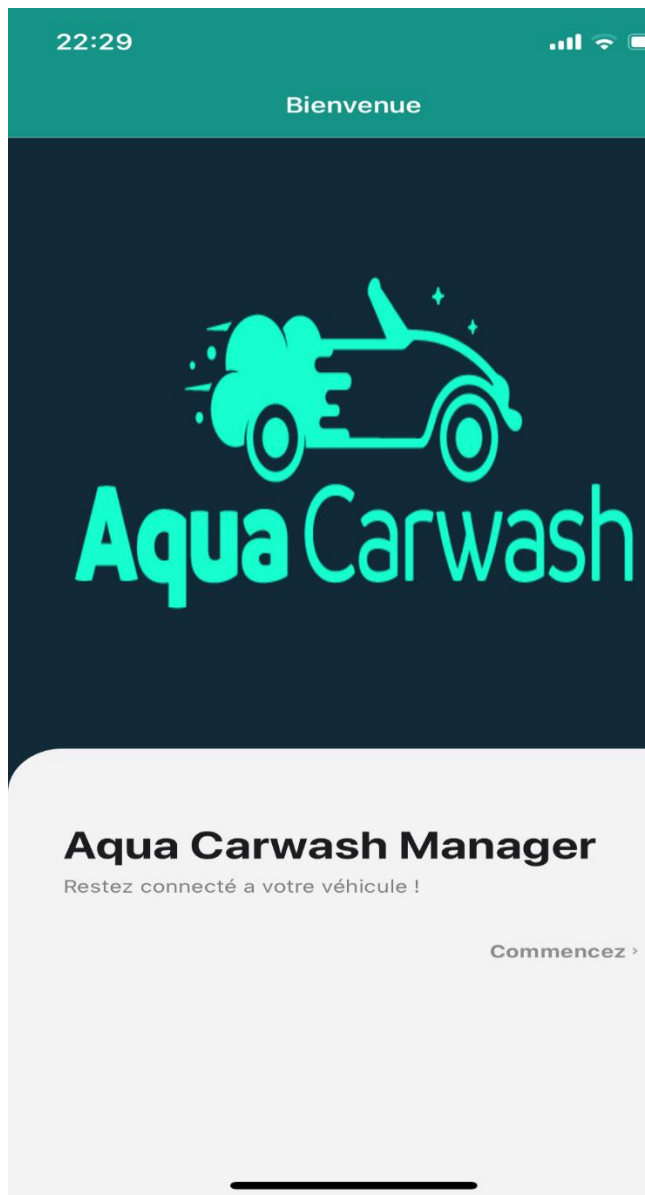


Figure 29: Capture d'écran - SplashScreen



Figure 30: Capture d'écran - Presentation

b) Le composant SignInScreen

A ce niveau, l'utilisateur doit saisir ses données et en fonction de s'il est un client ou s'il est l'administrateur il lui sera renvoyé la vue appropriée. Les traitements des données se font par des fonctions définies dans le composant.

