

Faculté des technologies de l'information et de la communication

Département informatique appliquée



RAPPORT PROJET TUTORE

Mo Belle

Réalisé par : Nathan Carlinot RANDRIAMIHAJA

Sous la direction du superviseur : Khadimoullah Ramoth

Année scolaire : 2023 – 2024

Sommaire

Introduction générale.....	1
Chapitre 1 : Présentation du cadre de projet.....	2
1- Introduction.....	2
2- Environnement du projet	2
3- Recherche préalable.....	4
4- Description de l'état de lieu	5
5- Planning prévisionnel	6
6- Conclusion	7
Chapitre 2 : Analyse des applications et logiciels.....	8
1- Introduction.....	8
2- Environnement Logiciel.....	8
3- Langage de programmation	11
4- Outils	12
5- Composant Électronique	14
6- Conclusion	18
Chapitre 3 : Spécification des besoins.....	19
1- Introduction.....	19
2- Spécification des besoins fonctionnel	19
3- Spécification des besoins non fonctionnel	22
4- Présentation des cas d'utilisations.....	24
5- Conclusion	30
Chapitre 4 : Conception du système	31
1- Introduction.....	31
2- Modélisation dynamique.....	31
3- Modélisation statique	45
4- Conclusion	47
Chapitre 5 : Réalisation du système	48
1- Introduction.....	48
2- Wokwi.....	48
3- Principales interfaces graphiques.....	48
4- Conclusion	56

Chapitre 6 : Evaluation	57
1- Introduction.....	57
2- Problème rencontrer	57
3- Amélioration dans les futures	57
4- Conclusion	58
Conclusion générale	59

Introduction générale

Chapitre 1 : Présentation du cadre de projet

1- Introduction

Ce projet "Mo Belle" est une initiative innovante visant à automatiser la gestion des poubelles grâce à une application mobile et un système IoT. Il introduit l'environnement du projet, les technologies employées, et les objectifs poursuivis. Ce document présente les recherches préalables effectuées pour comprendre les besoins des utilisateurs, ainsi que la description de l'état des lieux avant la mise en œuvre du projet. Un planning prévisionnel des différentes étapes du projet est également fourni pour illustrer l'organisation et les délais de réalisation.

2- Environnement du projet

Mo Belle est un projet innovant visant à créer une application mobile et un système IoT pour la gestion automatique des poubelles. L'objectif est de proposer une solution intelligente et écologique pour la gestion des déchets urbains, facilitant ainsi le travail des services de propreté et améliorant la qualité de vie des citoyens. Mo Belle utilise des capteurs IoT pour surveiller les niveaux de remplissage des poubelles et une application mobile pour permettre aux utilisateurs de consulter l'état des poubelles en temps réel.

Spécificités du projet:

1. Surveillance des Poubelles:

- **Capteurs de Niveau de Remplissage** : Chaque poubelle est équipée de capteurs (ultrason et infrarouge) pour mesurer le niveau de remplissage. Les capteurs envoient des données en temps réel à la plateforme cloud.
- **Capteurs de Poids** : Les poubelles sont également équipées de capteurs de poids HX711 pour mesurer le poids des déchets.

2. Application Mobile:

- **Authentification** : Les utilisateurs (administrateurs et personnel de nettoyage) peuvent s'authentifier pour accéder à l'application.
- **Suivi en Temps Réel** : Les utilisateurs peuvent voir en temps réel le niveau de remplissage et le poids des poubelles.
- **Notifications** : L'application envoie des notifications lorsque les poubelles sont pleines et doivent être vidées.
- **Historique des Données** : Les utilisateurs peuvent consulter l'historique des niveaux de remplissage et des poids pour chaque poubelle.

3. Interface Administrateur:

- **Gestion des Poubelles** : Les administrateurs peuvent ajouter, modifier ou supprimer des poubelles depuis l'application.
- **Rapports et Statistiques** : Accès à des rapports détaillés sur les performances des poubelles et les tendances de remplissage.
- **Gestion du Personnel** : Les administrateurs peuvent gérer les comptes des membres du personnel et leurs accès.

4. Fonctionnalités IoT:

- **Connectivité** : Les capteurs envoient des données via Wi-Fi ou réseau cellulaire au serveur cloud.
- **Traitement des Données** : Les données collectées sont traitées et analysées pour fournir des insights utiles.
- **Intégration avec l'Application Mobile** : Les données traitées sont intégrées à l'application mobile pour une visualisation facile.

