

République Algérienne Démocratique et Populaire

Université Abou Bakr Belkaid– Tlemcen

Faculté des Sciences

Département d'Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Master en Informatique

Option : Réseaux et Systèmes Distribués (R.S.D)

Thème

**Réingénierie d'un système de stationnement
intelligent à critères multiples en utilisant la
découverte des services IoT**

Réalisé par :

- M^r. SALMI Mohammed Nadjib
- M^r. LATIGUI Fayssal

Présenté le 02 Juillet 2023 devant le jury composé de MM.

- ILES Nawel..... (Présidente)
- MERZOUG Mohamed..... (Encadreur)
- ETCHIALI Abdelhak..... (Co- Encadreur)
- BENAHMED Hamed..... (Co- Encadreur)
- ABDELDJELIL Hanane..... (Examinatrice)
- TADLAOUI Mohamed (Examineur)

Année Universitaire : 2022-2023

Dédicace

À nos parents qui ont été nos piliers et nos guides tout au long de ce voyage académique, cette dédicace est un témoignage de notre gratitude éternelle. Votre soutien inconditionnel, vos encouragements sans faille et vos sacrifices inestimables ont été la force motrice derrière notre persévérance et notre réussite. Ce mémoire est le reflet de l'amour et de l'éducation que vous nous avez prodigués, et il vous est dédié avec tout notre amour.

À nos sœurs et frères, qui ont partagé avec nous des moments de joie, de complicité et de soutien, cette dédicace est une expression de notre reconnaissance pour votre présence constante dans nos vies. Vos encouragements, vos conseils avisés et votre amour inconditionnel ont été des sources d'inspiration inestimables tout au long de ce parcours académique.

À mes frères

Vous êtes une équipe de choc, pour la vie.

Remerciements

Avant tout, nous exprimons notre profonde gratitude à **ALLAH exalté soit-il**, le très miséricordieux, pour son aide et sa générosité qui ont été cruciales tout au long de la préparation de ce mémoire.

Nous adressons aussi nos remerciements les plus sincères à nos encadreurs, **M^{rs}. MERZOUG Mohamed, BENAHMED Hamed** et **ETCHIALI Abdelhak** pour leurs conseils avisés et leurs discussions constructives qu'ils nous ont prodigués et ont grandement contribué à la réussite de ce travail.

Nous tenons également à exprimer notre reconnaissance à **M^r. HADJILA Fethallah** pour son assistance précieuse.

Nous exprimons aussi notre grande reconnaissance envers les membres du jury pour leur disponibilité, leur précieux temps consacré et leur engagement lors de l'évaluation de notre mémoire.

Nous saisissons cette occasion pour témoigner notre profonde gratitude au **P^r. RAHMOUN Abdellatif** pour son soutien inestimable, tant sur le plan matériel, scientifique que moral.

Enfin, nous souhaitons exprimer notre gratitude infinie envers nos familles pour leur amour indéfectible, leur aide inestimable et leur soutien constant. Nous remercions particulièrement nos parents pour leur soutien inconditionnel et leurs encouragements incessants tout au long de ce parcours.

ملخص:

نستكشف في هذه المذكرة الدور الأساسي لإنترنت الأشياء في ربط العناصر. لقد قمنا بتطوير نموذج أولي ذكي لمواقف السيارات باستخدام خوارزمية متعددة المعايير، بهدف مساعدة المستخدمين في العثور على أفضل مكان لوقوف السيارات بناءً على معايير مثل التوفر والسعر والسمعة. تعتمد منهجيتنا المبتكرة على اكتشاف خدمات إنترنت الأشياء من خلال تطبيق جوال متزامن (أندرويد). يساهم هذا العمل في معالجة قضايا ووقوف السيارات من خلال إنترنت الأشياء ويمهد الطريق للتطورات المستقبلية في هذا المجال.

Résumé :

Ce mémoire explore le rôle essentiel de l'Internet des Objets (IoT) dans la connectivité des éléments. Nous avons développé un prototype de parking intelligent en utilisant un algorithme multicritère, visant à aider les utilisateurs à trouver le parking le plus pertinent en fonction de critères tels que la disponibilité, le prix et la réputation. Notre approche novatrice repose sur la découverte des services IoT via une application mobile (Android) en temps réel. Ce travail contribue à résoudre les problèmes de stationnement grâce à l'Internet des Objets et ouvre la voie à de futures améliorations dans ce domaine.

Abstract:

This thesis explores the essential role of the Internet of Things (IoT) in connecting the elements. We have developed a smart parking prototype using a multicriteria algorithm, aiming to assist users in finding the best parking spot based on criteria such as availability, price, and reputation. Our innovative approach relies on IoT service discovery through a real-time mobile application (Android). This work contributes to addressing parking issues through the Internet of Things and paves the way for future advancements in this field.

Table des matières

➤ Introduction générale	1
➤ 1- Contexte :	1
➤ 2- Motivation :	1
➤ I. Internet des objets (Internet of Things)	4
○ I.1 Historique	4
○ I.2 Définition	4
II. Composantes de l'IoT	5
○ II.1 Les capteurs	5
○ II.2 Les réseaux de capteurs	5
○ II.3 L'énergie.....	6
○ II.4 Les actionneurs.....	6
○ II.5 La connectivité.....	6
○ II.6 Traitement de données.....	6
○ II.7 Interface utilisateur.....	7
➤ III. Domaines d'application de l'Internet des objets	7
➤ IV. La découverte de services IoT	10
○ IV.1 Différences entre les services IoT et les services Web Traditionnels	10
○ IV.2 Défis et exigences de la découverte des services IoT	11
➤ V. Conclusion.....	12
➤ I. Introduction	14
➤ II. Stationnement traditionnel	14
➤ III. Stationnement intelligent.....	15
III.1 Principe de fonctionnement du système de smart parking.....	16

III.2 Les capteurs utilisés dans les systèmes de parking intelligent	17
III.2.1 Capteur infrarouge (IR)	17
III.2.2 Capteur ultrasonique	18
III.2.3 Identification par radiofréquence (RFID)	18
III.2.4 Magnétomètre	19
III.2.5 Caméras	19
➤ IV. Conclusion :	20
➤ I. Introduction :	22
➤ II. Architecture proposée	22
○ II.1 Couche physique et réseau	22
○ II.2 Couche middleware :	23
○ II.3 Couche applicative	24
➤ III. Principe de fonctionnement de notre système.	24
○ III.1 Choix Multicritères	24
○ III.2 Qualité statique et dynamique des Critères utilisés	24
▪ III.2.1 La Disponibilité en tant que Donnée IoT Dynamique	24
▪ III.2.2 La Nature Statique du Prix et de la Réputation	25
➤ IV. Algorithmes utilisés dans les systèmes de stationnement intelligents	25
○ IV.1 Méthode Utilisée	26
IV.2 Normalisation des critères	27
IV.3 Segmentation de notre Algorithme utilisé	29
○ IV.4 Matériel utilisé	30
● Nous avons choisi l'ESP 8266 pour plusieurs raisons	32
● Intégration Wi-Fi	33
● Faible coût	33

●	Taille compacte.....	33
●	Grande communauté de développeurs.....	33
➤	V. Prototype de projet	33
○	V.1 Architecture de prototype.....	33
○	V.2 Description de prototype.....	33
○	V.3 Implémentation de prototype	35
○	V.4 Interconnexion entre les différents modules de système.....	35
➤	VI. Logiciels & Technologies Utilisées	36
➤	VII. Application mobile	38
○	VII.1 Analyse de besoins	38
○	VII.2. Modélisation.....	39
➤	VIII. Scénarios des acteurs de système	47
○	VIII.1. Propriétaire de parc	47
	VIII.2 Conducteur	49
➤	IX. Conclusion	55
	Références	59

Liste des tableaux

Tableau 1 : Services IoT et Services Web traditionnels.	11
Tableau 2 : Exigences et les défis de la découverte de services IoT.....	11
Tableau 3 : Matériel utilisé.	30
Tableau 4 : Logiciels & Technologies Utilisées.	36

Liste des figures

Figure 1: Internet des objets	4
Figure 2: Composantes de l'IoT [10].....	5
Figure 3: Domaines d'application de l'Internet des objets [9].....	7
Figure 4: La domotique intelligente	8
Figure 5 : Santé connectée (HealthCare).....	8
Figure 6: Ville intelligentes	9
Figure 7 : Boucle de découverte dans le monde IoT [30].....	10
Figure 8 : Parking traditionnel.....	15
Figure 9 : Exemple-1 Smart-Parking.....	15
Figure 10 : Exemple-2 Smart-Parking.....	16
Figure 11: Fonctionnement du système de smart parking [15].	16
Figure 12: Capteur PIR.....	17
Figure 13 : Capteur IR actif.....	18
Figure 14 : Capteur ultrasonique.	18
Figure 15: Capteur d'identification par radiofréquence (RFID).....	18
Figure 16 : Capteur sous-sol (Magnétomètre).....	19
Figure 17: L'architecture proposée.	23
Figure 18 : Architecture des maquettes des parcs 1,2 et 3.	33
Figure 19: Schéma général de branchement du système prototype.	34
Figure 20 : Représentation schématique du prototype.	34
Figure 21: Circuit proposé pour les maquettes.....	35
Figure 22: Création des capteurs IR sur Firebase.....	35
Figure 23: Schéma représente la communication entre les différents modules de système....	36
Figure 24: Diagramme de cas d'utilisation.	40

Figure 25 : Diagramme de séquence pour s'authentifier (propriétaire du Park).	41
Figure 26 : Diagramme de séquence pour gérer les informations du parking.	42
Figure 27 : Diagramme de séquence d'authentification (conducteur).....	43
Figure 28 : Diagramme de séquence pour choisir une destination (conducteur).....	44
Figure 29 : Diagramme de séquence choisir une destination (invité).	45
Figure 30 : Diagramme de classes.....	46
Figure 31 : Interface inscription 1.	47
Figure 32 : Interface d'accueil.	47
Figure33 : Interfaced'inscription2.	47
Figure 34 : Interface de connexion (propriétaires de parc).	48
Figure 35 :Interface information.	48
Figure 36 : Inscription d'un nouveau parc.	48
Figure 37 : Interface page d'accueil.....	49
Figure 38 : Interface inscription conducteur.	49
Figure 39 : Interface connexion conducteur.....	49
Figure 40 : Interface de home application.....	49
Figure 41 : Interface de recherche parcs.	49
Figure 42 : Interface de filtre optionnel.....	49
Figure 43 : Interface MapParkB.....	50
Figure 44 : Interface parcs disponibles.....	50
Figure 45 : Interface guidage par Google Maps.....	50
Figure 46 : L'inscription pour un conducteur.	51
Figure 47 : Connexion conducteur.	52
Figure 48 : Recherche de destination.	52
Figure 49 : Filtrage de la recherche (Optionnel).	53
Figure 50 : Sélection d'un parc.	54

Figure 51 : Demande de direction vers le parc choisi.	54
Figure 52 : Redirection vers la liste des parkings disponibles.	55

Liste des abréviations

- **IoT** : Internet of Things.
- **PMS** : Park Management System.
- **SIoT**: Social Internet of Things.
- **PIR**: Passif Infrarouge.
- **Top K** : Top des éléments.
- **API**: Application Programming Interface.
- **Wi-Fi**: Wireless Fidelity.
- **BMC**: Business Model Canvas.

Introduction générale

Introduction générale

➤ **Introduction générale**

➤ **1- Contexte :**

L'apparition de l'Internet des objets a permis d'ouvrir plusieurs nouvelles perspectives dans de nombreux domaines, révolutionnant ainsi la communication entre les objets dit physiques et l'être humain. Cette dernière permet la communication entre des dispositifs intelligents, créant ainsi un réseau interconnecté de "choses" qui peuvent partager des données et agir de manière autonome. Cette nouvelle technologie trouve son application dans divers domaines et a impacté notre société ainsi que son environnement.

Dans le domaine particulier de la gestion des parkings, les moyens traditionnels sont confrontés à des défis tels que la congestion, la recherche de places de stationnement et une mauvaise gestion des espaces disponibles ce qui a pour conséquence sur la gestion du temps et des espaces et une grande frustration des automobilistes.

➤ **2- Motivation :**

La principale motivation de notre projet est d'apporter des solutions concrètes et efficaces aux différents problèmes sus citent grâce à l'Internet des objets. A cet effet nous avons pensé à un système de gestion de parking intelligent, qui nous amène à une meilleure utilisation des espaces de stationnement, tout en réduisant la congestion, avec une facilité d'accès et de réservation. Enfin une amélioration globale de l'expérience des automobilistes.

De plus, nous avons été inspirés par le volet socio-économique d'une gestion efficace des parkings. Ce projet permet donc une meilleure exploitation des espaces de stationnement, en réduisant considérablement la pollution, mais surtout en améliorant la fluidité urbaine en offrant des services pratiques aux conducteurs.

➤ **Objectifs de la recherche :**

Dans cette section, nous allons définir les objectifs de notre recherche sur l'Internet des objets et la conception d'un système de gestion de parking innovant. Ces derniers orientent notre travail et déterminent les résultats que nous souhaitons atteindre.

➤ **Objectif principal :**

Notre objectif principal est la conception d'un système de stationnement intelligent basé sur l'Internet des objets, qui améliore l'efficacité et l'accueil dans la gestion des parkings.

➤ Objectifs spécifiques

Pour répondre aux exigences de notre objectif principal nous avons tracé les objectifs spécifiques suivants :

- Analyser les règles d'action et les connaissances de l'Internet des objets dans le domaine de la gestion des parkings, tout en examinant les travaux de recherche existants, les technologies utilisées et les solutions mises en œuvre.
- Chercher à comprendre les besoins ainsi que les exigences des conducteurs et des gestionnaires de parkings, afin de concevoir un système qui réponde de façon efficace à leurs attentes.
- Conception d'une architecture de système de gestion de parking basée sur l'Internet des objets, en déterminant les composants clés tout en définissant leurs interactions.
- Chercher les technologies et les dispositifs adéquats pour la mise en œuvre du système, tels que les capteurs de détection de places de stationnement, les systèmes de communication sans fil et les plateformes de gestion des données.
- Mettre en place puis tester un prototype fonctionnel du système de gestion de parking, en utilisant des outils de développement adaptés à l'Internet des objets.
- Mettre en place puis effectuer une évaluation de l'efficacité du système de gestion de parking tout en faisant des tests et des mesures quantitatives, avec analyse de l'efficacité de l'optimisation des espaces et la réduction de la congestion.
- Évaluer le système du point de vue des utilisateurs, tout en proposant des améliorations.
- Enfin fournir des recommandations d'ordre pratiques pour le déploiement et l'adoption du système de gestion de parking basé sur l'Internet des objets, en tenant compte des volets techniques, économiques et sociaux.

En réalisant ces 08 objectifs, nous pourrons enfin affirmer qu'on a proposé une solution innovante et efficace basée sur l'Internet des objets

Chapitre 01

Introduction à l'Internet des objets

Et à la découverte des services IoT

➤ **I. Internet des objets (Internet of Things)**

○ **I.1 Historique**

C'est en 1999 que fut utilisé pour la première fois le terme l'Internet des Objets par un britannique du nom de Kevin Ashton, pour décrire un système dans lequel des objets physiques sont connectés à Internet. L'IoT comporte des systèmes pouvant créer et transmettre des données afin de générer de la valeur pour les utilisateurs par le biais de différents services comme l'agrégation et l'analyse de données [1].