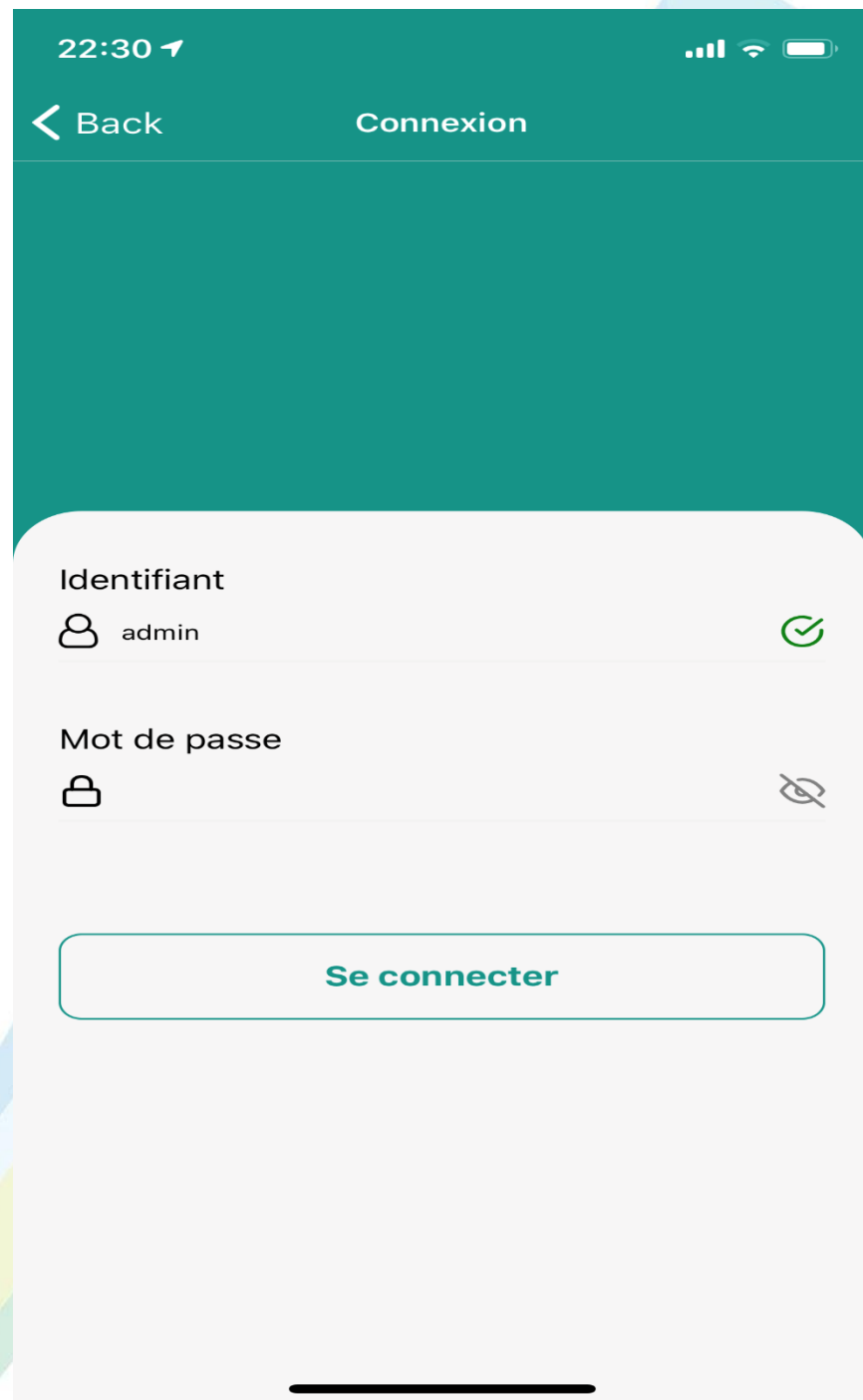


Figure 31: Capture d'écran – SignInScreen

c) **Le composant StatusCheck**

C'est l'un des composants renvoyé lorsqu'un utilisateur se connecte en tant que client. C'est par ce composant que l'utilisateur peut suivre en temps réel l'état d'avancement. Il doit saisir le numéro de sa plaque d'immatriculation et ensuite sera redirigé sur une autre page où il pourra donc visualiser l'état d'avancement du service.

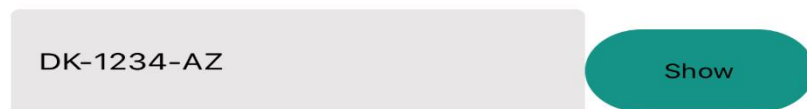


Figure 32: Capture d'écran - StatusCheck

d) Le composant Status

Dans ce composant qui renvoie l'état d'avancement du service en cours, tous les éléments sont définis pour s'assurer que le composant renvoie bien la vue adéquate. Cette page est renvoyé du côté client, Après avoir saisi ses références de plaque d'immatriculation il est redirigé vers cette page ou il peut visualiser les informations.