5. Sécurité et Confidentialité:

- **Sécurité des Données** : Les données transmises sont sécurisées via des protocoles de cryptage.
- **Accès Sécurisé** : L'accès à l'application et aux données est protégé par des méthodes d'authentification robustes.

Technologies Utilisées:

- **Matériel** : Capteurs infrarouges, capteurs ultrasoniques, capteurs de poids HX711, modules ESP32.
- **Logiciel** : Flutter pour l'application mobile, Firebase pour la gestion des données et l'authentification, plateformes cloud pour le traitement et le stockage des données IoT.
- **Communication** : Protocole MQTT pour la communication IoT, API REST pour l'intégration avec l'application mobile.

Objectifs du Projet:

- **Améliorer l'Efficacité** : Faciliter la gestion des poubelles en fournissant des données en temps réel sur leur état de remplissage.
- **Réduire l'Impact Environnemental** : Optimiser les tournées de ramassage des déchets pour réduire les émissions de CO2.
- **Assurer la Propreté** : Garantir que les poubelles ne débordent pas en alertant les services de nettoyage à temps.

- **Améliorer la Qualité de Vie** : Offrir aux citoyens un environnement plus propre et plus sain.

3- Recherche préalable

Avant de commencer à développer "Mo Belle", j'ai réalisé des recherches approfondies pour mieux comprendre les besoins des services de gestion des déchets et les défis liés à la gestion des poubelles dans les environnements domestiques. Mes recherches ont inclus les étapes suivantes :

1. Étude des Problèmes Actuels :

- **Identification des Problèmes de Débordement** : J'ai étudié les situations où les poubelles débordent fréquemment, causant des nuisances visuelles et olfactives. Cela m'a permis de comprendre l'importance de la surveillance en temps réel.
- **Analyse des Itinéraires de Ramassage** : J'ai examiné comment les itinéraires de ramassage des déchets sont planifiés actuellement et les inefficacités potentielles, telles que les trajets inutiles vers des poubelles non remplies.

2. Analyse des Solutions Existant sur le Marché :

- **Évaluation des Technologies IoT** : J'ai recherché les solutions existantes utilisant des capteurs IoT pour surveiller le niveau de remplissage des poubelles. Cela m'a permis de déterminer les capteurs les plus efficaces et les plus fiables, tels que les capteurs ultrasoniques et les capteurs de poids HX711.
- **Étude des Interfaces Utilisateurs** : J'ai analysé les interfaces des applications mobiles et des systèmes de gestion de déchets existants pour identifier les meilleures pratiques et les opportunités d'amélioration.

3. Consultation avec les Parties Prenantes :

- **Entretiens avec les Services de Propreté** : J'ai mené des entretiens avec les responsables des services de propreté pour comprendre leurs besoins et défis quotidiens. Cela a aidé à définir les fonctionnalités clés de l'application Mo Belle, telles que les notifications en temps réel et les rapports de performance.
- **Feedback des Citoyens** : J'ai recueilli des avis auprès des citoyens sur leurs expériences avec les poubelles publiques. Cette étape a été cruciale pour concevoir une solution qui répond aux attentes des utilisateurs finaux.

4. Analyse des Tendances Technologiques :

- **Exploration des Avancées IoT** : J'ai exploré les tendances technologiques récentes dans le domaine de l'Internet des objets (IoT) pour intégrer les innovations les plus récentes dans le projet.
- **Étude des Protocoles de Communication** : J'ai étudié les différents protocoles de communication, tels que MQTT et REST API, pour garantir une transmission fiable et sécurisée des données des capteurs à l'application mobile.

4- Description de l'état de lieu

Le projet "Mo Belle" vise à développer un système de gestion de poubelle automatique pour les maisons, en intégrant des technologies avancées de l'Internet des Objets (IoT) et des solutions mobiles innovantes. Voici un aperçu détaillé de l'état des lieux actuel, mettant en lumière les défis et les opportunités associés à ce projet.

Contexte et Justification :

Les ménages génèrent quotidiennement une quantité considérable de déchets, nécessitant une gestion efficace pour maintenir la propreté et l'hygiène. Les systèmes de gestion des déchets actuels reposent principalement sur des pratiques manuelles, qui peuvent être inefficaces et fastidieuses. Le projet "Mo Belle" s'inscrit dans une démarche d'innovation visant à automatiser la gestion des poubelles domestiques pour offrir une solution plus pratique et plus respectueuse de l'environnement.