Figure 1: Internet des objets

○ **I.2 Définition**

Créé pour faciliter les connexions réseau et technologiques, l'Union internationale des télécommunications (UIT) explique que l'Internet des objets (IoT) forme une infrastructure d'information mondiale. Cette infrastructure permet l'interconnexion d'objets réels et virtuels pour fournir des services avancés - tout cela grâce aux technologies de communication et d'information actuelles ou en développement. Depuis, la définition de l'IoT s'est élargie pour intégrer les acteurs de la connexion au-delà des seuls fabricants de capteurs et éditeurs de logiciels.

Cette expansion comprend de nouveaux opérateurs du marché, des opérateurs historiques, des intégrateurs et beaucoup d'autres. Cette pluralité fait la richesse de cet écosystème. Les appareils sont non seulement connectés à Internet, mais aussi à d'autres appareils locaux. Ils peuvent aussi communiquer directement avec d'autres appareils sur

Internet. Le concept de l'IoT social (SIoT) fait son apparition, là où l'intelligence humaine est complètement intégrée [1].

II. Composantes de l'IoT

Les composants de l'IoT sont résumés en éléments clés qui sont :



Figure 2: Composantes de l'IoT [10].

○ II.1 Les capteurs

Les capteurs sont des éléments matériels importants. Ils sont essentiels au succès de la connexion. Ils sont capables de convertir des grandeurs physiques en signaux numériques. Ce sont des appareils qui détectent les changements environnementaux et mesurent les données. Ils sont largement utilisés en conjonction avec les logiciels. L'intégration de capteurs dans divers objets leur permet de mesurer la température, la lumière, le mouvement et d'autres paramètres physiques présents dans leur environnement. Différents types de capteurs sont disponibles à cet effet.

Ces derniers jouent un rôle primordial en utilisant des données du monde physique en les convertissant en informations exploitables ultérieurement par des logiciels. Ils ont été intégrés de nos jours dans les domaines de l'automatisation industrielle, les véhicules autonomes, les appareils médicaux, de surveillance, électroniques etc... [2].

○ II.2 Les réseaux de capteurs

Ils sont constitués de capteurs équipés de dispositifs sans fil pour l'émission et la réception de données, afin de répondre à leurs besoins de communication mutuelle. Cependant, la simple présence de ces dispositifs sans fil ne suffit pas à rendre un ensemble de capteurs

accessible ou du moins interopérable de manière transparente et simplifiée. A cet effet, il faut préciser que pour atteindre cet objectif, les capteurs doivent également s'organiser entre eux.

Ce qui définit un réseau de capteurs, c'est que ce sont des éléments constitués de petits appareils dotés de capacités de transmission sans fil [2].

○ **II.3 L'énergie**

L'énergie représente la contrainte la plus importante à laquelle les réseaux de capteurs sont confrontés. L'autonomie des nœuds, c'est-à-dire la durée pendant laquelle ils peuvent fonctionner de manière autonome, est évaluée en termes d'années [2].

○ **II.4 Les actionneurs**

Les actionneurs sont des dispositifs qui transforment une donnée numérique en un phénomène physique afin de générer une action. Ils sont en quelque sorte l'inverse des capteurs. On citera comme exemples : les afficheurs, les alarmes, les caméras, les haut-parleurs, les interrupteurs, les lampes etc.... [2].

○ **II.5 La connectivité**

La connectivité de l'objet est assurée par une antenne de radiofréquence qui permet la communication de l'objet avec un ou plusieurs réseaux. Les commandes et les données opérationnelles peuvent être reçues par des objets en plus de transmettre des informations telles que l'identité, l'état, les alarmes ou les données des capteurs. Le module de connexion est responsable de la gestion du cycle de vie de l'objet, qui implique l'authentification, l'enregistrement réseau et l'exploitation [2].

○ **II.6 Traitement de données**

Dès que les données collectées par les capteurs arrivent au cloud, elles sont traitées pour en extraire des informations utiles. Cette étape peut varier en fonction de la nature des données et des objectifs spécifiques.

Dans certains cas, le traitement peut être relativement simple, comme vérifier si les lectures de température provenant de dispositifs tels que les climatiseurs ou les radiateurs se situent dans des limites acceptables. Ceci a pour but de détecter d'éventuels dysfonctionnements ou de performances.

L'analyse d'images recueillis par des caméras de surveillance ou des drones peut parfois rendre le traitement des données assez difficile. Ces situations peuvent nécessiter la mise en

œuvre de techniques de vision par ordinateur pour des identifications d'objets, de personnes ou de modèles spécifiques. Remarquablement, la reconnaissance faciale ou la détection d'anomalies peuvent même être exécutées [3].

○ **II.7 Interface utilisateur**

L'interface utilisateur (IU) dans joue un rôle essentiel en offrant aux utilisateurs une expérience conviviale pour interagir avec les données collectées. Son objectif est de rendre les informations compréhensibles et accessibles, facilitant ainsi la prise de décisions et le contrôle des dispositifs connectés. [3].

➤ **III. Domaines d'application de l'Internet des objets**

Il y a quelques années à peine, on en entendait à peine parler, et maintenant ils sont partout. Les objets connectés ont envahi notre quotidien sans que nous y prêtions la même attention. Des télévisions intelligentes aux voitures connectées, ces nouveaux outils facilitent nos loisirs et nos déplacements, améliorant grandement notre confort. Le potentiel des objets connectés est immense. Ses applications sont variées et multiples et couvrent de nombreux domaines comme l'industrie, les sciences, la santé, etc. [4].

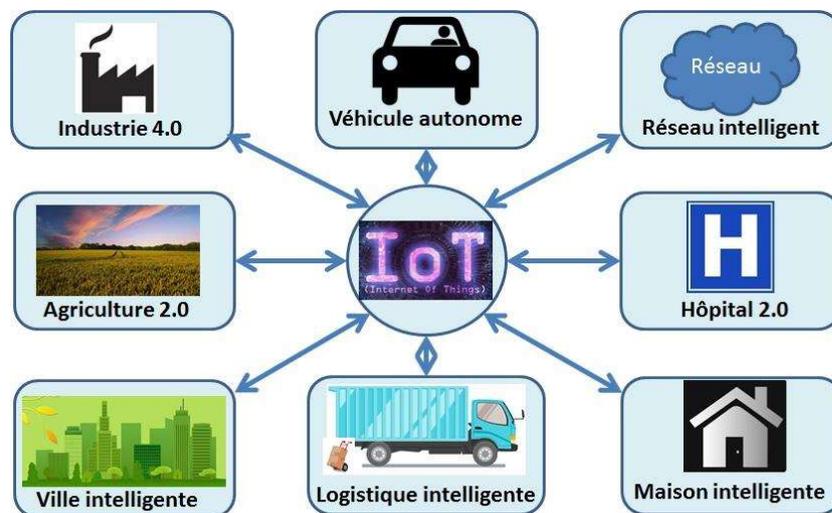


Figure 3: Domaines d'application de l'Internet des objets [9].

L'utilisation de l'IoT permettra l'émergence de plusieurs applications intelligentes qui auront un impact sur les domaines suivants :

- **La domotique intelligente** : celle-ci désigne l'ensemble des techniques visant le contrôle automatisé des appareils électriques, de l'éclairage, de la sécurité et de la gestion de l'énergie dans les maisons [4].



Figure 4: La domotique intelligente

- **Santé connectée** : Elle concerne les dispositifs médicaux portables, le suivi à distance de certains malades, la gestion des maladies chroniques et l'optimisation des soins de santé [4].

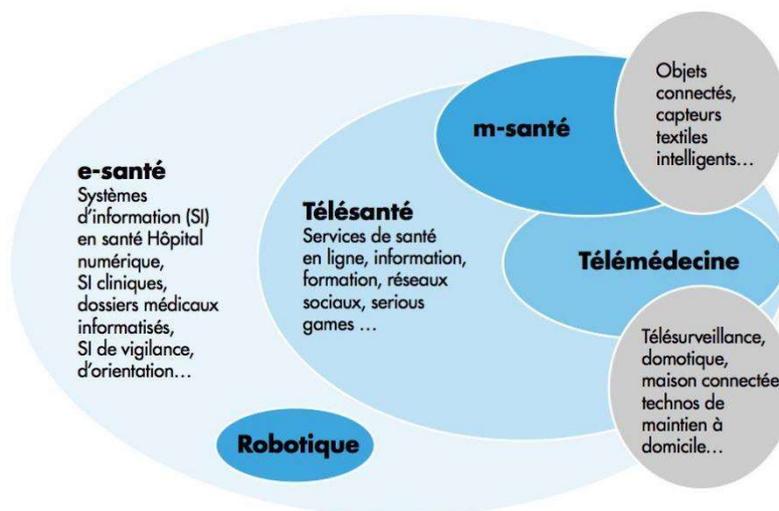


Figure 5 : Santé connectée (HealthCare)

- **Villes intelligentes** : C'est la gestion intelligente de toutes les infrastructures d'une ville qui inclut le contrôle de la qualité de l'air, des transports publics, ainsi que l'amélioration de la sécurité dans les espaces publics [4].

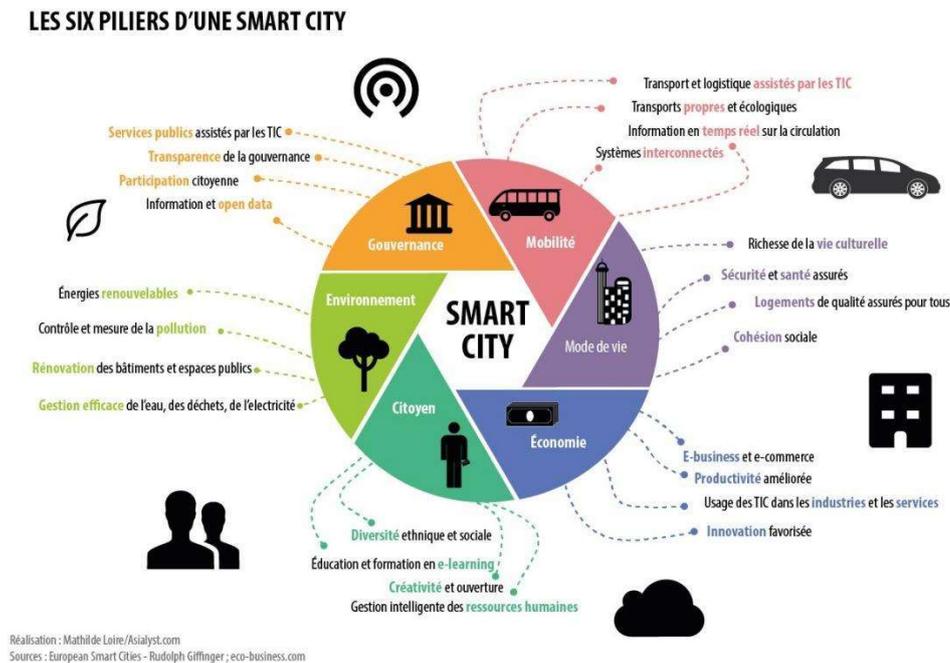


Figure 6: Ville intelligentes [11].

- **Agriculture intelligente** : Elle concerne la gestion de l'agriculture qui comprend les moyens de l'irrigation, les conditions météorologiques et la production agricole [4].
- **Logistique et transport** : Concernant toute la logistique de transport avec le suivi des colis en temps réel, tout en réduisant les coûts grâce à une bonne optimisation des itinéraires et une gestion intelligente de la flotte [4].
- **Industrie** : consiste en la surveillance et la maintenance prédictive des équipements industriels, gestion de la chaîne d'approvisionnement, optimisation des processus de fabrication et augmentation de l'efficacité globale [5].
- **Environnement** : L'objectif principal étant de préserver l'écosystème en assurant des économies d'énergie, une surveillance de la qualité de l'eau et une bonne gestion des déchets.

Ces multiples exemples illustrent donc les multiples applications de l'IoT dans divers domaines et démontrent son potentiel à changer notre façon de vivre et de travailler [6].

➤ **IV. La découverte de services IoT**

Avec l'émergence de l'internet des objets, des données riches sur les objets connectés et des événements du monde réel seront générés en grande quantité et stockées dans des systèmes d'information hétérogènes largement distribués [31] [30].

En raison du nombre excessif d'objets connectés, rechercher et découvrir des services spécifiques offerts par ces objets devient une tâche difficile [34].

Le principal défi est de savoir comment trouver le service requis dans l'IoT en respectant la grande quantité, la mobilité, l'hétérogénéité et la large distribution des services déployés dans des appareils contraints et en même temps souhaitable pour les utilisateurs [32].

La découverte des services IoT représente un outil indispensable pour permettre aux clients tels que les plates-formes middleware, les utilisateurs finaux et les applications de localiser les ressources disponibles pour l'IoT (Capteurs, actionneurs, services, données de contexte), répondant aux changements de l'environnement IoT et à la situation de l'utilisateur.

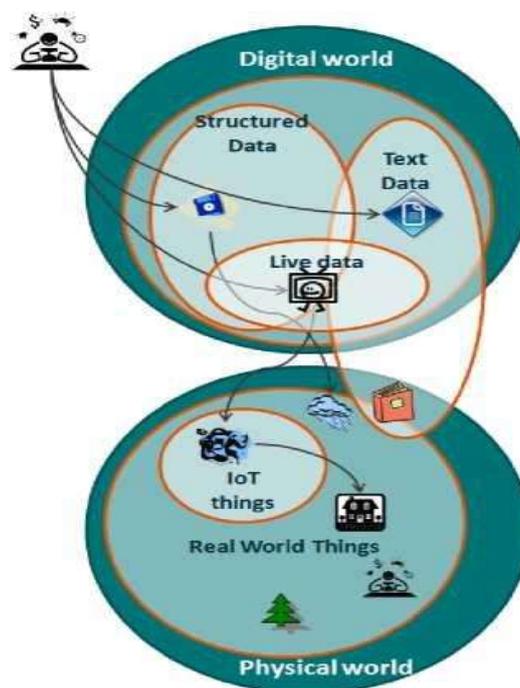


Figure 7 : Boucle de découverte dans le monde IoT [30].

○ **IV.1 Différences entre les services IoT et les services Web Traditionnels**

Même si la découverte des services représente un sujet bien étudié dans les systèmes distribués, les approches traditionnelles ne sont pas adaptées à l'IoT, à cause des différences entre les services du monde réel et les services Web traditionnels [32] [34] :

Tableau 1 : Services IoT et Services Web traditionnels.

	Services IoT	Services Web Traditionnel
Ressources	Limitées : Capacité du calcul, bande passante de réseau, batterie et capacité de stockage...	Riches : Puissance de calcul, capacité énorme de stockage, bande passante importante...
Environnements	Très dynamiques : Déconnexion intermittente du réseau sans fil et la mobilité des appareils	Très stables : Connexion câblé, dispositifs de stockage stables et sécurisés
Etat d'objets	Quasi temps réel	Entièrement virtuelles.