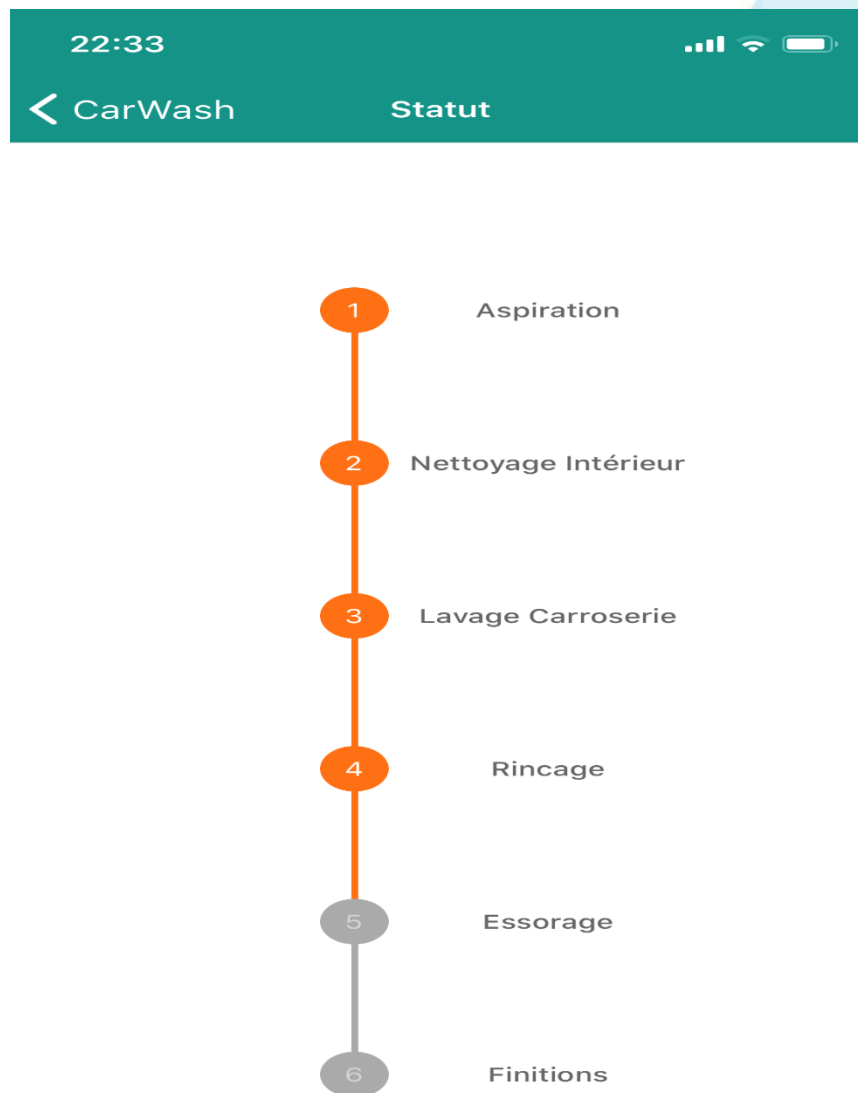


Figure 33: Capture d'écran - Status

e) Le composant Search

Sur cette page l'utilisateur peut visualiser les différents points de lavages automobiles. Il peut aussi voir les adresses de ces points de lavages.

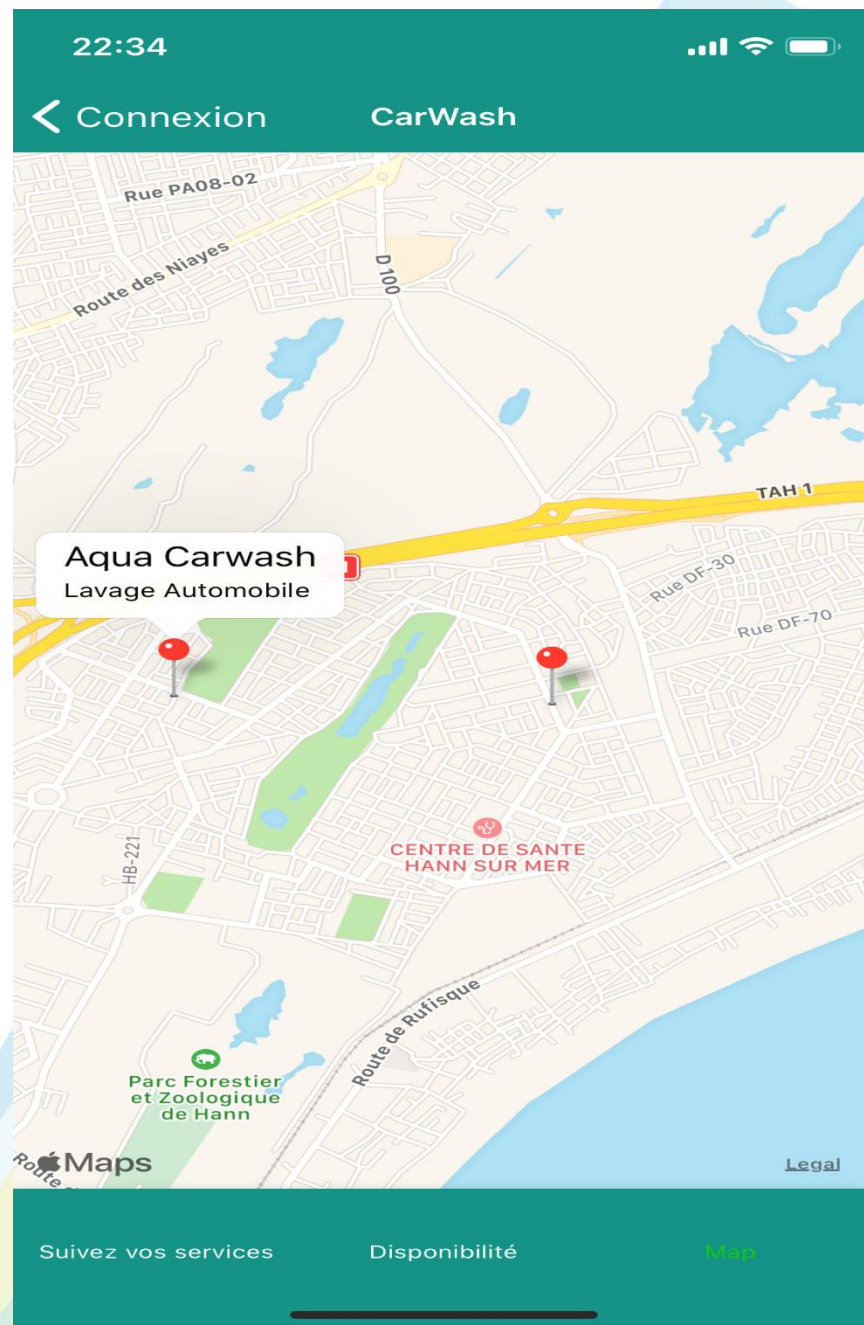


Figure 34: Capture d'écran - Search

f) **Le composant PlaceCheck**

Cette page est renvoyée au client ou est affichée les informations concernant la disponibilité des places de parkings au niveau des points de lavages.