Objectifs du Projet :

- **Automatisation de la Gestion des Poubelles** : Développer un système qui surveille automatiquement le niveau de remplissage des poubelles à l'aide de capteurs IoT (ultrasons, capteurs de poids HX711, etc.) et envoie des notifications lorsque les poubelles doivent être vidées.
- **Optimisation du Tri des Déchets** : Intégrer des fonctionnalités de tri automatique pour séparer les différents types de déchets (organique, plastique, papier, etc.) dès leur dépôt.
- **Interface Utilisateur Intuitive** : Créer une application mobile conviviale permettant aux utilisateurs de suivre le statut de leurs poubelles en temps réel, de recevoir des alertes et de consulter des rapports sur leur production de déchets.

Technologies Utilisées :

- **Capteurs IoT** : Utilisation de capteurs ultrasoniques pour mesurer le niveau de remplissage des poubelles et de capteurs de poids HX711 pour peser les déchets.
- **Communication sans Fil** : Mise en place de modules WiFi pour transmettre les données des capteurs à l'application mobile.

- **Application Mobile** : Développement d'une application mobile pour Android et iOS, offrant une interface utilisateur intuitive et des fonctionnalités de gestion des poubelles.

Défis Identifiés :

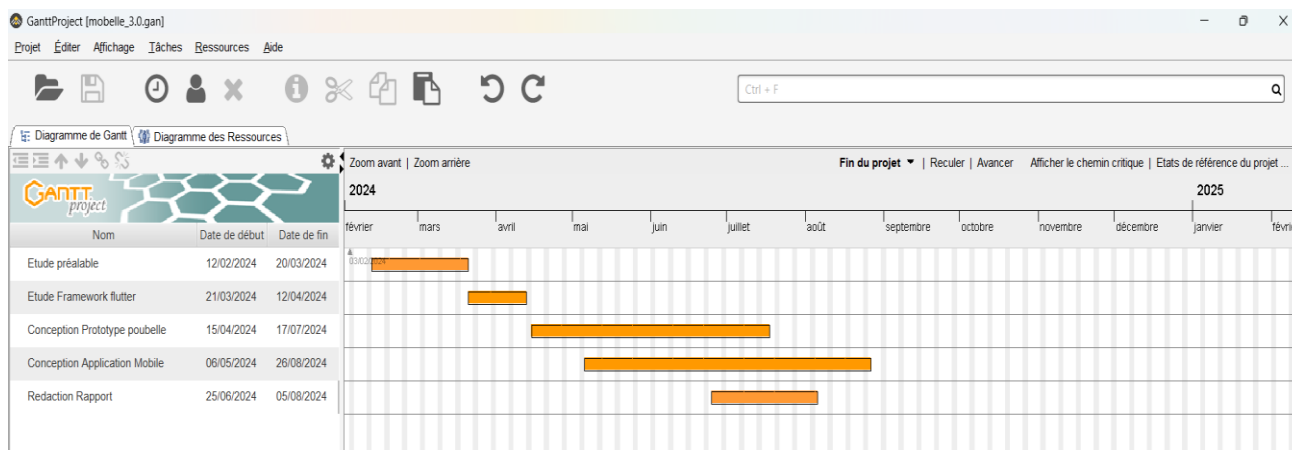
- **Fiabilité des Capteurs** : Assurer la précision et la fiabilité des capteurs IoT dans différentes conditions environnementales.
- **Gestion de l'Énergie** : Optimiser la consommation d'énergie des modules IoT pour garantir une autonomie suffisante des appareils.
- **Sécurité des Données** : Mettre en place des mesures de sécurité robustes pour protéger les données des utilisateurs et garantir la confidentialité de leurs informations personnelles.

Opportunités :

- **Amélioration de l'Hygiène** : Réduire les débordements de poubelles et améliorer l'hygiène domestique grâce à une surveillance en temps réel.
- **Engagement Écologique** : Encourager les pratiques de tri sélectif et de recyclage grâce à des fonctionnalités de tri automatique et des rapports de suivi.
- **Confort Utilisateur** : Offrir une solution pratique et sans tracas pour la gestion quotidienne des déchets, facilitant la vie des utilisateurs.