○ **IV.2 Défis et exigences de la découverte des services IoT**

La nature des objets et la dynamique de l'environnement IoT ; apporte à la découverte du service de nouveaux des défis et exigences tels que c'est décrit dans le tableau suivant [33] :

Tableau 2 : Exigences et les défis de la découverte de services IoT

Exigences IoT	Défis IoT
Grande quantité	Fiabilité
Mobilité	Dynamicité
Hétérogénéité	Performance
Large distribution	Evolutivité
Appareils contraints	Interopérabilité
Sensibilité au contexte	Sécurité et Confiance

➤ **V. Conclusion**

Dans ce chapitre, nous avons exploré les diverses facettes de l'internet des objets qui reste une technologie qui a complètement révolutionné notre vie en s'insérant dans notre quotidien. Qu'ils soient d'ordre environnemental ou simplement pour améliorer notre confort, les bénéfices de cette technologie sont notoires et chamboulent tous nos codes et usages et vont continuer à bâtir et dessiner notre futur. Dans le chapitre à venir, nous examinerons davantage l'état de l'art sur les Parkings Intelligents.

Chapitre 02

Un état de l'art sur les Parkings Intelligents

➤ **I. Introduction**

Avec l'urbanisation croissante, l'utilisation des véhicules à moteur a augmenté, causant des problèmes de circulation et de stationnement. En fait, la plupart des embouteillages urbains sont causés par des propriétaires de véhicules qui cherchent une place pour se garer. Cela est dû à un certain nombre de facteurs, notamment le nombre croissant de voitures dans les villes, le manque de places de stationnement et la manière chaotique dont les voitures sont souvent garées. Par conséquent, les gens perdent beaucoup de temps à chercher une place de stationnement, ce qui peut entraîner de la frustration et du stress. De plus, la recherche de stationnement peut contribuer à la congestion du trafic et à la pollution.

➤ **II. Stationnement traditionnel**

Un parking traditionnel est un espace de stationnement temporaire des véhicules. Il est généralement situé dans un garage de stationnement ou sur le bord de la route. Le stationnement est l'action d'arrêter et de désengager un véhicule et de le laisser inoccupé.

Les parkings traditionnels présentent un certain nombre de problèmes. Ils sont d'abord souvent très étroits pour accueillir toutes les voitures qui doivent s'y garer. Cela peut entraîner le stationnement de voitures dans des zones non autorisées, comme devant des bouches d'incendie ou des places de stationnement pour handicapés [12].

Les parkings traditionnels peuvent être très chaotiques. Les voitures sont souvent garées au hasard, ce qui peut rendre difficile la recherche d'une place. De plus, le stationnement chaotique peut entraîner souvent des accidents [13].



Figure 8 : Parking traditionnel.

➤ III. Stationnement intelligent

➤ Le stationnement intelligent est une nouvelle technologie conçue pour résoudre les problèmes du stationnement traditionnel. Les systèmes de stationnement intelligents utilisent des capteurs, des caméras ou d'autres dispositifs pour suivre la disponibilité des places de stationnement. Ces informations sont ensuite transmises aux conducteurs en temps réel, afin qu'ils puissent facilement trouver une place [14].

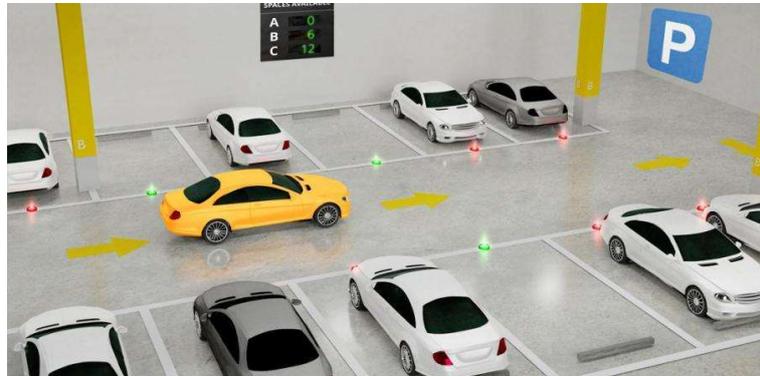


Figure 9 : Exemple-1 Smart-Parking



Figure 10 : Exemple-2 Smart-Parking

Les systèmes de stationnement intelligents présentent un certain nombre d'avantages par rapport aux parkings traditionnels puisqu'ils sont beaucoup plus efficaces et peuvent utiliser jusqu'à 50 % d'espace en moins que les parkings traditionnels. En effet, ils peuvent utiliser l'espace vertical ainsi que l'espace horizontal.

Les systèmes de stationnement intelligents sont beaucoup plus pratiques pour les conducteurs. Ces derniers peuvent facilement trouver une place de stationnement sans avoir à tourner autour d'un parking pendant une longue période, ce qui leur fait perdre un temps précieux. Ils peuvent aussi aider à réduire les embouteillages et la pollution.

III.1 Principe de fonctionnement du système de smart parking

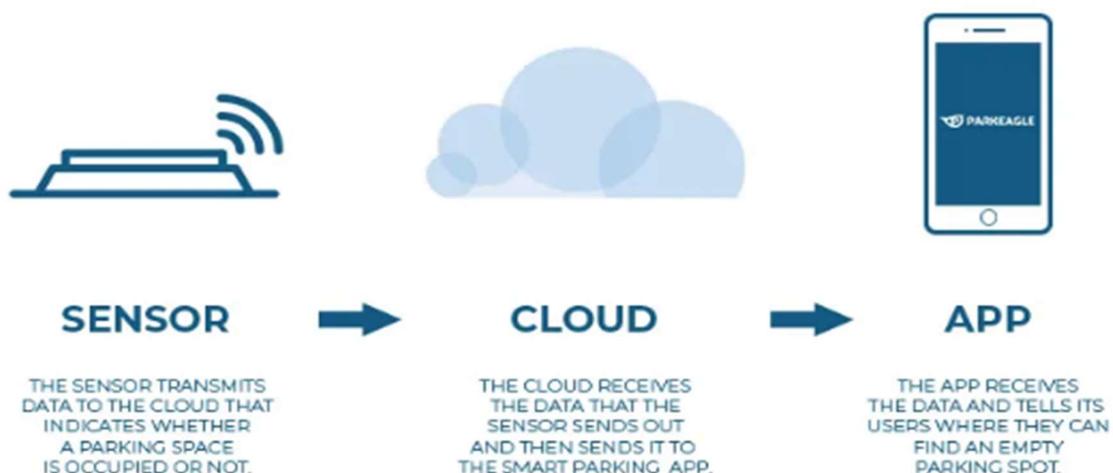


Figure 11: Fonctionnement du système de smart parking [15].

III.2 Les capteurs utilisés dans les systèmes de parking intelligent

Il existe plusieurs capteurs qui facilitent la détection des informations relatives à l'occupation des places de stationnement. Chaque capteur a des caractéristiques et des avantages. Les capteurs les plus largement adoptés sont les suivants :

III.2.1 Capteur infrarouge (IR)

Les plus utilisés dans les smart parkings, et il en existe deux types :

- **Capteur infrarouge passif (PIR)** : ces capteurs détectent les changements d'énergie thermique lorsqu'un véhicule occupe une place de stationnement, ils conviennent surtout aux parkings fermés situés à l'intérieur des bâtiments et ne conviennent pas aux parkings extérieurs ou ouverts. Les PIR doivent être placés sous le sol ou au plafond [16].

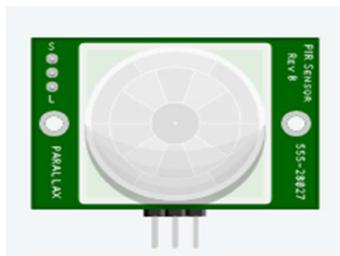


Figure 12: Capteur PIR

- **Capteur infrarouge actif** : Ces capteurs émettent de l'énergie infrarouge permettent de détecter tout objet ou véhicule par la quantité d'énergie thermique réfléchi. Ils sont également sensibles au changement climatique (pluies, neige,). Ces capteurs sont généralement placés en hauteur et conviennent aux parkings intérieurs fermés [16].

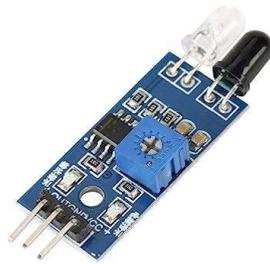


Figure 13 : Capteur IR actif.

III.2.2 Capteur ultrasonique

Ces capteurs émettent des ondes sonores entre 25 et 50 KHz et détectent les objets en fonction des rebonds sonores réfléchis. Ils sont généralement montés au plafond et sont sensibles aux changements environnementaux (pluies, neige,). En fonction de la distance à laquelle les ondes sont réfléchies, ils peuvent faire la distinction entre un véhicule et une personne. Afin d'obtenir l'état d'occupation du parking, ces capteurs doivent être placés au-dessus de chaque place de parking [16].



Figure 14 : Capteur ultrasonique.

III.2.3 Identification par radiofréquence (RFID)

Chaque véhicule recevra une étiquette de radio fréquence pour être identifié et ceci grâce à un émetteur-récepteur et une antenne installée à l'entrée du parking. Ces systèmes sont adoptés aux parkings fermés et couverts. Ils ne conviennent pas aux parkings ouverts, car ils sont librement disponibles [16].



Figure 15: Capteur d'identification par radiofréquence (RFID).

III.2.4 Magnétomètre

Ces capteurs détectent la présence d'un véhicule en détectant la variation du champ électromagnétique. Ils doivent être proches du véhicule, c'est pourquoi ils sont placés sous la surface. Ils conviennent aux parkings ouverts et fermés. Il existe des capteurs sans fil qui ont une autonomie de quelques années et qui permettent de détecter en temps réel l'occupation d'un parking. Les capteurs doivent être placés sous chaque place de stationnement pour connaître l'occupation des places [16].



Figure 16 : Capteur sous-sol (Magnétomètre).

III.2.5 Caméras

Les dispositifs de solution IoT de stationnement basés sur la caméra sont une approche technologique relativement nouvelle qui a évolué avec les progrès de la vision par ordinateur. Cette méthode de traitement d'image/vidéo nécessite un support de surveillance informatique pour identifier toutes les voitures dans un espace de stationnement. Alors que les caméras peuvent surveiller plusieurs places de stationnement, cette approche peut avoir des angles morts. De plus, le traitement de l'image ou la vidéo en temps réel nécessite un matériel de traitement Cloud de plateforme. Il s'agit d'une solution coûteuse étant donné que les caméras et les systèmes informatiques ne peuvent pas être utilisés de manière évolutive. Les solutions basées sur des caméras ne peuvent vraiment être considérées comme une approche viable que dans des champs ouverts relativement grands [17].

➤ **IV. Conclusion :**

Dans ce chapitre, nous avons vu les parkings intelligents et leur rôle important, car ils contribuent à réduire les embouteillages et de nombreux problèmes d'accessibilité urbaine. Les parkings traditionnels présentent plusieurs inconvénients si on les compare aux parkings intelligents, notamment en raison de leur mode de fonctionnement, mais surtout des capteurs utilisés. Dans le chapitre suivant, nous aborderons la conception et la réalisation.

Chapitre 03

Conception et Réalisation

➤ **I. Introduction :**

Les systèmes de stationnement intelligent sont devenus essentiels pour résoudre les problèmes de stationnement dans les zones urbaines densément peuplées. En utilisant des technologies avancées telles que l'Internet des objets (IoT) et l'intelligence artificielle, ces systèmes offrent une gestion optimisée du stationnement en améliorant l'expérience des conducteurs avec une meilleure utilisation des espaces disponibles.

Notre projet se concentre sur la conception et l'optimisation d'un système de stationnement intelligent en utilisant des approches multicritères et la découverte des services IoT. L'objectif est de fournir des recommandations personnalisées aux conducteurs. Cette recherche vise à améliorer l'efficacité et la durabilité du stationnement dans les environnements urbains en adoptant une approche multidisciplinaire et en exploitant les technologies émergentes.

Le but de notre projet est de mettre en place un prototype de stationnement intelligent (Smart-Parking) multicritères sous forme d'un système d'aide à la décision intelligent qui peut aider et orienter l'utilisateur à avoir la position de stationnement la plus pertinente suite à ses besoins d'une manière simple. En utilisant une application Android avec un interface graphique l'utilisateur peut facilement saisir sa requête et visualiser les Top-K des parcs les plus pertinents.

➤ **II. Architecture proposée**

Pour réaliser notre projet nous avons proposé une architecture IoT basée sur trois couches :

○ **II.1 Couche physique et réseau**

Dans cette couche on trouve des compagnies qui représentent chacune l'ensemble des capteurs installés dans les positions de stationnement d'un parc donné, inscrit dans notre système, ainsi que les moyens réseaux nécessaires pour assurer la connexion des capteurs avec l'internet.

Chaque compagnie contient un référentiel d'événements qui stocke les adresses des capteurs de la compagnie ainsi que les états des positions dont ils sont liés. À chaque

changement d'état (occupée ou libre) un message est envoyé via le Gateway réseau vers la couche middleware.

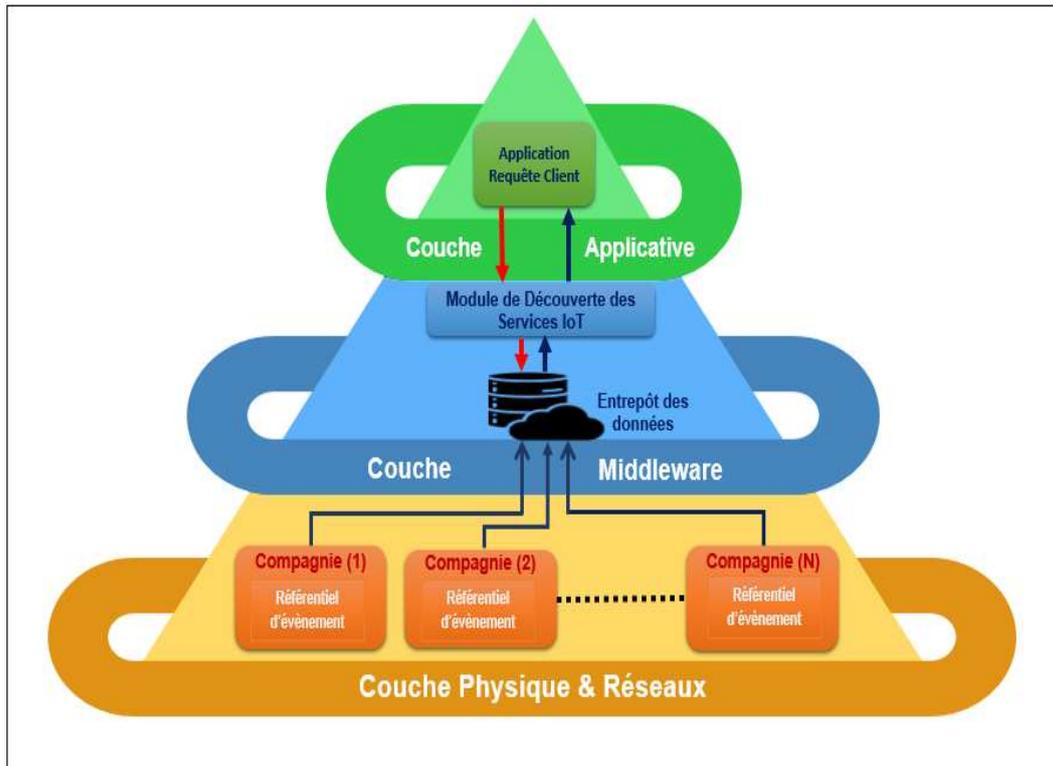


Figure 17: L'architecture proposée.