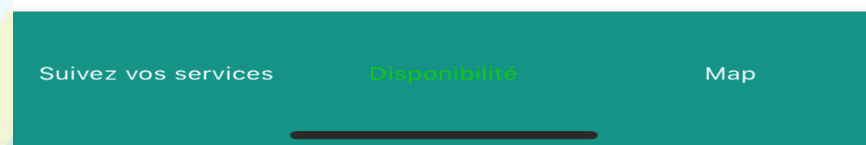
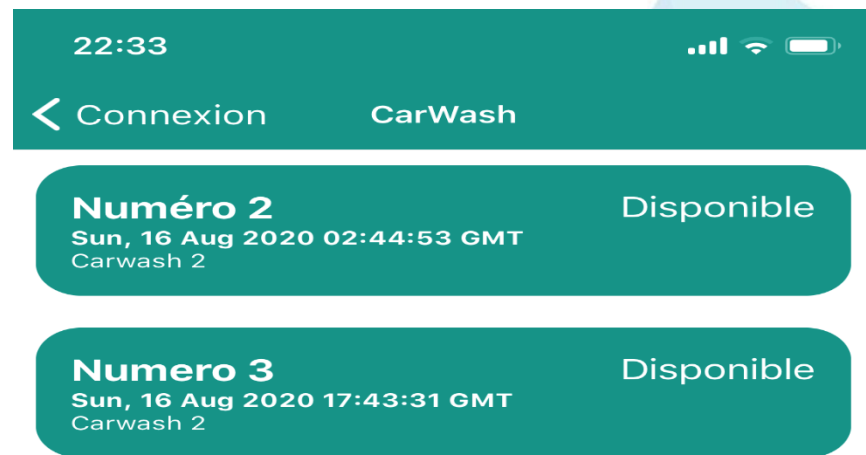


Figure 35: Capture d'écran - PlaceCheck

g) Le composant PlaceSet

Ici l'administrateur visualise les places disponibles sur le parking, ensuite ses informations seront renvoyées au composant PlaceCheck du côté du client ou il pour les visualiser. Il peut ainsi donc supprimer une place en fonction de sa disponibilité.

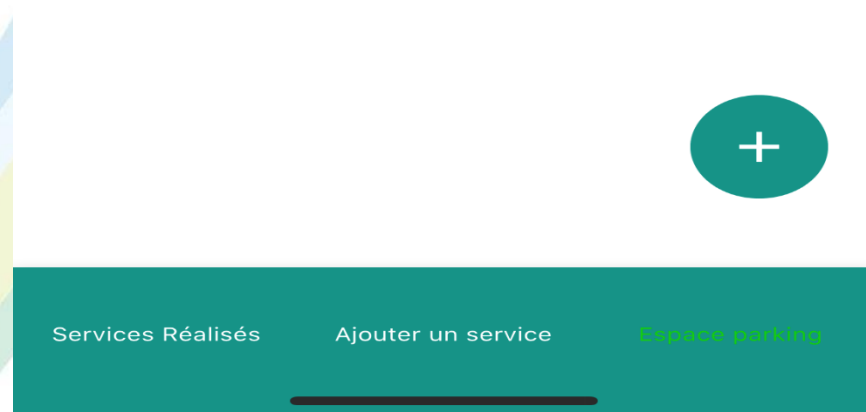
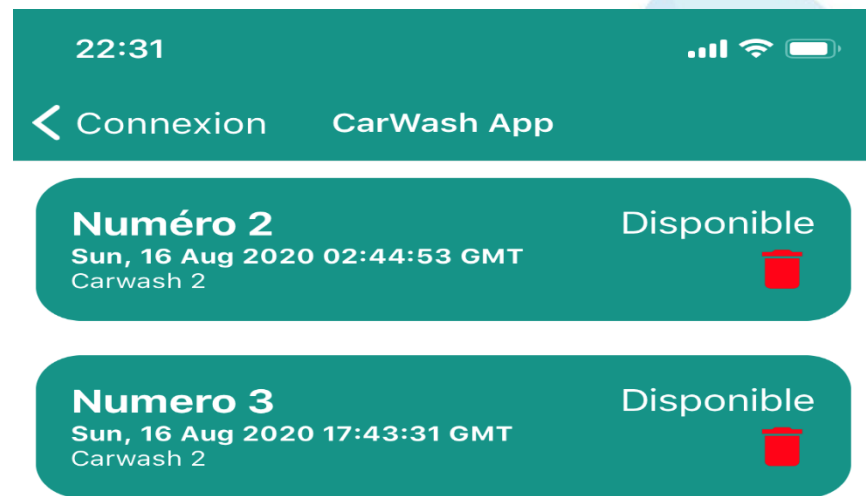


Figure 36: Capture d'écran - PlaceSet

h) Le composant AddPlace

Sur ce composant l'administrateur ajoute les places disponibles sur le parking qui seront renvoyés du côté client au niveau du composant PlaceCheck et du côté de l'administrateur au niveau du composant PlaceSet