5- Planning prévisionnel

Le projet "Mo Belle" a suivi une planification minutieuse avec des étapes clés. Il a débuté le 12 février 2024 avec une étude préalable qui a duré 20 jours, suivie par une étude du framework Flutter commençant le 21 mars 2024 et durant 23 jours. Ensuite, la conception du prototype de la poubelle a débuté le 15 avril 2024 et s'est étalée sur 94 jours. Par la suite, la conception de l'application mobile a démarré le 6 mai 2024 et a pris 112 jours. Enfin, la rédaction du rapport a commencé le 25 juin 2024 et a duré 62 jours. Tout au long de ces étapes, le projet a avancé de manière fluide et professionnelle.



6- Conclusion

En résumé, ce projet de gestion automatisée des poubelles à travers l'application "Mo Belle" constitue une avancée significative vers des solutions intelligentes et durables pour la gestion des déchets. Les recherches effectuées et la compréhension des besoins des utilisateurs ont jeté les bases solides pour la réussite du projet. L'analyse de l'état des lieux a permis d'identifier les opportunités d'amélioration. Le planning détaillé assure une organisation rigoureuse et une exécution efficace. Ce projet promet d'améliorer considérablement l'efficacité de la gestion des déchets urbains.

Chapitre 2 : Analyse des applications et logiciels

1- Introduction

Dans ce chapitre, nous explorerons en profondeur la conception et l'architecture du système de gestion des poubelles intelligentes "Mo Belle". Nous détaillerons les choix technologiques, les diagrammes UML, ainsi que les différentes interactions entre les composants du système. Cette analyse permettra de comprendre les fondations techniques du projet. Enfin, nous aborderons les méthodologies adoptées pour garantir la robustesse et l'efficacité du système.

2- Environnement Logiciel

2-1- Android Studio

Android Studio, édité par Google, se présente comme l'environnement de développement intégré (IDE) officiel pour la création d'applications sur le système d'exploitation Android. Fondé sur la plateforme IntelliJ IDEA de JetBrains, cet outil est spécifiquement pensé pour répondre aux besoins des développeurs Android. Disponible en téléchargement sur les principales plateformes telles que Windows, macOS et Linux, il offre une interface conviviale et des fonctionnalités avancées pour faciliter le processus de développement. En remplaçant les anciens outils comme Eclipse (E-ADT), Android Studio s'impose comme le choix privilégié des développeurs pour concevoir des applications natives optimisées pour l'écosystème Android. Grâce à ses mises à jour régulières et à son intégration étroite avec les services Google, Android Studio reste à la pointe de l'innovation dans le domaine du développement d'applications mobiles.

android studio



2-2- Le SDK (Software Development Kit)

Le Kit de développement logiciel Android (Android SDK) constitue une boîte à outils essentielle pour la création, la compilation et le déploiement d'applications Android. Principalement basés sur des commandes en ligne, ces outils englobent diverses fonctionnalités cruciales pour les développeurs. Ils incluent les bibliothèques Java, fondamentales pour la création de logiciels, ainsi que les outils de mise en boîte indispensables au processus de développement. L'Android Virtual Device (AVD) offre quant à lui un émulateur de tablettes permettant de tester les applications dans un environnement virtuel. Parmi ces outils figure également le SDK Manager, une application dédiée à la gestion du SDK, permettant de choisir et de mettre à jour les composants nécessaires au développement. Grâce à ces fonctionnalités et à leur intégration cohérente, le SDK Android reste un pilier incontournable pour les développeurs souhaitant créer des applications pour la plateforme Android.



2-3- Firebase

Firebase est une plateforme de développement d'applications mobiles et web, appartenant à Google, qui offre une multitude de services et d'outils pour faciliter le développement, la croissance et la monétisation des applications. Son rôle est de fournir une infrastructure robuste et évolutive, notamment des bases de données en temps réel, des services d'authentification utilisateur, des solutions d'hébergement, des analyses d'utilisation et bien plus encore. Grâce à Firebase, les développeurs peuvent créer des applications de haute qualité plus rapidement en utilisant des fonctionnalités prêtes à l'emploi, réduisant ainsi le temps de développement et les coûts associés. De plus, Firebase propose des outils de suivi et de gestion

de la performance des applications, permettant aux développeurs de prendre des décisions éclairées pour améliorer l'expérience utilisateur et maximiser la rentabilité de leurs applications. En résumé, Firebase joue un rôle essentiel en offrant une infrastructure cloud complète et des fonctionnalités clés pour simplifier le développement et la gestion d'applications mobiles et web.