○ II.2 Couche middleware :

C'est une couche intermédiaire qui assure et permet la communication entre le monde virtuel (Couche applicative) et le monde physique (Couche Physique et réseaux). Dans notre architecture cette tâche est assurée par deux modules :

- **Module entrepôt des données** : Il joue le rôle d'une base de données où sont stockés les états des positions de stationnement envoyés par la couche physique et réseau.
- **Module de découverte des services IoT** : Ce module représente le noyau de notre système car c'est lui qui alimente la couche applicative par les données nécessaires en temps réel pour qu'elle puisse répondre à la requête utilisateur. Ce Module utilise une approche de découverte des services par compagnie (découverte horizontale), en interrogeant l'état des positions liées au parking demandé par le client au niveau de l'entrepôt des données, et en envoyant au client un état de parking contenant les positions libres et occupées.

○ **II.3 Couche applicative**

Cette couche comprend l'interface utilisateur de notre système représenté par une application de requête utilisateur (Application Android) qui permet aux clients de notre système de s'inscrire pour bénéficier des services offerts en temps réel.

➤ **III. Principe de fonctionnement de notre système.**

○ **III.1 Choix Multicritères**

Plusieurs critères sont utilisés pour déterminer le choix d'une place de stationnement (exemple : Prix, Distance, Réputation, Impact environnemental, Politiques de remboursement, Présence de services supplémentaires (Ex : recharge pour les véhicules électriques), Taille des places de parking, Temps d'attente ...)

Dans notre système de stationnement intelligent, les critères de disponibilité, de prix et de réputation sont intégrés de manière cohérente dans le processus de recommandation.

Chaque critère apporte une dimension spécifique à l'évaluation des places de stationnement et joue un rôle clé dans la prise de décision des conducteurs.

La disponibilité est un critère crucial car elle permet de déterminer si une place de stationnement est réellement accessible à un moment donné. Les conducteurs peuvent filtrer les options de stationnement en fonction de leur disponibilité, ce qui garantit une correspondance entre leurs besoins et les places de stationnement disponibles.

Le prix et la réputation, bien que statiques, sont des critères importants qui contribuent à la qualité globale de chaque place de stationnement. Les conducteurs peuvent définir leurs préférences en termes de prix minimal et de réputation maximale, ce qui leur permet de trouver des options qui répondent à leurs critères de sélection personnels.

○ **III.2 Qualité statique et dynamique des Critères utilisés**

▪ **III.2.1 La Disponibilité en tant que Donnée IoT Dynamique**

La disponibilité des places de stationnement est un critère essentiel dans le processus de recommandation du système de stationnement intelligent. Contrairement au prix et à la réputation, qui sont des données statiques, la disponibilité est une donnée dynamique qui évolue en temps réel. Cette caractéristique dynamique est rendue

possible grâce à l'utilisation de l'Internet des objets (IoT) et des capteurs intégrés aux places de stationnement.

Les capteurs IoT installés dans les places de stationnement détectent l'occupation ou la libération des espaces, fournissant ainsi des informations précises sur leur disponibilité. Ces données sont collectées en temps réel par un système central, qui les utilise pour actualiser continuellement la base de données des places de stationnement disponibles. Les conducteurs bénéficient ainsi d'une vision en temps réel de la disponibilité des places de stationnement dans différentes zones.

La nature dynamique de la disponibilité en tant que donnée IoT présente plusieurs avantages. Elle permet aux conducteurs d'accéder à des informations actualisées et fiables, ce qui facilite leur processus de décision lorsqu'ils cherchent un stationnement. De plus, cette fonctionnalité permet au système de recommandation d'adapter ses suggestions en fonction des variations de disponibilité, offrant ainsi une expérience utilisateur optimisée.

▪ III.2.2 La Nature Statique du Prix et de la Réputation

Contrairement à la disponibilité, le prix et la réputation sont des critères statiques dans le contexte du système de stationnement intelligent. Le prix est généralement fixé par les fournisseurs de stationnement et peut varier en fonction de différents facteurs tels que l'emplacement, la période de la journée ou la demande. Cependant, une fois défini, le prix reste relativement constant, du moins à court terme.

De même, la réputation d'une place de stationnement est basée sur des évaluations et des commentaires d'utilisateurs précédents qui sont statiques à court terme. Ces évaluations reflètent l'expérience passée des conducteurs et peuvent fournir une indication sur la qualité et la satisfaction associée d'une place de stationnement donnée. Cependant, la réputation est considérée comme une donnée statique car elle évolue moins fréquemment par rapport à la disponibilité en temps réel.

➤ IV. Algorithmes utilisés dans les systèmes de stationnement intelligents

L'algorithme le plus simple et couramment utilisé est l'algorithme de recherche linéaire. Il consiste à parcourir toutes les places de stationnement disponibles dans une zone donnée et à sélectionner celle qui répond au critère de disponibilité ou de distance. Cependant, cette

approche peut être inefficace. Cet algorithme explore toutes les places de stationnement disponibles et sélectionne la première qui satisfait le critère de disponibilité.

Certains systèmes de stationnement intelligent utilisent également des techniques de machine Learning pour prédire la disponibilité des places de stationnement. Les modèles d'apprentissage automatique, tels que les réseaux de neurones et les arbres de décision, sont entraînés sur des données historiques pour prédire la disponibilité future des places de stationnement mais il reste moins efficace face à un algorithme qui traite des données en direct et multicritères.

Différents algorithmes ont été développés pour gérer cette complexité et faciliter la prise de décision multicritères. L'approche la plus populaire est l'utilisation des méthodes d'agrégation multicritère. Ces méthodes permettent de combiner les valeurs des critères en une seule mesure globale pour chaque place de stationnement. Parmi les méthodes d'agrégation les plus courantes, on trouve la méthode de somme pondérée (Weighted Sum Method), la méthode de Tchebycheff, la méthode de Prométhée, et la méthode de technique for order preference by similarity to idéal solution (TOPSIS).

○ IV.1 Méthode Utilisée

Dans notre système on a choisi la méthode de somme pondérée qui peut être formulée mathématiquement comme suite :

Soit $C = \{c_1, c_2, \dots, c_n\}$ l'ensemble des critères,

Et $W = \{w_1, w_2, \dots, w_n\}$ l'ensemble des poids associés à chaque critère.

Pour chaque place de stationnement p , le score $S(p)$ peut être calculé comme suit :

$$S(p) = w_1 * f_1(p) + w_2 * f_2(p) + \dots + w_n * f_n(p)$$

Où $f_i(p)$ est la valeur normalisée du critère c_i pour la place de stationnement p .

IV.2 Normalisation des critères

A- Normalisation du prix

La première étape consiste à normaliser le prix en fonction d'une valeur maximale et minimale. Le prix donné est soustrait du prix maximal et le résultat est divisé par la différence entre le prix maximal et le prix minimal. Cela permet d'obtenir une valeur normalisée entre 0 et 1, indiquant la position du prix donné dans la plage de prix possible.

La formule pour normaliser le prix (N) entre 0 et 1 est la suivante :

$$N = (P - P_{min}) / (P_{max} - P_{min})$$

Où :

- P : Est le Prix donné
- P_{max} : Est le prix maximal
- P_{min} : Est le prix minimal :

B- Normalisation de la notation :

La notation est normalisée en utilisant une échelle de notation de 1 à 5. La notation donnée est soustraite de 1 et le résultat est divisé par 4 pour obtenir une valeur normalisée entre 0 et 1, indiquant la position de la notation donnée dans la plage de notation possible.

$$N = (X - 1) / 4$$

où :

- N : Est la valeur normalisée est la notation
- X : Est la notation donnée.

C- Normalisation de la disponibilité

La disponibilité est normalisée en utilisant le nombre de places disponibles dans le parc spécifique et le nombre total de places disponibles dans tous les parcs. Le nombre

de places disponibles dans le parc spécifique est divisé par le nombre total de places disponibles pour obtenir une valeur normalisée entre 0 et 1, indiquant la proportion de places disponibles dans le parc spécifique par rapport au total.

$$N = D / T$$

Où :

- N : est la valeur normalisée de la disponibilité
- D : est le nombre de places disponibles dans le parc spécifique
- T : est le nombre total de places disponibles dans tous les parcs

D- Normalisation de la pénalité

Lorsque le coût total du stationnement dépasse le budget maximal donné par l'utilisateur, une pénalité est appliquée. La pénalité normalisée est calculée en fonction de la différence entre le coût total et le budget maximal, puis en la divisant par le budget maximal. Pour obtenir une valeur négative, la pénalité est inversée et soustraite de 1.

Formule de normalisation de la pénalité :

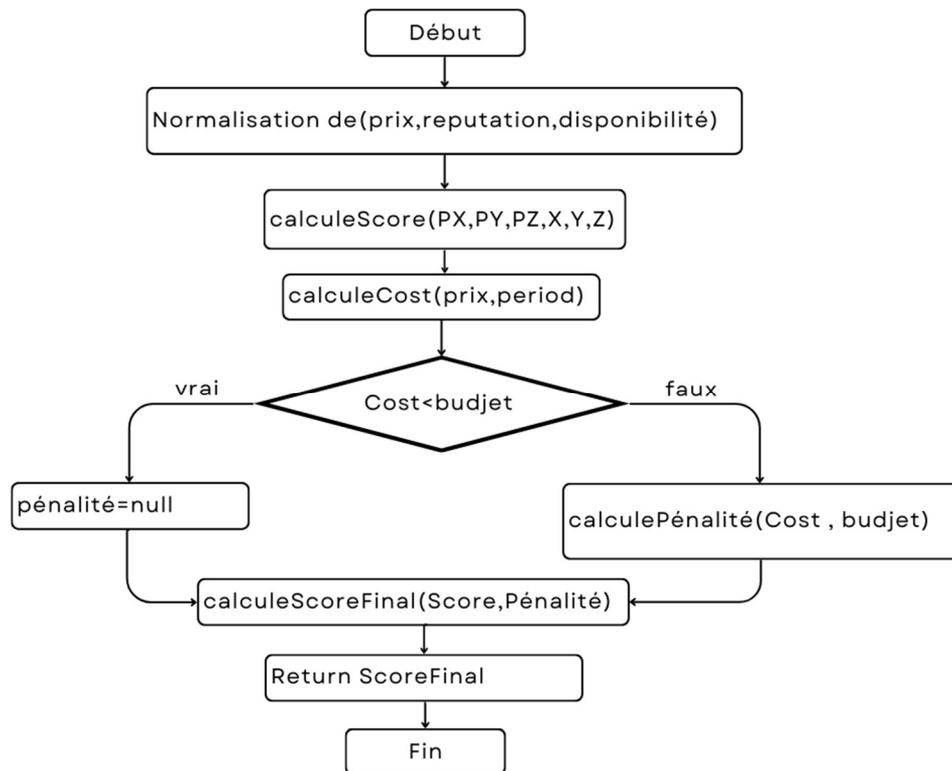
$$P = -((C - B) * 2 / B + 1)$$

Où :

- C : est le coût total du stationnement
- B : est le budget maximal
- P : est la pénalité normalisée

Cette formule attribue une pénalité plus élevée lorsque le coût total du stationnement dépasse davantage le budget maximal. Une pénalité de -1 indique que le coût total du stationnement dépasse le double du budget maximal, tandis qu'une pénalité de 0 indique qu'aucune pénalité n'est appliquée.

IV.3 Segmentation de notre Algorithme utilisé



Où :

- *X* : Le Prix (Associé au stationnement dans un parc donné).
- *Y* : La Notation de parc (Représente la notation globale attribuée à un parc de stationnement, basée sur des facteurs tels que la qualité des services, la sécurité, etc.).

- *Z : La disponibilité* (Indique le nombre de places de stationnement disponibles dans un parc spécifique à un moment donné).
- *PX : Poids du prix*(Indique l'importance relative accordée au critère du prix dans la décision finale).
- *PY : Poids de la notation*(Exprime l'importance relative accordée au critère de la notation dans la prise de décision).
- *PZ : Poids de la disponibilité* (Représente l'importance relative accordée au critère de disponibilité lors de l'évaluation des options de stationnement).

○ IV.4 Matériel utilisé

Pour la réalisation de notre prototype nous avons choisis de mettre en place 3 maquettes, 2 maquettes identiques avec 6 places et la 3-ème maquette comptent 12 places.

Toutes les maquettes utilisent le même matériel Arduino ci-dessus :

Tableau 3: Matériel utilisé.

Composant	Qté	Description
 <p>NodeMCU ESP8266</p>	03	<p>Microcontrôleur léger, puissant, avec Wifi intégré. Idéal pour projets connectés avec interface Web. Créer des objets connectés via réseau Wifi, héberger page web pour contrôle à distance [18].</p>

 <p>Servomoteur SG90</p>	<p>03</p>	<p>Le servomoteur est un actionneur qui produit un mouvement précis en réponse à des commandes externes. Le modèle SG90 est petit, populaire et compatible avec Arduino. Il a un engrenage en plastique qui se dégrade rapidement. Le couple et la vitesse sont des paramètres importants à considérer [19].</p>
 <p>Capteur IR</p>	<p>07</p>	<p>Le capteur infrarouge Arduino détecte le rayonnement infrarouge dans l'environnement, utilisé dans les projets de domotique et de contrôle. Il détecte les changements d'intensité du rayonnement infrarouge émis par diverses sources, fournissant des informations pour des actions basées sur les signaux infrarouges détectés [20].</p>
 <p>LED 5mm</p>	<p>07</p>	<p>LED émet de la lumière lorsque le courant passe. Utilisée dans diverses applications et comme indicateur marche/arrêt. Connecter LED aux broches numériques d'Arduino pour contrôle d'allumage/ extinction programmée [21].</p>

 <p>Ultrasonique</p>	<p>03</p>	<p>Les Capteurs ultrasons (sonar) mesurent la distance comme chauves-souris et dauphins. Capteur HC-SR04 émet signal 40 kHz, reçoit écho réfléchi. Temps déplacement son permet déterminer distance aux obstacles avec précision [22].</p>
 <p>Plaques de prototypage</p>	<p>03</p>	<p>Les plaques de prototypage Arduino sont des plateformes permettant de créer des circuits électroniques personnalisés. Elles offrent des rangées de trous pour connecter les composants et facilitent le prototypage rapide de projets Arduino.</p>
 <p>Jumpers</p>	<p>400</p>	<p>Les jumpers Arduino sont des fils de connexion utilisés pour établir des connexions entre les broches et les composants sur une carte Arduino. Ils permettent de relier facilement et temporairement différents points du circuit lors du prototypage ou du dépannage.</p>

Nous avons choisi l'ESP 8266 pour plusieurs raisons

- Facilité de programmation

- Intégration Wi-Fi
- Faible coût
- Taille compacte
- Grande communauté de développeurs

➤ V. Prototype de projet

○ V.1 Architecture de prototype

Notre prototype est composé de trois maquettes qui représentent les parcs utilisés pour tester notre application.

Le parc 1 et 2 contiennent chacun 5 positions de stationnement et le parc trois contient 12 positions.