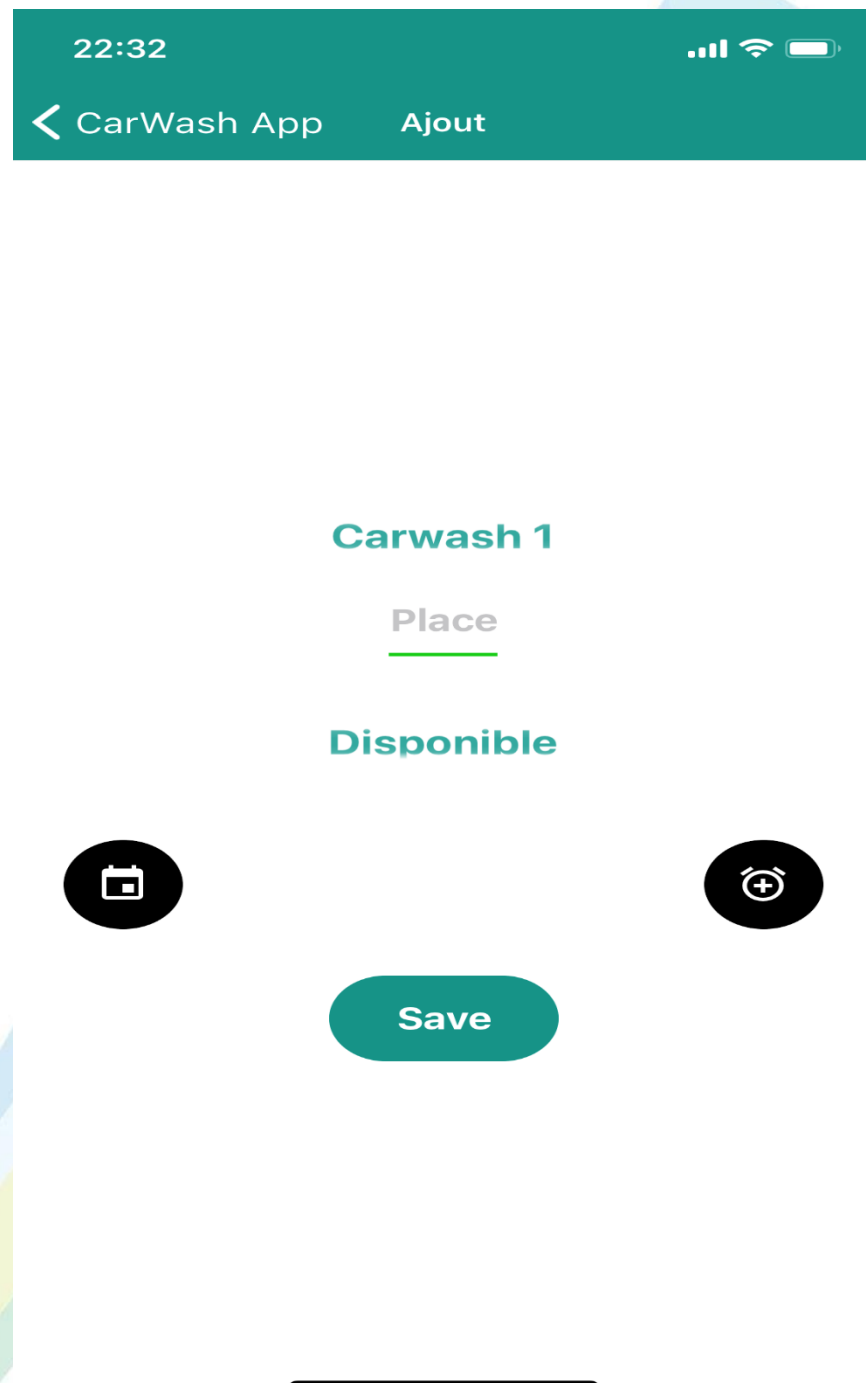


Figure 37: Capture d'écran - AddPlace

i) Le composant AddWash

Les éléments qui permettent de retourner la vue où il est possible d'ajouter le service demandé par un client afin qu'il puisse suivre l'avancement en temps réel. L'administrateur se charge d'ajouter les services en cours en renseignant les informations nécessaires qui seront mises à jour plus tard au fur et à mesure de l'avancement.