2-4- Flutter Framework

Flutter est un framework open-source développé par Google, conçu pour faciliter la création d'interfaces utilisateur multiplateformes, notamment pour les applications mobiles iOS, Android et web, à partir d'un seul code source. Son rôle principal est de permettre aux développeurs de construire des applications dynamiques et réactives avec une seule base de code, ce qui réduit les coûts de développement et accélère le processus de création. En utilisant le langage de programmation Dart, Flutter offre des fonctionnalités avancées telles que le hot reload, permettant des modifications en temps réel et une expérience de développement fluide. Grâce à ses widgets personnalisables et à sa performance élevée, Flutter est devenu un choix populaire pour le développement d'applications modernes et est largement utilisé par les développeurs pour créer des interfaces utilisateur esthétiques et fonctionnelles sur différentes plateformes.



2-5- Star Uml

UML, ou Unified Modeling Language, est un langage de modélisation graphique utilisé dans le domaine du développement logiciel pour représenter visuellement les différents aspects d'un système. Son rôle principal est de fournir un ensemble de notations standardisées pour décrire la structure, le comportement, les interactions et les contraintes d'un système logiciel. En utilisant des diagrammes comme les diagrammes de classes, de séquence et d'activité, UML permet aux développeurs de communiquer efficacement entre eux et avec les parties prenantes du projet. En outre, UML favorise la compréhension, la documentation et la conception des systèmes logiciels, ce qui facilite le processus de développement, la maintenance et l'évolution des applications. En résumé, UML joue un rôle crucial en tant que langage de modélisation standardisé pour la conception et la documentation des systèmes logiciels.



3- Langage de programmation

3-1- Dart

Dart, conçu par Lars Bak et Kasper Lund et soutenu par Google, est un langage de programmation polyvalent utilisé pour créer une variété d'applications, y compris des applications web, mobiles, serveur et de bureau. Il est entièrement orienté objet, reposant sur des classes, avec une syntaxe familière de style C et un système de gestion automatique de la

mémoire. La compilation de Dart peut se faire en code machine natif, JavaScript ou WebAssembly, offrant ainsi une grande flexibilité pour le déploiement. Ses fonctionnalités avancées incluent la prise en charge des interfaces, des mixins, des classes abstraites, des génériques réifiés et l'inférence de type, offrant aux développeurs des outils puissants pour concevoir des applications robustes et évolutives dans divers contextes de développement.



4- Outils

4-1- Visual Studio Code

Visual Studio Code, un éditeur de code source robuste, fonctionne sur les systèmes d'exploitation Windows, macOS et Linux, offrant ainsi une portabilité maximale. Doté d'une intégration native pour JavaScript, TypeScript et Node.js, il propose également un large éventail d'extensions pour d'autres langages et environnements d'exécution, comme C++, C#, Java, Python, PHP, Go et .NET. Grâce à sa légèreté et à sa puissance, Visual Studio Code est devenu un choix populaire parmi les développeurs pour leur workflow de développement.



4-2- Git

Git est un système de contrôle de version distribué, offrant un suivi précis des modifications apportées au code source lors du développement logiciel. Son rôle essentiel est de permettre aux équipes de collaborer efficacement sur des projets de toutes tailles, en facilitant la gestion des versions du code, la création de branches pour les fonctionnalités en cours de développement, et la fusion des modifications en toute sécurité. Grâce à sa flexibilité, sa robustesse et sa facilité d'utilisation, Git est devenu un pilier de l'industrie du développement logiciel moderne, favorisant un workflow collaboratif fluide et une intégration continue efficace.