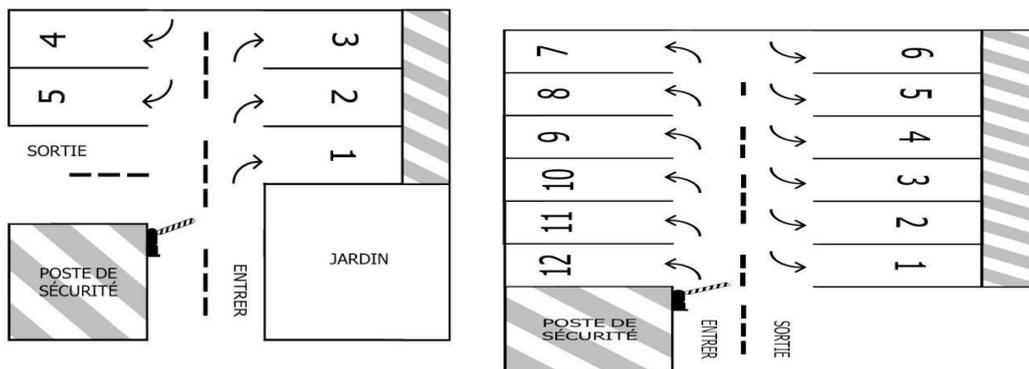


Figure 18 : Architecture des maquettes des parcs 1,2 et 3.

○ V.2 Description de prototype

La figure suivante représente le schéma de branchement pour le système de stationnement intelligent typique :

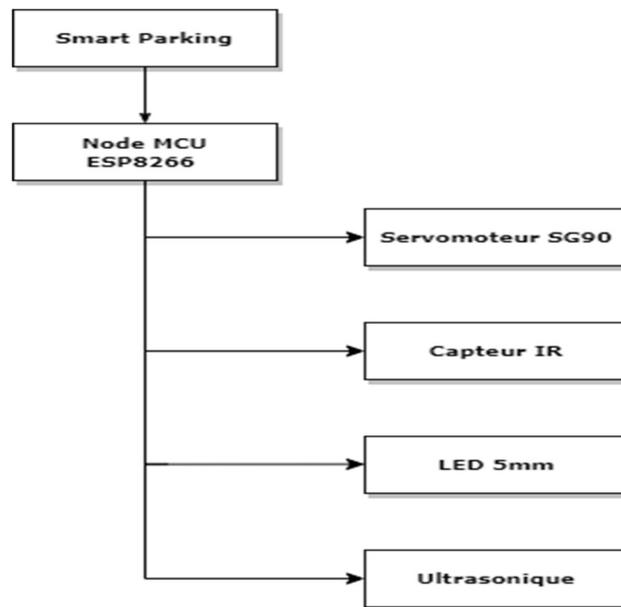


Figure 19: Schéma général de branchement du système prototype.

Pour mieux comprendre le fonctionnement de notre prototype, nous avons décomposé notre projet trois parties :

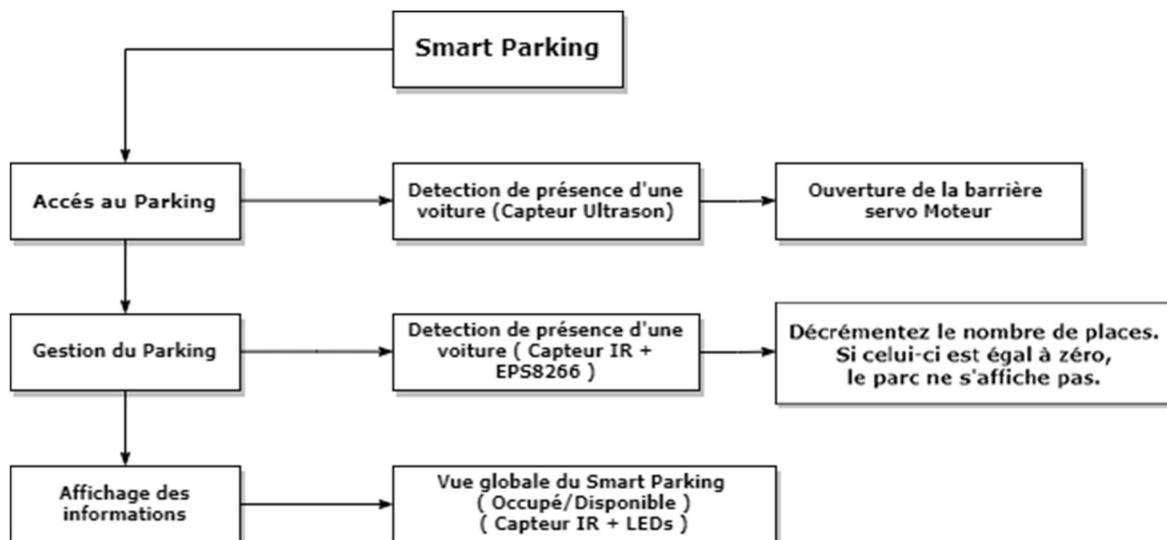


Figure 20 : Représentation schématique du prototype.

○ V.3 Implémentation de prototype

Avant de passer à la réalisation de notre prototype, il est très intéressant de simuler notre prototype, et la figure ci-dessous illustre le circuit de notre maquette :

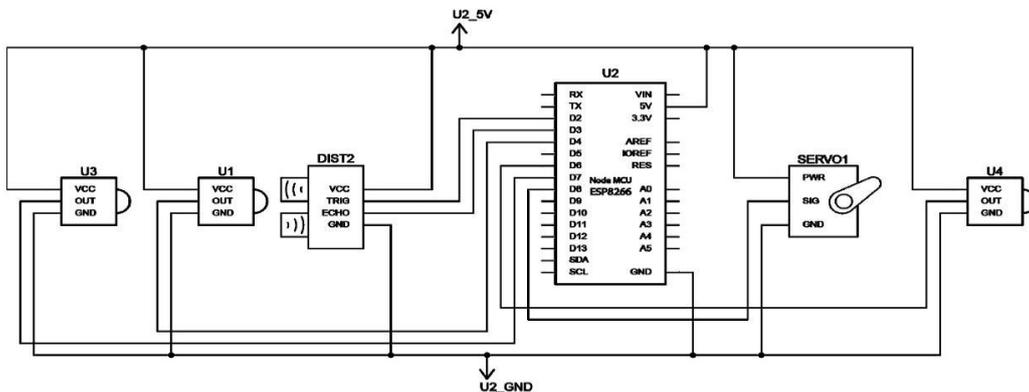


Figure 21: Circuit proposé pour les maquettes.

○ V.4 Interconnexion entre les différents modules de système

Pour assurer la connexion des capteurs IR installés aux places de stationnement de l'internet, nous les avons connectés à la carte ESP8266 qui communique l'état des positions requises à partir des capteurs vers la base de données Firebase en temps réel.



Figure 22: Création des capteurs IR sur Firebase.

La base de données qui a été créée sur Firebase est considérée comme un PMS (Park Management System).

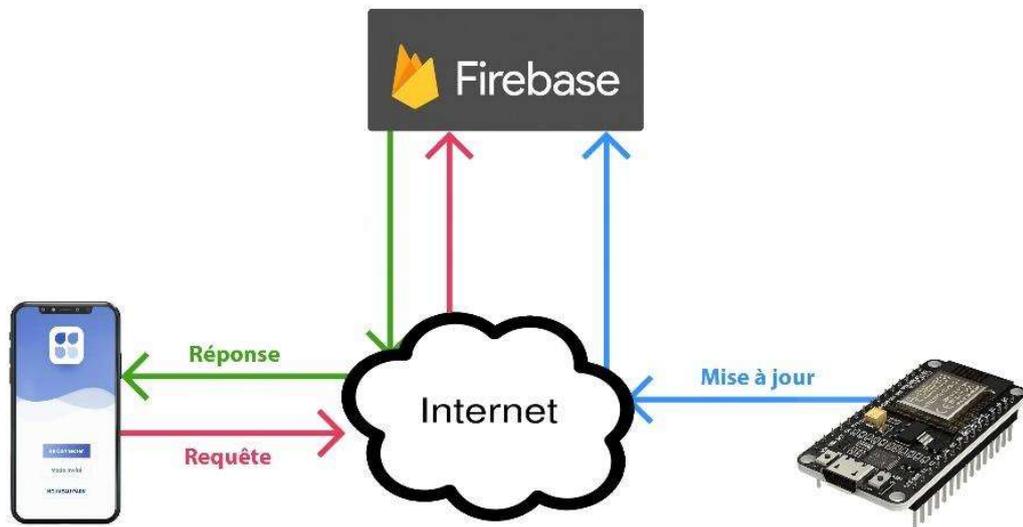


Figure 23: Schéma représente la communication entre les différents modules de système.

➤ VI. Logiciels & Technologies Utilisées

Tableau 4 : Logiciels & Technologies Utilisées.

Logo	Description
 Arduino IDE	L' <i>IDE Arduino</i> est un logiciel open source et gratuit qui permet de compiler des programmes en langage "machine" pour Arduino. Il permet également de téléverser ces programmes dans la mémoire de l'Arduino via le port USB de l'ordinateur, et de communiquer avec la carte Arduino en utilisant le terminal ou le moniteur série. Cette communication est possible tant que la connexion entre l'Arduino et l'ordinateur est active [23].

 <p>Firebase</p>	<p><i>Firebase</i> est une plateforme de développement d'applications mobiles et Web de Google. Elle offre des outils pour créer des applications de qualité, développer l'audience et augmenter les revenus. Firebase utilise une base de données NoSQL qui stocke les données au format JSON. Ses services comprennent une base de données en temps réel, l'authentification avec Google et un hébergement rapide grâce à Firebase Hosting. [24].</p>
 <p>Photoshop</p>	<p><i>Photoshop</i> est un logiciel de retouche d'images raster qui travaille avec des grilles de pixels. Il est apprécié pour sa capacité à reproduire des dégradés de couleurs subtiles et offre une bibliothèque de filtres et d'outils graphiques puissants. Il est utilisé par les professionnels du graphisme et enseigné dans les écoles d'art, les académies et les studios de création [25].</p>
 <p>Illustrator</p>	<p><i>Illustrator</i> est un logiciel professionnel de conception d'images vectorielles, édité par Adobe. Contrairement aux images bitmap, les visuels enregistrés en format vectoriel sont composés de courbes mathématiques, permettant de redimensionner l'image sans perte de qualité. Illustrator est complémentaire à Photoshop et Indesign, et est largement utilisé par les graphistes [26].</p>
 <p>Android Studio</p>	<p><i>Android Studio</i> est l'IDE officiel pour les applications Android. Basé sur IntelliJ IDEA, il offre de nombreuses fonctionnalités pour améliorer la productivité des développeurs. Il dispose d'un système de construction flexible avec Gradle, d'un émulateur rapide, d'une édition en direct, de modèles de code et d'une intégration GitHub. Il propose également des outils de test, un outil Lint pour détecter les problèmes, la compatibilité C++ et NDK, et une intégration avec Google Cloud Platform [27].</p>

 <p>Java</p>	<p><i>Java</i> est un langage de programmation et une plateforme informatique largement utilisés depuis leur introduction par Sun Microsystems en 1995. Il est devenu essentiel dans le monde numérique, soutenant de nombreux services, applications et produits innovants. Certaines applications et sites Web nécessitent une version de bureau de Java. Les utilisateurs peuvent consulter java.com pour les applications de bureau, dev.java pour la programmation Java et oracle.com/java pour des informations professionnelles [28].</p>
 <p>XML</p>	<p><i>XML</i>, langage de balisage extensible, est utilisé dans la conception de sites Web et pour faciliter l'échange d'informations en ligne. Il permet de formaliser des données textuelles en créant de nouvelles balises. Contrairement à d'autres langages, XML est facile à lire et à comprendre, ne nécessite pas de connaissances techniques avancées et peut être édité avec un simple traitement de texte. Il est générique et compatible avec de nombreux systèmes [29].</p>

➤ VII. Application mobile

○ VII.1 Analyse de besoins

A- Besoins fonctionnels

Cette partie est réservée aux besoins fonctionnels des différents acteurs de l'application. Le système doit assurer :

- L'authentification de l'utilisateur.
- La consultation des places libres.
- L'orientation de l'utilisateur vers les parks disponibles.
- Propose un classement multicritère.
- Ajouter des parkings.

B- Besoins non-fonctionnels

- **Utilisabilité** : l'interface utilisateur est facile à comprendre et à utiliser et elle est attractive.
 - **Disponibilité** : possibilité d'utiliser l'application par n'importe quel utilisateur.
 - **Performance** : les performances du système en termes de temps de réponse.
 - **Maintenabilité** : facilité de modification, stabilité et testabilité.
 - **Sécurité** : le système doit être sécurisé (une sécurité totale) surtout au niveau de gestion des accès aux données.
 - **Ergonomie** : l'utilisateur peut réaliser les tâches de façon sécuritaire, confortable et efficace.
- **VII.2. Modélisation**

Pour notre modélisation d'application, nous avons utilisé trois diagrammes : le diagramme de classe pour modéliser l'aspect statique, le diagramme de cas d'utilisation et d'activité pour modéliser l'aspect dynamique.

➤ **Diagramme de cas d'utilisation**

La figure ci-dessous représente le diagramme de cas d'utilisation donnant à apercevoir des fonctionnements du notre système :

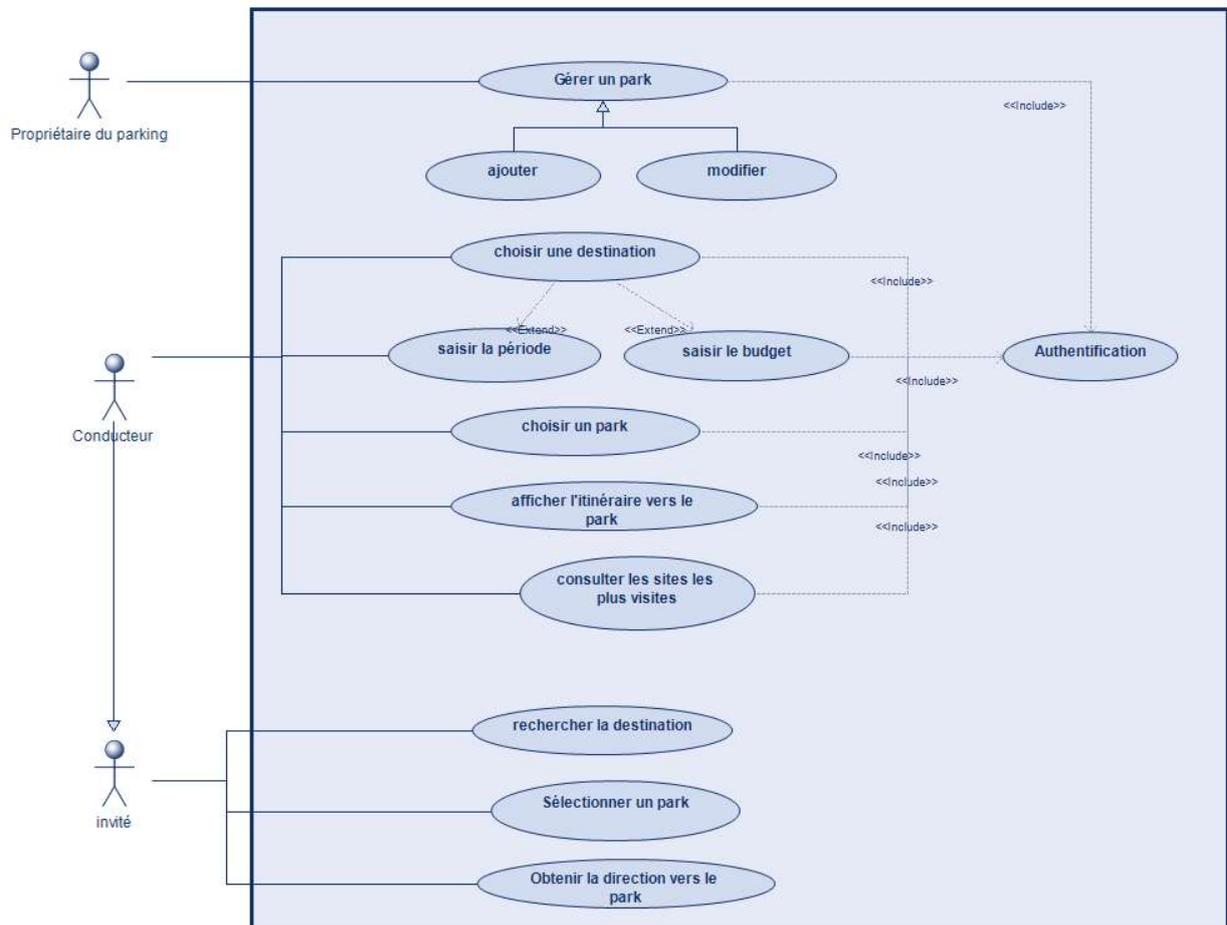


Figure 24: Diagramme de cas d'utilisation.