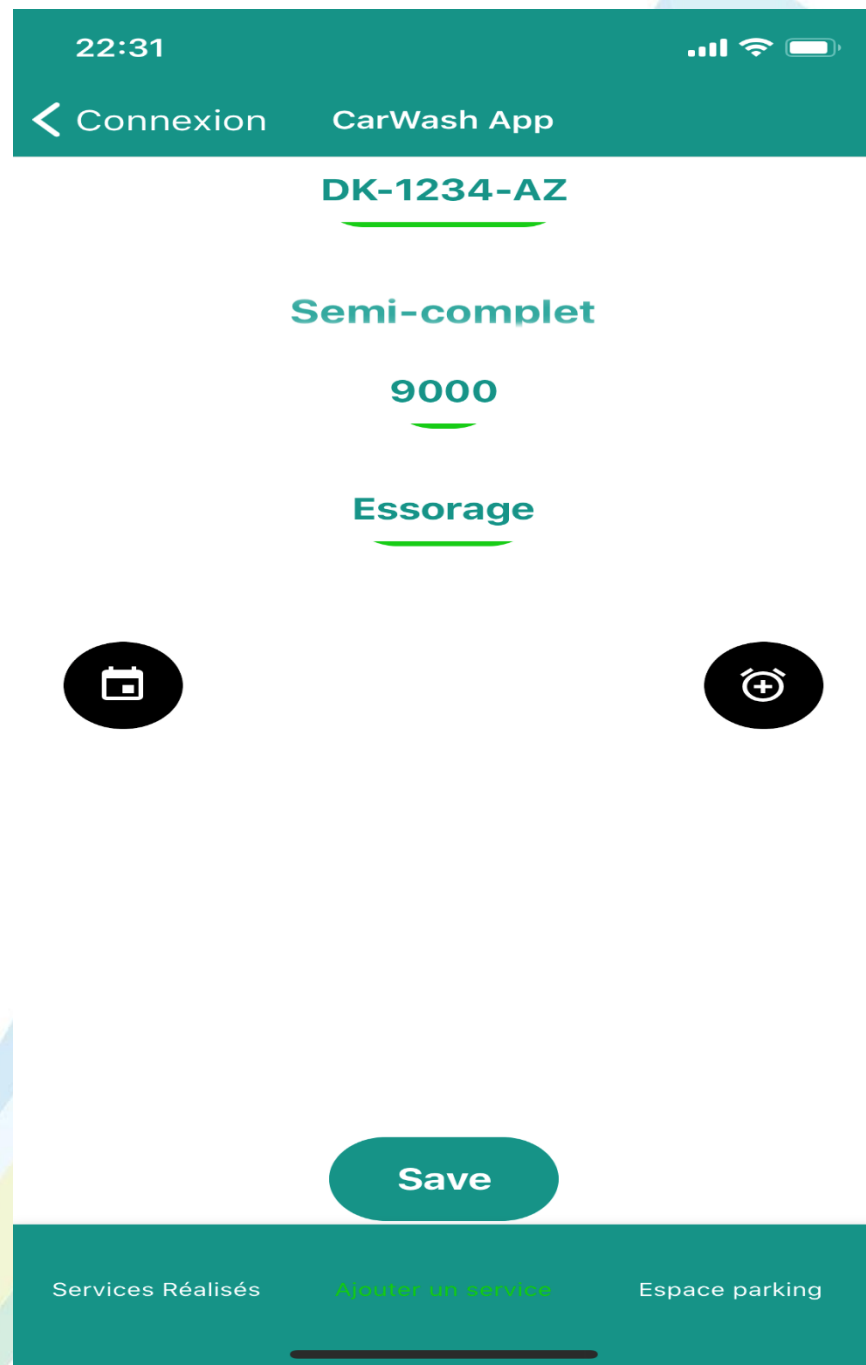


Figure 38: Capture d'écran - AddWash

j) **Le composant ListWash**

Ce composant est l'un des composants affichés lorsque l'utilisateur se connecte en tant qu'administrateur. Ici, sont définis les éléments qui renvoient à l'administrateur la liste des services en cours. En fonction de l'avancement il peut donc mettre à jour les informations pour que le client puisse suivre en temps réel de son côté l'état.

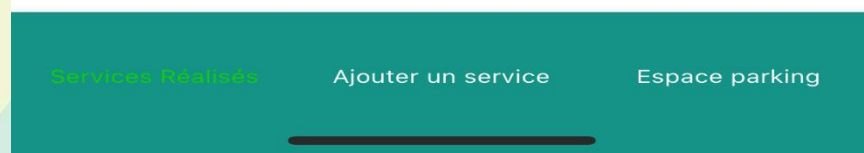
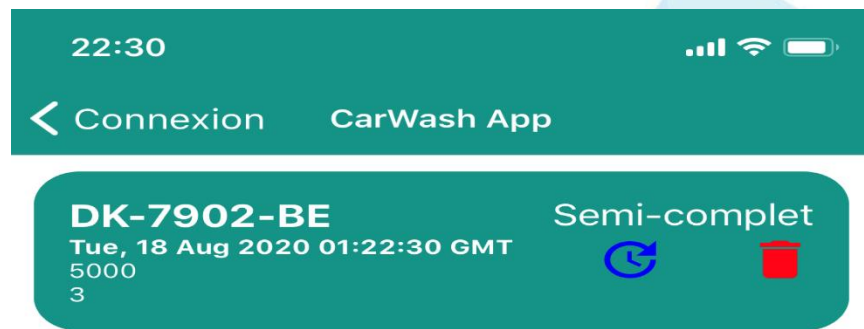
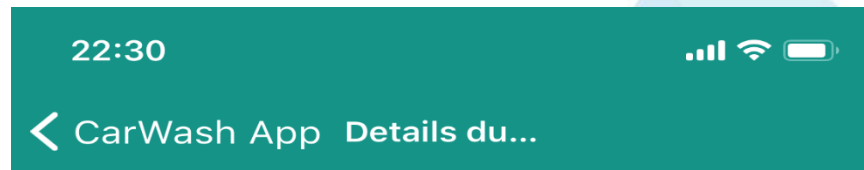


Figure 39: Capture d'écran - ListWash

k) Le composant WashDetail

Lorsque l'administrateur souhaite mettre à jour les informations concernant l'avancement d'un service en cours, il sera redirigé sur cette page où il pourra donc modifier et mettre à jour.



Aspiration

Nettoyage Inté...

Lavage Carros...