4-3- GitHub

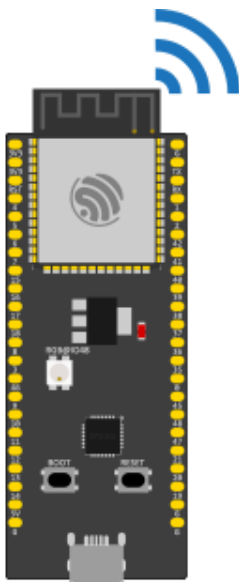
GitHub, une plateforme cloud, offre des services pour le développement logiciel et le contrôle de version via Git, permettant aux développeurs de gérer leur code efficacement. En plus du contrôle de version distribué de Git, GitHub propose des fonctionnalités avancées telles que le contrôle d'accès, le suivi des bogues, les demandes de fonctionnalités, la gestion des tâches et l'intégration continue. Fondée en Californie, GitHub est devenue une filiale de Microsoft en 2018, consolidant ainsi son influence dans l'écosystème du développement logiciel.



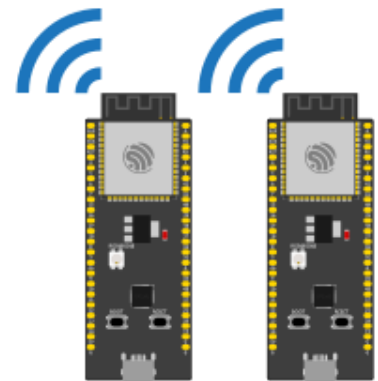
5- Composant Électronique

5-1- Arduino Esp32

L'Arduino Nano ESP32 intègre le NORA-W106, un module équipé de la puce ESP32-S3, offrant une connectivité Wi-Fi® et Bluetooth® (version 5.0 et ultérieure), ce qui en fait une solution de choix pour le développement d'applications IoT. Son format Nano, largement adopté, assure une compatibilité étendue avec une variété d'accessoires matériels. Avec cette combinaison de fonctionnalités avancées et de facteur de forme compact, l'Arduino Nano ESP32 répond aux besoins des développeurs en matière de conception et de déploiement d'applications IoT hautement performantes.



ESP32 as Wi-Fi Access Point (AP)



ESP32' as Wi-Fi Station (STA)

5-2- Servo Moteur

Le servo-moteur, une variété de moteur électrique, est couramment employé dans le modélisme, notamment pour réguler la direction des véhicules télécommandés. Ce dispositif permet de maintenir l'axe dans une position donnée, tandis que son angle peut être ajusté entre 0 et 180 degrés en réponse à un signal spécifique. Son utilisation offre un contrôle précis et fiable, permettant aux amateurs de modélisme de réaliser des mouvements fluides et précis dans leurs créations. Cette technologie est appréciée pour sa polyvalence et sa capacité à répondre aux exigences de diverses applications, allant des loisirs aux projets industriels.



Composition d'un servo-moteur

Le servomoteur est composé de plusieurs éléments visibles et invisible :

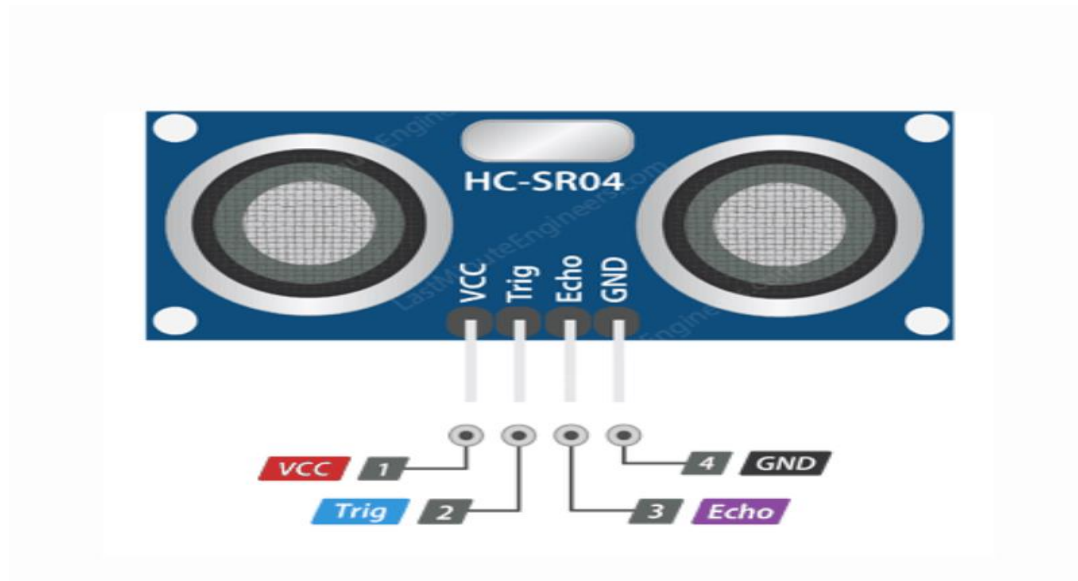
- Un moteur à courant continu constitue la première composante essentielle du servomoteur.
- Des engrenages, en plastique ou en métal, forment un réducteur qui permet de réduire la vitesse de rotation du moteur.
- Un capteur de position de l'angle d'orientation de l'axe, généralement un potentiomètre, assure la détection précise de la position de l'axe.
- Une carte électronique dédiée contrôle la position de l'axe en fonction des signaux reçus et pilote le moteur à courant continu en conséquence.
- Trois fils sont présents pour la connexion électrique et la transmission des signaux de contrôle.
- L'axe de rotation supporte un accessoire en plastique ou en métal, fournissant la sortie mécanique du servomoteur.
- Enfin, un boîtier protège l'ensemble des composants, assurant la durabilité et la sécurité du dispositif.