➤ Diagramme de séquence pour s'authentifier (propriétaire du Park)

Le système envoie au propriétaire du parking un formulaire d'inscription. Après avoir saisi les informations requises, le système demande la localisation du parking. Le propriétaire mentionne alors la localisation, et le système répond en affichant l'écran de connexion. Le propriétaire répond en saisissant les informations d'authentification, et le système valide la connexion.

Une fois l'e-mail et le mot de passe saisis, le système peut répondre par un message indiquant que l'e-mail ou le mot de passe est incorrect. Le propriétaire peut alors ressaisir le mot de passe, et le système donne l'accès à la page d'accueil.

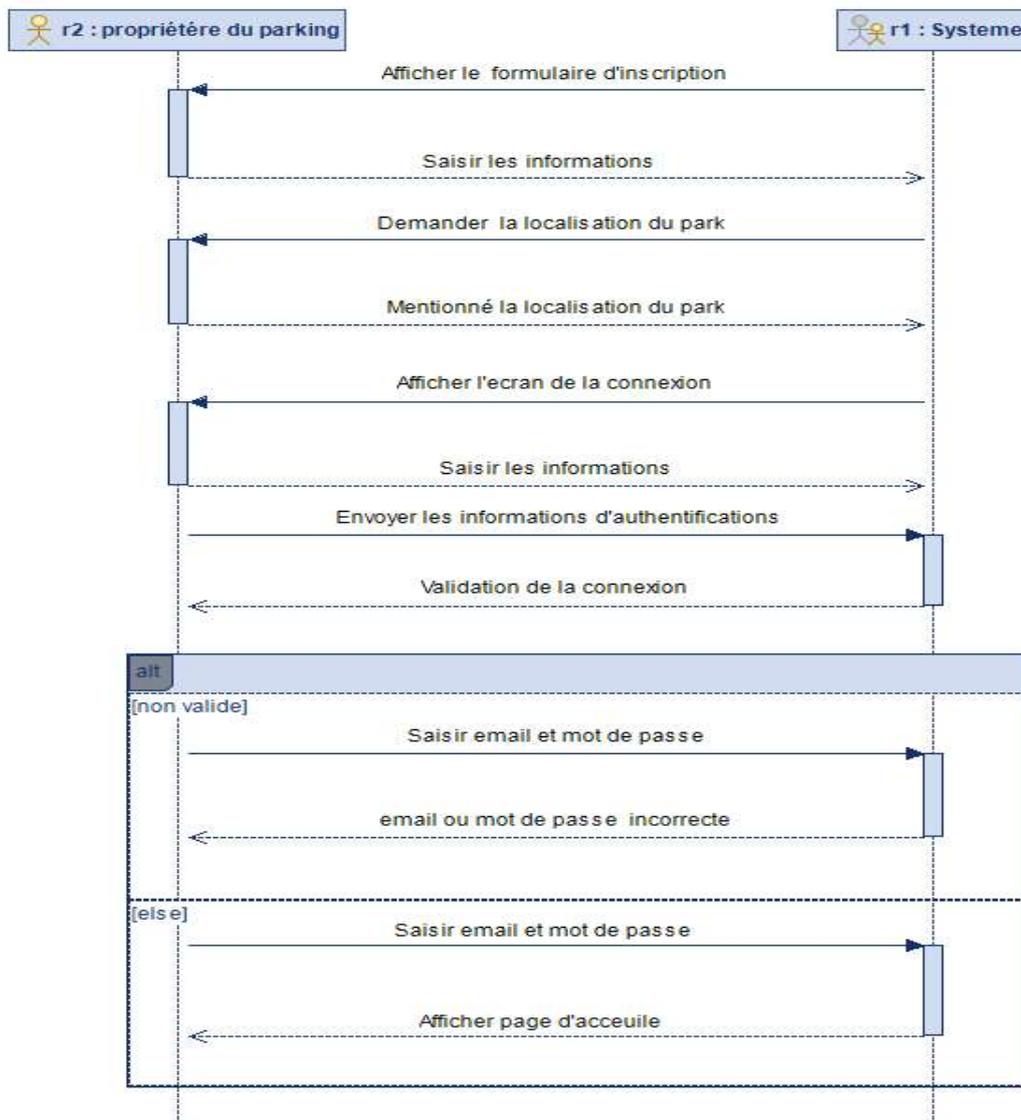


Figure 25 : Diagramme de séquence pour s'authentifier (propriétaire du Park).

➤ Diagramme de séquence gérer les informations de parking :

Le propriétaire du parking s'authentifie avec informations. Il a la possibilité de les modifier, et une fois les modifications effectuées dans le système, une confirmation sera envoyée.

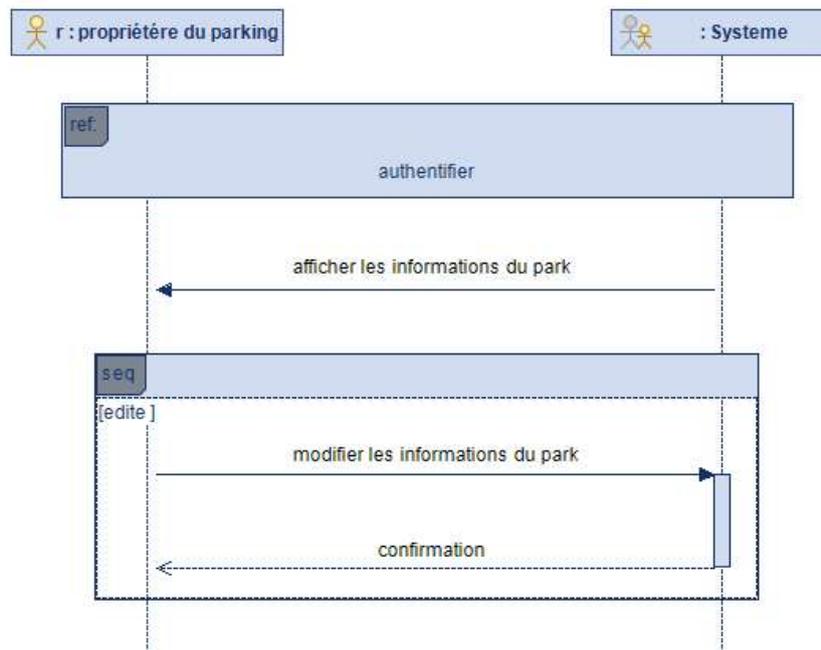


Figure 26 : Diagramme de séquence pour gérer les informations du parking.

➤ **Diagramme de séquence d'authentification (conducteur) :**

Le système affiche un formulaire d'inscription au conducteur. Ce dernier saisit les informations requises, puis le système affiche l'écran de connexion. Le conducteur saisit ses informations de connexion et les envoie pour s'authentifier.

Le système valide la connexion. Le conducteur saisit son adresse e-mail et son mot de passe, puis les envoie au système. Si l'e-mail ou le mot de passe est incorrect, le système répond en indiquant une erreur d'authentification. Le conducteur peut alors ressaisir son adresse e-mail et son mot de passe. Une fois que les informations ont été vérifiées, le système répond en affichant la page d'accueil.

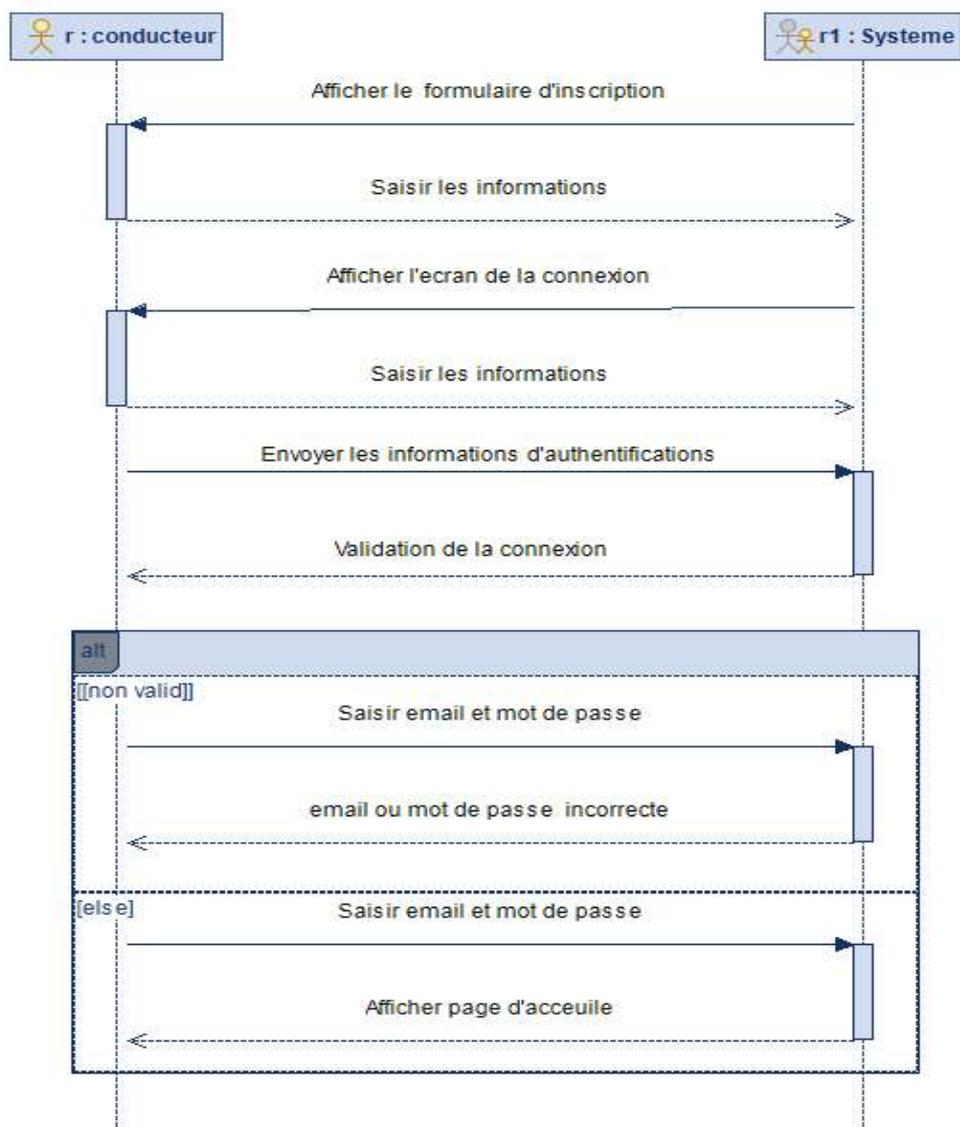


Figure 27: Diagramme de séquence d'authentification (conducteur).

➤ **Diagramme de séquence pour choisir une destination (conducteur) :**

Une fois le conducteur authentifié, il sélectionne sa destination dans le système. Le système affiche ensuite une liste des parcs filtrés. Le conducteur saisit son budget et la période souhaitée, puis le système filtre les parcs en fonction de ces deux critères. Le conducteur sélectionne ensuite un parc, et le système affiche les indications pour se rendre au parc. Si le parc choisi est plein, le système envoie également une notification au conducteur en lui proposant de choisir un nouveau parc de la liste des parcs affichée après sa mise à jour par le système. Le conducteur répond et le système affiche alors les indications pour se rendre à ce nouveau parc.

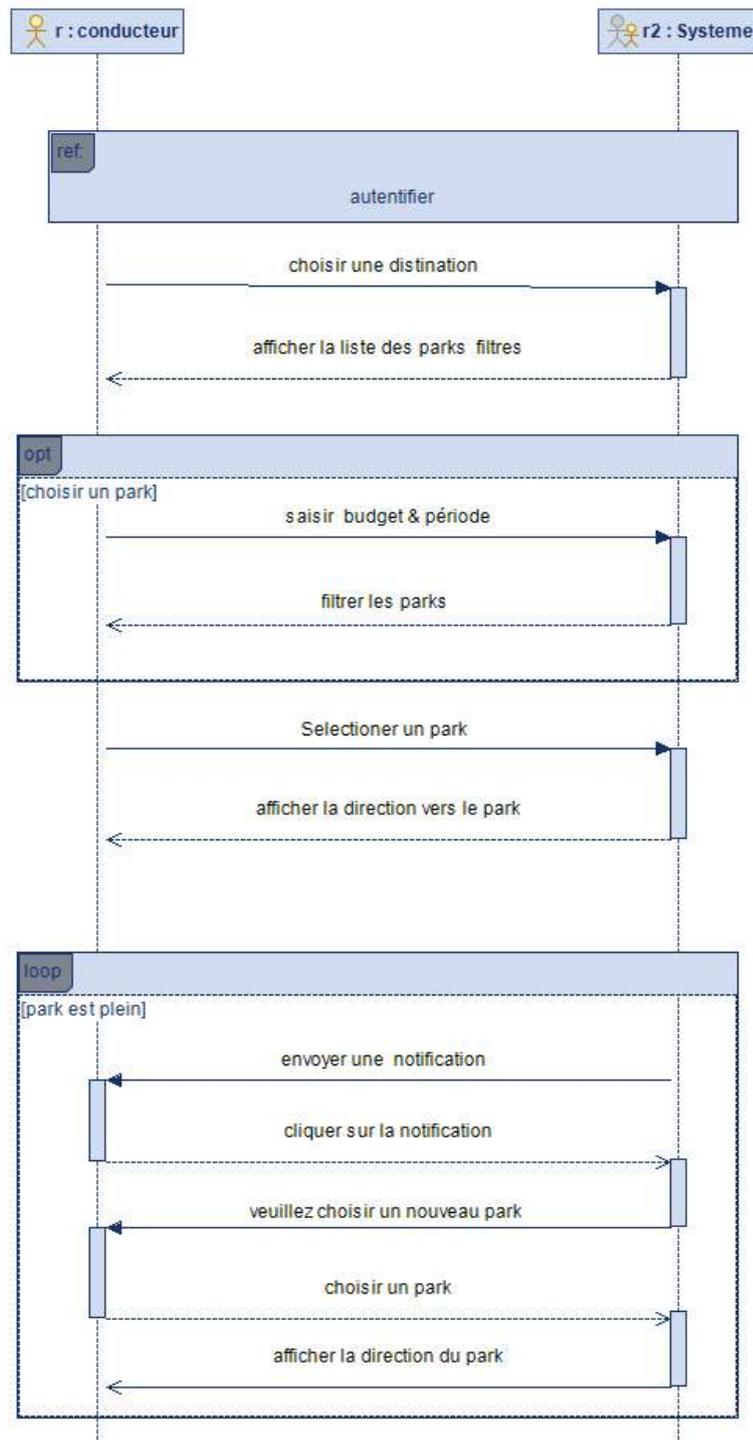


Figure 28 : Diagramme de séquence pour choisir une destination (conducteur).