Rincage

Save

Figure 40: Capture d'écran - WashDetail

CONCLUSION

Au cours de cette présente étude, nous avons pu aller explorer le secteur du lavage automobile et avons pu dégager certaines contraintes qui sont essentiellement le souci de gestion des espaces de lavage, le suivi sur l'avancement des services demandés par les clients et permettre de visualiser sur une carte de géolocalisation les points de lavages automobiles.

Afin de résoudre les problèmes évoqués, nous avons étudié le domaine sous plusieurs aspects en passant par l'analyse conceptuelle du problème, et l'implémentation d'une application qui pour se faire a nécessité des recherches approfondies sur les frameworks, plus précisément React Native, la technologie qui selon nous était la plus appropriée.

Grace à toutes ces recherches pour la réalisation de ce projet, nous avons pu développer certaines autres aptitudes en tant que développeur d'application et avons pu améliorer nos connaissances sur les frameworks et les bases de données.

Toutefois, la solution que nous avons proposée peut faire l'objet d'améliorations, toujours dans l'optique de faciliter la gestion, d'améliorer la qualité des services pour les clients et de prévoir de nouvelles fonctionnalités comme l'implémentation d'une fonctionnalité de prise de rendez-vous qui permettrait à tout client de prendre rendez-vous pour le lavage de sa voiture ou tout autre service disponible via l'application ; mais aussi la possibilité de notifier en fin de service.

WEBOGRAPHIE

I – Framework

1- Avènement du premier framework

https://fr.wikipedia.org/wiki/IBM_Lotus_Symphony. Consulté le 07 Août 2020

2- Présentation générale des framework et présentation de React Native

[https://fr.wikipedia.org/wiki/Framework_\(logiciel\)](https://fr.wikipedia.org/wiki/Framework_(logiciel)). Consulté le 07 Août 2020

<https://www.1min30.com/dictionnaire-du-web/framework>. Consulté le 07 Août 2020

<https://fr.wikipedia.org/wiki/Framework>. Consulté le 07 Août 2020

https://blog.axopen.com/2018/06/framework-utilisation-pour-contre/#framework_avantage. Consulté le 09 Août 2020

II - React Native

Utilisation du framework React Native

<https://openclassrooms.com/fr/courses/4902061-developpez-une-application-mobile-react-native/4902068-decouvrez-le-developpement-mobile-actuel>. Consulté le 11 Août 2020

<https://reactnative.dev/>. Consulté le 14 Août 2020

III – Firebase

Présentation et utilisation de firebase

<https://fr.wikipedia.org/wiki/Firebase>. Consulté le 14 Août 2020

<https://firebase.google.com/products/realtime-database?hl=fr>. Consulté le 14 Août 2020

https://www.techotopia.com/index.php/Firebase_Realtime_Database#An_Overview_of_the_Firebase_Realtime_Database. Consulté le 14 Août 2020

IV – Skyscanner

Présentation de l'application Skyscanner

<https://apps.apple.com/us/app/skyscanner-travel-deals/id415458524>. Consulté le 14 Août 2020

V - Le Figaro

Présentation de l'application Le Figaro

<https://apps.apple.com/fr/app/le-figaro-actualit%C3%A9s-et-info/id319557427>. Consulté le 14 Août 2020

VI – Shazam

Présentation de l'application Shazam

<https://www.shazam.com/fr>. Consulté le 14 Août 2020

ANNEXE



I.ANNEXE 1

Autres Applications de lavage automobile

- App TOTAL WASH

Une application mobile qui permet de commander son lavage automobile depuis son smartphone ou tablette. Elle offre la possibilité de trouver sur une carte de géolocalisation le centre de lavage automobile le plus proche et d'acheter directement son lavage. Ceci pour mettre fin à l'attente à la caisse.

- WashMEE

C'est une application qui référence les types de lavages automobiles. Il offre la possibilité d'avoir l'adresse des points de lavages sur une carte de géolocalisation, la possibilité d'acheter un lavage, etc.

II. ANNEXE 2

Quelques applications mobiles développées avec le framework React Native

- Instagram

C'est une application, un réseau social et un service de partage de photos et de vidéos fondés.

- Facebook

Tout comme Instagram, c'est aussi un réseau social.

- Uber Eats

L'application de service de livraison de repas à domicile de Uber

Il existe toute une panoplie d'applications mobiles développées à partir de React Native.

<https://blog.bam.tech/business-news/react-native-exemple>

III.ANNEXE 3

Les limites du framework React Native

En comparaison avec d'autres framework de développement d'application mobiles, ceux développées avec React Native suscitent une attention particulière. L'un des premiers inconvénients est qu'il ne supporte pas toutes les API natives au monde. En effet, toutes les solutions qui s'offrent à vous ne sont pas toutes disponibles et accessibles via ce framework. En fin les mises à jour ne sont pas automatiques, ce qui peut avoir comme conséquences un fonctionnement fragilisé.

Des ressources humaines limitées

Les ressources humaines compétentes en ReactNative sont un point positif mais aussi légèrement négatif. Tous les développeurs ne sont pas spécialement formés pour créer des applications mobiles via ReactNative. La recherche de développeurs sera donc plus complexe que si vous développez votre application mobile de manière active.