Principe de fonctionnement d'un servo-moteur

- Les servomoteurs sont souvent commandés par un câble électrique à trois fils, fournissant à la fois l'alimentation et les instructions de position.
- Les instructions de position sont transmises via un signal PWM, où la durée des impulsions détermine l'angle de l'axe de sortie.
- Ce signal est régulièrement répété toutes les 20 millisecondes pour un contrôle continu.
- L'électronique du servomoteur ajuste constamment la position de l'axe de sortie en fonction des informations reçues.
- Cette correction en temps réel permet de maintenir la position angulaire désirée, mesurée par le potentiomètre intégré.
- Ainsi, les servomoteurs offrent un contrôle précis et réactif, ce qui en fait un composant essentiel pour de nombreuses applications.

5-3- Capteur ultrasons HC-SR04

Le capteur ultrasons HC-SR04 est un dispositif électronique utilisé pour mesurer des distances en émettant des ondes sonores à haute fréquence et en détectant leur écho après réflexion sur un obstacle. Son rôle principal est de fournir des données précises sur la distance entre le capteur et l'objet le plus proche, permettant ainsi aux systèmes autonomes de naviguer, d'éviter les collisions ou de contrôler des dispositifs à distance. Grâce à sa conception compacte et à sa fiabilité, le capteur ultrasons HC-SR04 est largement utilisé dans les projets d'automatisation, de robotique, de domotique et d'autres applications nécessitant une détection de proximité précise.



5-4- Capteur Infrarouge

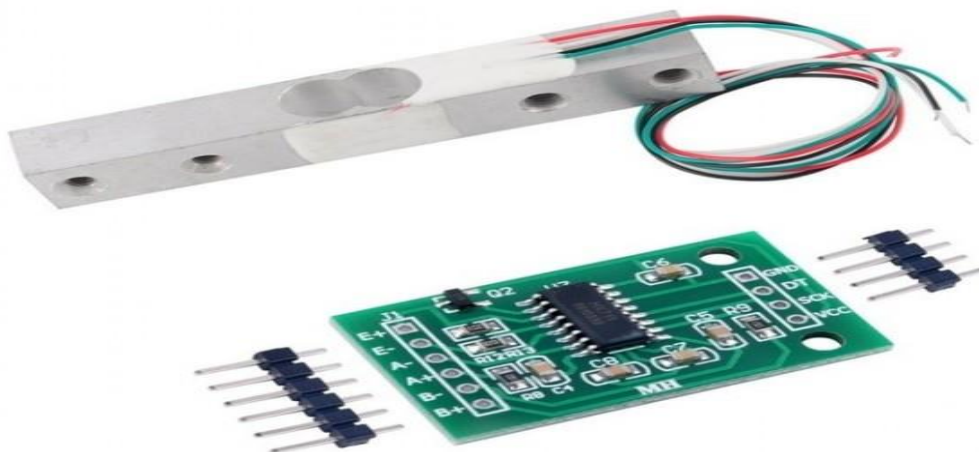
Un capteur infrarouge est un dispositif électronique qui détecte les radiations infrarouges émis par des objets. Il fonctionne en émettant un faisceau infrarouge et en mesurant la réflexion de ce faisceau pour déterminer la présence et la distance d'un objet. Les capteurs infrarouges sont largement utilisés dans les télécommandes, les systèmes de détection de mouvement, et les dispositifs de mesure de distance. Ils permettent la détection sans contact et peuvent fonctionner dans des conditions de faible luminosité. Leur rôle principal est d'améliorer l'automatisation et la sécurité dans divers systèmes.