➤ Diagramme de séquence choisir une destination (invité) :

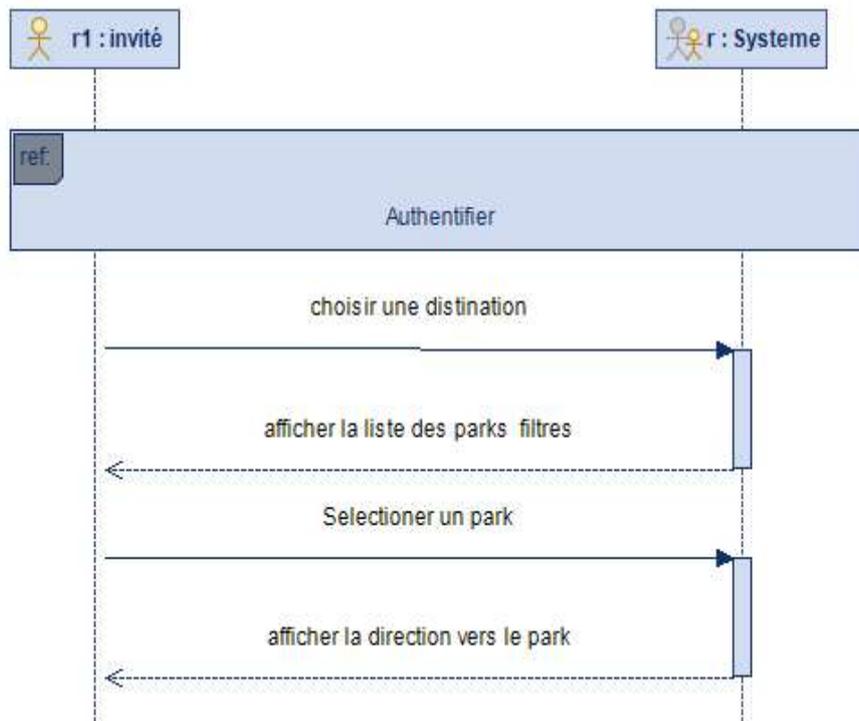


Figure 29 : Diagramme de séquence choisir une destination (invité).

➤ Diagramme de classes :

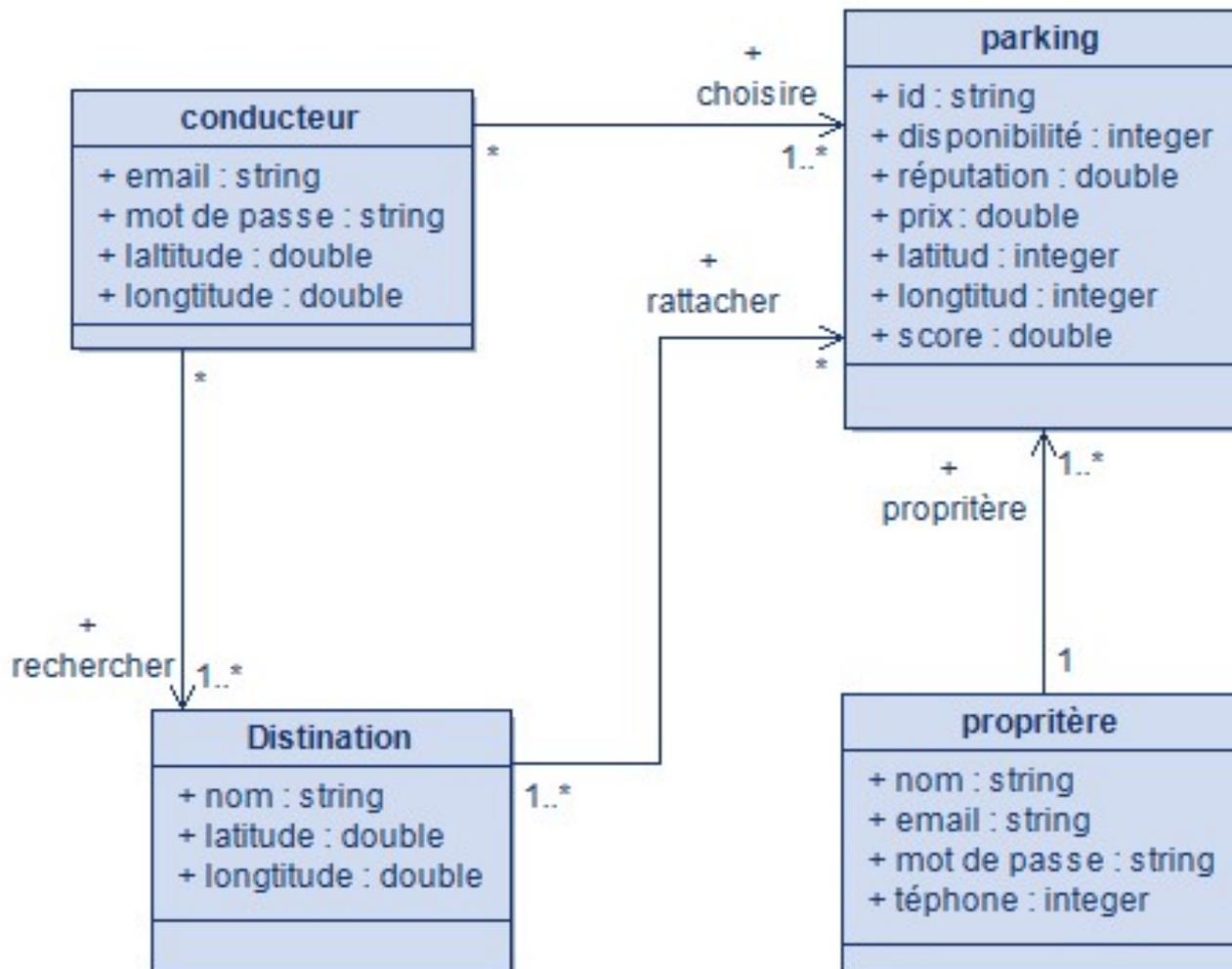


Figure 30 : Diagramme de classes.

Le diagramme de classe "Conducteur" représente les caractéristiques principales d'un conducteur. Il comprend les attributs suivants : "email" pour stocker l'adresse e-mail du conducteur, "mot de passe" pour enregistrer le mot de passe associé à son compte, "latitude" pour stocker la latitude de sa position actuelle, et "longitude" pour stocker la longitude correspondante.

La classe "Parking" représente les informations relatives à un parking. Elle comprend les attributs suivants : "id" pour l'identifiant unique du parking, "disponibilité" pour le nombre de places disponibles dans le parking, "réputation" pour la réputation du parking, "prix" pour le

tarif de stationnement, "latitude" et "longitude" pour les coordonnées géographiques du parking, et "score" pour évaluer la qualité globale du parking.

La classe "Destination" représente une destination spécifique avec les attributs "nom" pour le nom de la destination, "latitude" pour sa latitude, et "longitude" pour sa longitude.

Enfin, la classe "Propriétaire" représente les informations relatives au propriétaire d'un parking, avec les attributs "nom" pour le nom du propriétaire, "email" pour son adresse e-mail, "mot de passe" pour son mot de passe associé, et "téléphone" pour son numéro de téléphone.

➤ VIII. Scénarios des acteurs de système

○ VIII.1. Propriétaire de parc

➤ Les interfaces Utilisées :

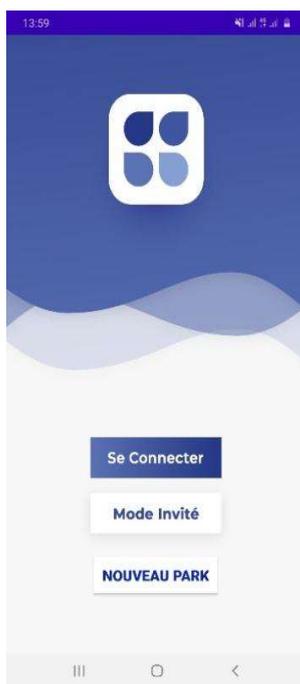


Figure 31: Interface inscription 1.



Figure 32: Interface d'accueil.



Figure 33: Interface d'inscription 2.



Figure 34: Interface de connexion (propriétaires de parc).



Figure 35: Interface information.

Scénario : inscription d’un nouveau parc :

Pour l’inscription d’un nouveau parc, il suffit de fournir les informations essentielles : nom, propriétaire, email, mot de passe, places disponibles, prix, numéro de téléphone et localisation.

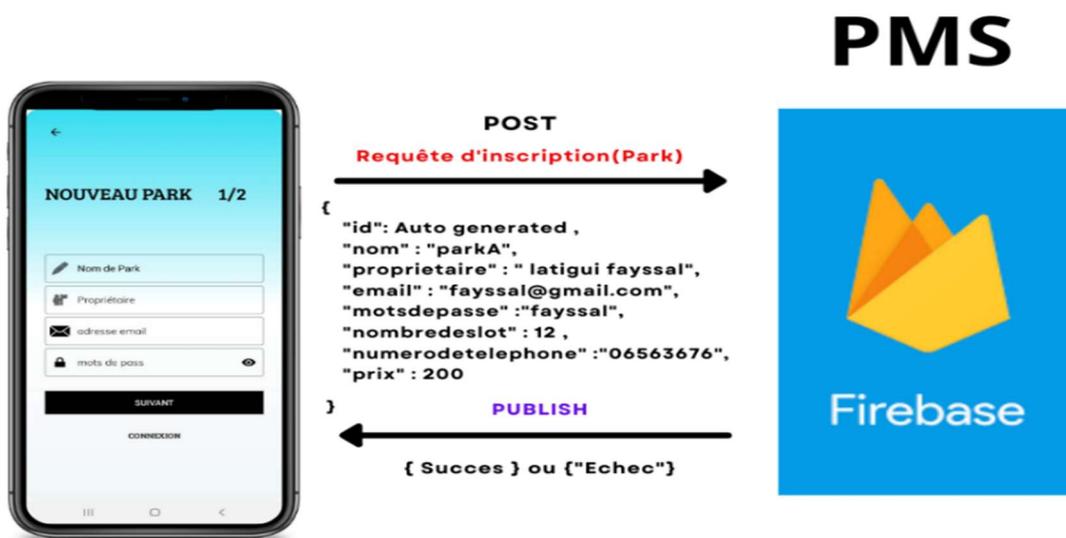


Figure 36 : Inscription d’un nouveau parc.

VIII.2 Conducteur

➤ Les interfaces Utilisées :

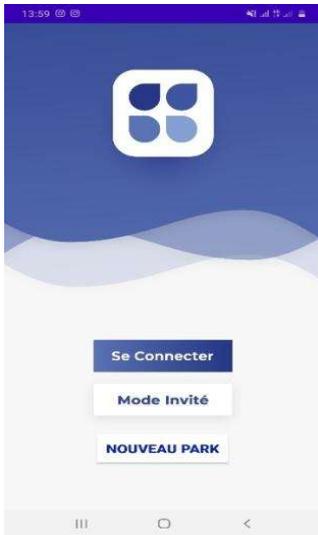


Figure 37: Interface page d'accueil.



Figure 38: Interface inscription conducteur.



Figure 39 : Interface connexion conducteur.

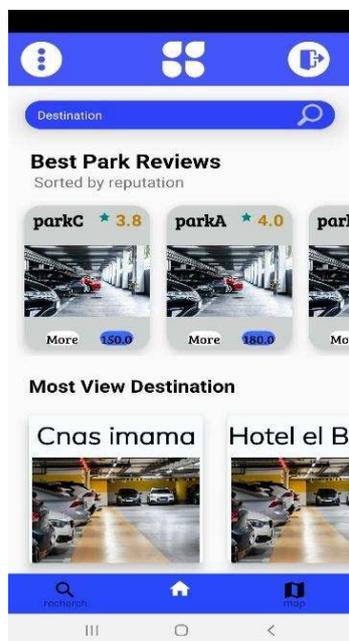


Figure 40 : Interface de home application.

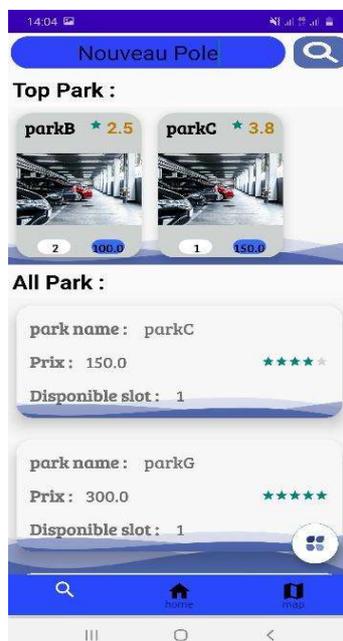


Figure 41: Interface de recherche parcs.

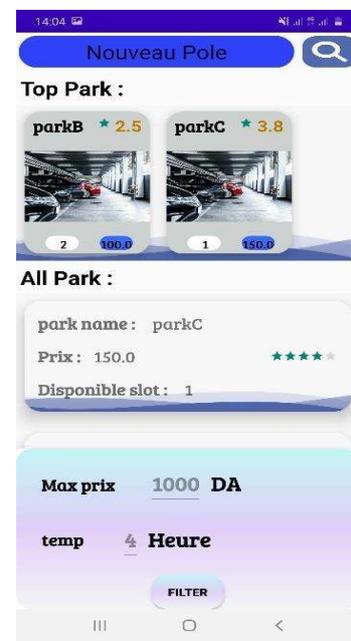


Figure 42 : Interface de filtre optionnel.



Figure 43: Interface MapParkB.

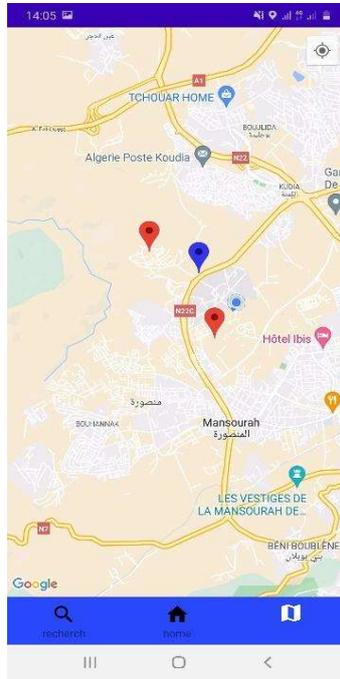


Figure 44: Interface parcs disponibles.