Des connaissances approfondies nécessaires

Développer son application mobile sur ReactNative demande de très fortes compétences en développement. La documentation officielle reste plutôt légère, il est donc crucial d'avoir le bon développeur sous le coude
De plus, le langage Javascript est un langage de programmation qui ne garantit pas les erreurs possibles de programmation.

TABLE DES MATIERES

DEDICACE	2
REMERCIEMENTS.....	3
TABLE DES FIGURES	4
GLOSSAIRE.....	5
AVANT-PROPOS	6
SOMMAIRE.....	7
INTRODUCTION	8
PRESENTATION GENERALE.....	9
I. Les espaces de lavages de automobiles à Dakar.....	10
II. Présentation du sujet.....	10
1. Problématique.....	10
2. Solution proposé.....	11
III. Cahier de charge	11
ANALYSE ET CONCEPTION.....	12
I. Diagramme de cas d'utilisation :	13
II. Diagramme de communication :	14
1. Diagramme de communication d'authentification :.....	14
2. Diagramme de communication d'Ajout	15
3. Diagramme de communication de Modification.....	16
4. Diagramme de communication de suppression	17
5. Diagramme de communication de Recherche.....	17
6. Diagramme de communication de Notification	18
7. Diagramme de communication de Consultation	18
III. Diagramme de séquences :	19
1. Diagramme de séquence du cas d'utilisation "authentification"	19
2. Diagramme de séquence du cas d'utilisation "Ajout"	20
3. Diagramme de séquence du cas d'utilisation "Modification"	21
4. Diagramme de séquence du cas d'utilisation "Consultation"	22
5. Diagramme de séquence du cas d'utilisation "Recherche"	23
IV. Diagramme de classe	24
CHOIX DES TECHNOLOGIES	26
I. Généralité sur les frameworks	27
1. Introduction générale aux frameworks	27
2. Les différents types de frameworks	27
3. Exemples de frameworks	27

a)	Symfony.....	28
b)	Django	28
c)	Laravel	28
d)	ReactJS	28
e)	Angular	28
f)	Ionic	29
g)	Flutter	29
4.	Les avantages des frameworks.....	29
5.	Les inconvénients	29
II.	Cas du framework React Native.....	30
1.	Présentation de React Native	30
2.	Particularités de React Native.....	30
III.	Gestion des données de l'application	31
1.	Présentation de Firebase	31
2.	Le service Firebase Realtime Database	33
	REALISATION DE LA PLATEFORME	34
I.	Mise en place de l'environnement de développement	35
1.	Création et structuration de base du projet React Native.....	35
2.	Ajout de fichiers complémentaires au projet.....	35
II.	Présentation des interfaces graphiques	35
a)	Les composants splashscreen et presentation.....	35
b)	Le composant SignInScreen	36
c)	Le composant StatusCheck	37
d)	Le composant Status.....	38
e)	Le composant Search	39
f)	Le composant PlaceCheck.....	40
g)	Le composant PlaceSet	41
h)	Le composant AddPlace	42
i)	Le composant AddWash	43
j)	Le composant ListWash	44
k)	Le composant WashDetail.....	45
	CONCLUSION.....	47
	WEBOGRAPHIE	48
	ANNEXE	49
I.	ANNEXE 1	50

II.	ANNEXE 2.....	51
III.	ANNEXE 3.....	52



MÉMOIRE FIN DE FORMATION POUR L'OBTENTION DU DIPLÔME DE LICENCE PROFESSIONNELLE EN TIC

 **OPTION : DAR (DEVELOPPEMENT D'APPLICATIONS)**

Abibou Falilou NGOM & Assima Geovani TAGBA

THEME :

Mise en place d'une plateforme de gestion de lavage automobile

DIRECTEUR DE MEMOIRE : M. Jean-Marie Preira

Il est nécessaire pour toute entreprise ou structure sans distinction de domaine d'activité de veiller à ce que leurs clients aient le nécessaire pour faire leurs opérations, de savoir l'ensemble des actions qu'ils peuvent faire et aussi de répondre à leur besoins ou questions à temps réel. C'est donc dans cette optique que nous avons décidé de mettre en place une plateforme mobile permettant de gérer les structures de lavage automobiles. Cela va permettre aux utilisateurs qui sont aussi les clients de pouvoir suivre l'avancement du service demandé, de visualiser sur une carte de géolocalisation les structures de lavages si besoin, d'avoir également des informations sur la disponibilité de place dans les parkings de ses entités de lavages automobiles ; et tout ceci en temps réel.

Il s'agit de la conception d'une application mobile, elle fonctionne comme une application de service ou d'assistance en ligne en temps réel. Les technologies utilisées pour la conception permettent d'avoir une application qui sera compatible avec les systèmes d'exploitation mobiles iOS et Android ; elles garantissent un certain niveau de sécurité des données qui seront manipulées par les utilisateurs. Les fonctionnalités de l'application pourront être au fur et à mesure améliorées en fonction des besoins des utilisateurs pour permettre une évolutivité dans le temps de l'application.

Après la conception de l'application, la structure de lavage automobile pourra s'en servir comme espace d'assistance client et d'espace d'échanges d'informations (Voir l'avancement du service demandé, la disponibilité des places de parking...) et s'en servir pour d'autres fins qui seront définies dans le temps en fonction des demandes des utilisateurs.