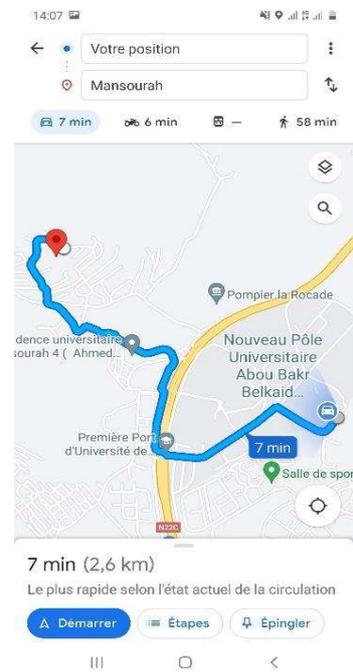


Figure 45: Interface guidage par Google Maps.

➤ Scénario : Inscription du conducteur :

- Le conducteur lance l'application.
- Il sélectionne l'option d'inscription.
- Le conducteur remplit les informations requises telles que son nom, son adresse e-mail et son mot de passe.
- Une fois les informations fournies, le conducteur valide l'inscription.

Le système enregistre les informations du conducteur et créer un compte utilisateur

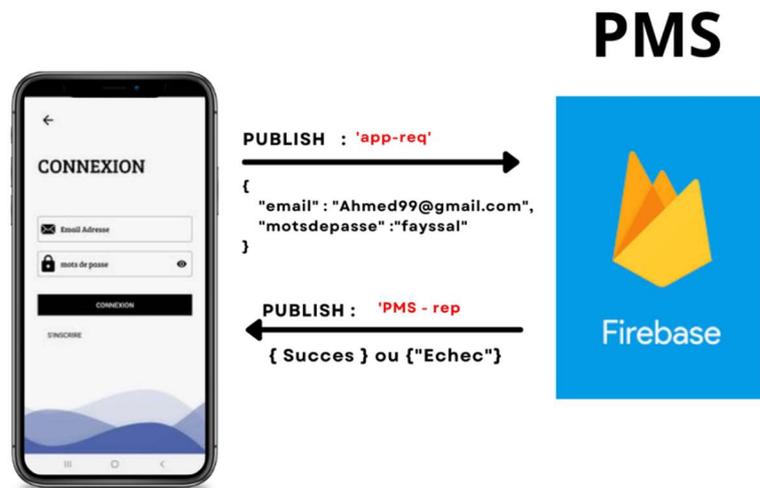


Figure 47 : Connexion conducteur.

➤ Scénario recherche d'une destination :

- Le conducteur se connecte à l'application.
- Il accède à l'option de recherche de destination.
- Le conducteur entre le nom de sa destination dans le champ de recherche.
- Le système utilise les données en temps réel pour afficher les parkings disponibles à proximité, triés par les meilleures options (prix, réputation, disponibilité de places).

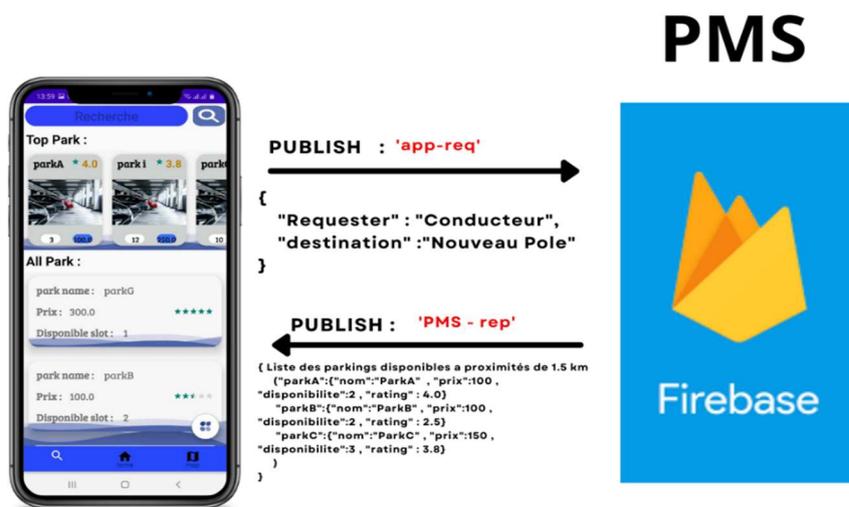


Figure 48: Recherche de destination.

➤ Scénario : Filtrage de la recherche (optionnel) :

- Le conducteur sélectionne l'option de filtrage de recherche.
- Il saisit son budget maximal dans un champ dédié.
- Il spécifie la période de temps pendant laquelle il souhaite utiliser le parking.
- Le système met à jour la liste des parkings disponibles en fonction du budget et de la période spécifiée.
- Le conducteur examine la liste filtrée des parkings disponibles répondant à ses critères de budget et de période.



Figure 49: Filtrage de la recherche (Optionnel).

➤ Scénario : Sélection d'un parc :

- Le conducteur examine et sélectionne un parking disponible en fonction de ses préférences (prix, emplacement, etc.) parmi la liste affichée.

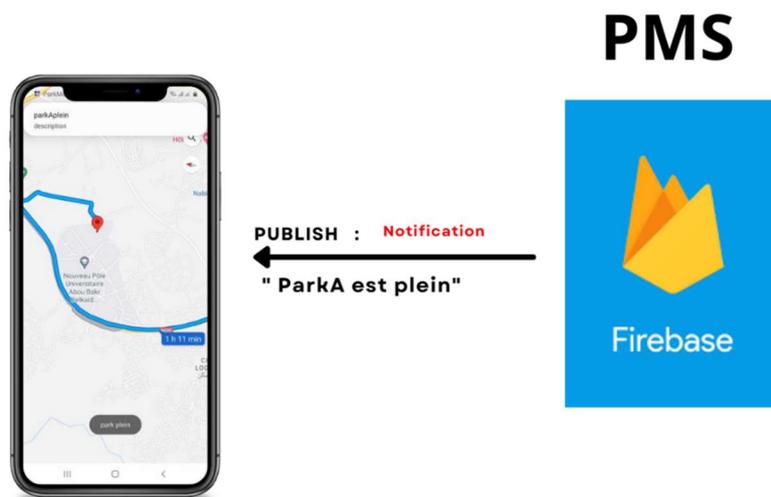


Figure 50 : Sélection d'un parc.

➤ **Scénario : Demande de direction vers le parking :**

Le conducteur choisit un parking en fonction de ses préférences, puis utilise l'option de demande de direction. L'application utilise l'API de Google Maps pour afficher un itinéraire détaillé de sa position actuelle au parking sélectionné, permettant ainsi au conducteur de naviguer facilement jusqu'à sa destination.



Figure 51: Demande de direction vers le parc choisi.

Le système vérifie en temps réel la disponibilité des places dans le parking sélectionné. Si le parking est plein, le conducteur reçoit une notification indiquant que le parking est plein.

Et il est redirigé vers la liste mise à jour des parkings disponibles. Le conducteur peut choisir un nouveau parking parmi les options proposées.



Figure 52 : Redirection vers la liste des parkings disponibles.

➤ IX. Conclusion

Dans ce chapitre, nous avons exploré les différents composants essentiels à la réalisation de notre projet de stationnement intelligent. Nous avons fourni une approche détaillée de l'algorithme multi critère utilisé pour le choix des places de stationnement. De plus, nous avons discuté des technologies et des applications mises en œuvre, tout en présentant des captures d'interface de l'application.

En résumé, ce chapitre a mis en évidence les éléments clés de notre projet. Nous avons examiné en détail les différents composants nécessaires à son fonctionnement, en mettant l'accent sur l'algorithme multi critère utilisé pour recommander les places de stationnement. De plus, nous avons présenté les technologies et les applications qui alimentent notre système, tout en illustrant son interface utilisateur à travers des captures d'écran. Ces éléments combinés constituent une base solide pour la réalisation de notre solution de stationnement intelligent.

Conclusion

Générale

Depuis les premiers temps de l'humanité, les êtres humains ont constamment cherché des moyens pour faciliter et accélérer l'accomplissement des tâches, tout en assurant la sécurité et l'efficacité. Cette quête perpétuelle découle de la nature ambitieuse de l'homme, qui aspire à des changements positifs dans tous les aspects de la vie, tout en suivant le développement technologique constant. L'objectif de ce projet était de créer un exemple concret de parking intelligent en utilisant un algorithme de choix multicritère.

Pour atteindre cette finalité, nous avons développé un système qui tire parti d'outils d'orientation au stationnement via une application mobile exploitant l'Internet des objets (IoT). Dans un premier temps, nous avons étudié la relation entre l'Internet des objets et le stationnement, puis nous avons exposé les dernières avancées en matière de parking intelligent. Par la suite, nous avons décrit les composants électroniques utilisés pour mettre en œuvre un système de parking intelligent. Par la suite, nous avons procédé à la conception et à la réalisation d'un prototype de parking intelligent.

Ce processus s'est divisé en trois maquettes distinctes, auxquelles s'est ajoutée le développement d'une application mobile en utilisant les langages Java et XML. En définitive, ce projet nous a été extrêmement bénéfique pour mettre en pratique nos connaissances, notamment dans les domaines de l'informatique et de l'Arduino. Il nous a permis de consolider nos connaissances aussi bien sur le plan théorique que pratique.

Au sein du centre I2E, nous avons également entrepris une étude économique pour évaluer la faisabilité et la rentabilité d'un système de stationnement intelligent. Cette approche innovante est résumée dans le Business Model Canvas (BMC) accompagnant ce mémoire. Ce dernier offre une vision globale des aspects clés du système de stationnement intelligent, incluant les coûts de mise en place, les canaux de distribution et les partenaires clés.

En somme, ce mémoire combine l'aspect technique d'un système de stationnement intelligent à une analyse économique visant à évaluer sa faisabilité et sa rentabilité.

- **Perspective du mémoire**

- Ajout d'un système de paiement en ligne pour les places de stationnement.
- Proposition d'un tableau de bord pour les gestionnaires de parkings, offrant une visualisation en temps réel des informations et des statistiques.
- Utilisation du deep learning pour prédire les places de stationnement disponibles.
- Utilisation de caméras intelligentes basées sur l'intelligence artificielle au lieu de capteurs de présence pour les véhicules, ce qui réduit les coûts.

Références

[1]: Hassija, V., et al., A survey on IoT security: application areas, security threats, and solution architectures. IEEE Access, 2019. 7 : p. 82721-82743.

[2] :Terir, K., Gestion de la confidentialité des données pour les dispositifs IOT (Internet of Things), 2020, University of Jijel.

[3] :<https://chipwired.com/analyze-data-using-arduino/>

[4] :<https://wikimemoires.net/2019/09/domaines-d-applications-de-l-iot/>

[5] : <https://praedictia.com/page/internet-des-objets/que-peut-faire-linternet-des-objets.html>

[6] :<https://www.iotforall.com/use-case/environmental-chemical-leak-detection>

[7] :<https://www.lemagit.fr/conseil/Microservices-comprendre-les-modeles-de-decouverte-de-services>

[8] :<https://iot.stackexchange.com/questions/1246/whats-the-difference-between-the-internet-of-things-and-the-traditional-interne>

[9] : https://upload.wikimedia.org/wikipedia/commons/4/40/Diversité_IOT.jpg

[10] :<https://www.digora.com/fr/blog/quest-ce-que-liot-et-pourquoi-mener-une-strategie-diot>

[11] :<https://asialyst.com/fr/2016/02/01/asiie-des-villes-intelligentes-quelles-reponses-au-defi-urbain/>

[12]: Oxford University Press (OUP). parking. Lexico.Com. <https://www.lexico.com/definition/parking> [05/07/2022]

[13]: WIKIPEDIA CONTRIBUTORS. (2022, 5 Juillet). PARKING. WIKIPEDIA. <https://en.wikipedia.org/wiki/Parking> [05/07/2022]

[14]: Karsten, J. (2019, 20 mars). What is Smart Parking? Parkeagle. <https://www.parkeagle.com/2018/05/12/what-is-smart-parking/> [20/05/2022]

[15]: Evdokimov, S., Fabian, B., Kunz, S., & Schoenemann, N. (2010, June). Comparison of discovery service architectures for the internet of things. In 2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (pp. 237-244). IEEE. [21/06/2022]

[16]: Paidi, V., Fleyeh, H., Håkansson, J., & Nyberg, R. G. (2018). Smart parking sensors, technologies and applications for open parking lots: a review. IET Intelligent Transport Systems, 12(8), 735-741. <https://doi.org/10.1049/ietits.2017.0406> [21/05/2022]

[17]: Nieves, J. M. (2022, 26 avril). How to choose a Smart Parking sensor? ThingPark Market. <https://market.thingpark.com/blog/post/how-to-choose-smart-parkingsensor> [22/05/2022]

[18]: <https://fr.wikipedia.org/wiki/NodeMCU>

[19]: Dimensionnel, l.d. Servo SG90, Micro Servo 9G. 2020 ; Available from : <https://ledisrupteurdimensionnel.com/>.

[20]: <https://mataucarre.fr/index.php/2017/05/24/capteur-de-proximite-infra-rouge-fc-51-arduino/>

[21]: <https://www.javatpoint.com/arduino-blinking-an-led>

[22]: <https://arduino.blaisepascal.fr/capteur-de-distance-a-ultrasons/>

[23]: <https://arduino.blaisepascal.fr/presentation/logiciel/>

[24]: <https://www.boryl.fr/glossaire/firebase/>

[25]: <https://www.techno-science.net/glossaire-definition/Adobe-Photoshop.html>

[26]: <http://www.mosaique-info.fr/glossaire-web-referencement-infographie-multimedia-informatique/i-glossaire-informatique-et-multimedia/231-illustrator-definition.html>

[27]: <https://developer.android.com/studio/intro>

[28] :https://www.java.com/fr/download/help/whatis_java.html

[29] :<https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203615-xml-extensible-markup-language-definition-traduction/>

[30]:Zaslofsky, A., & Jayaraman, P. P. (2015). Discovery in the Internet of Things. Ubiquity, 2015(October). <https://doi.org/10.1145/2822529> [21/06/2022]

[31]:Evdokimov, S., Fabian, B., Kunz, S., & Schoenemann, N. (2010, June). Comparison of discovery service architectures for the internet of things. In 2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (pp. 237-244). IEEE. [21/06/2022]

[32]:Aziez, M., Benharzallah, S., & Bennoui, H. (2017). A Comparative Analysis of Service Discovery Approaches for the Internet of Things. International Research Journal of Electronics and Computer Engineering, 3(1). <https://doi.org/10.24178/irjece.2017.3.1.17> [21/06/2022]

[33]:Pourghebleh, B., Hayyolalam, V. & Aghaeianvigh, A. Service discovery in the Internet of Things: review of current trends and research challenges. Wireless Netw 26, 5371–5391 (2020) [21/06/2022]

[34] :Zorgati, H., Djemaa, R. B., & Amor, I. A. B. (2019, October). Service discovery techniques in Internet of Things: a survey. In 2019 IEEE international conference on systems, man and cybernetics (SMC) (pp. 1720-1725). IEEE. [21/06/2022]