Banggood.com

5-5- Capteur de Poids Hx711

Le capteur de poids HX711 est un module de précision utilisé pour mesurer le poids en convertissant les signaux analogiques des cellules de charge en données numériques. Il intègre un amplificateur et un convertisseur analogique-numérique (ADC) 24 bits pour des mesures précises. Couramment utilisé dans les balances électroniques et les systèmes de pesage, il permet une interface simple avec les microcontrôleurs. Le HX711 offre deux canaux d'entrée et une sortie série pour la transmission des données. Sa haute précision et sa stabilité en font un choix populaire pour les applications nécessitant des mesures de poids précises.



6- Conclusion

En conclusion, ce chapitre a fourni une vision détaillée de l'architecture et des choix technologiques de "Mo Belle". Les diagrammes UML et les interactions des composants ont permis de comprendre la structure et le fonctionnement du système. Les méthodologies adoptées garantissent sa robustesse et son efficacité. Ces fondations techniques solides sont essentielles pour le succès et la pérennité du projet.

Chapitre 3 : Spécification des besoins

1- Introduction

Dans cette partie, nous allons aborder la spécification des besoins pour le projet "Mo Belle". Nous détaillerons les exigences fonctionnelles et non fonctionnelles nécessaires pour le développement de notre système de gestion des poubelles automatiques. L'objectif est de définir clairement les fonctionnalités attendues et les contraintes techniques afin de garantir la réussite et l'efficacité du projet.

2- Spécification des besoins fonctionnel

2.1. Gestion des Poubelles

Chaque spécification de besoin fonctionnel décrit une fonctionnalité spécifique de l'application "Mo Belle" et explique comment elle répond aux besoins des utilisateurs. Ces fonctionnalités permettent aux utilisateurs de consulter, rechercher, ajouter, modifier et supprimer des informations relatives aux poubelles.

2.1.1. Afficher la liste des Poubelles

L'application affiche une liste des poubelles disponibles, permettant aux utilisateurs de parcourir les options disponibles et d'obtenir des informations telles que le niveau de remplissage, l'emplacement, l'état de la poubelle, etc.

2.1.2. Afficher les détails d'une Poubelle

Lorsque les utilisateurs sélectionnent une poubelle spécifique à partir de la liste, l'application affiche les détails de cette poubelle, y compris le niveau de remplissage actuel, l'emplacement précis, les dernières dates de vidage, etc.

2.1.3. Rechercher une Poubelle

Les utilisateurs peuvent utiliser une fonction de recherche pour trouver une poubelle spécifique en entrant des mots-clés, des emplacements ou d'autres critères de recherche pertinents.

2.1.4. Ajouter une nouvelle Poubelle

Les administrateurs de l'application peuvent ajouter de nouvelles poubelles à la liste disponible sur l'application. Ils peuvent fournir les détails tels que l'emplacement, le type de capteur, les niveaux de remplissage initiaux, etc., pour chaque poubelle ajoutée.

2.1.5. Modifier les informations d'une Poubelle

Les administrateurs peuvent mettre à jour les informations existantes des poubelles, comme l'emplacement, le type de capteur, les niveaux de remplissage, etc. Cela leur permet de corriger les erreurs, de mettre à jour les informations ou de modifier les paramètres des poubelles.

2.1.6. Supprimer une Poubelle

Si une poubelle n'est plus en service ou doit être retirée du système, les administrateurs peuvent la supprimer de la liste des poubelles disponibles.

2.2. Gestion des Utilisateurs

La gestion des utilisateurs permet aux administrateurs de gérer les comptes des utilisateurs et de configurer les permissions. Cependant, les utilisateurs n'ont pas besoin de s'inscrire et peuvent seulement consulter les informations des poubelles et la carte des emplacements.

2.2.1. Authentification des Administrateurs

Les administrateurs peuvent se connecter à leur compte en utilisant leurs identifiants (adresse e-mail et mot de passe) pour accéder aux fonctionnalités de gestion des poubelles et des utilisateurs.

2.2.2. Gestion du Profil Administrateur

Les administrateurs peuvent mettre à jour leurs informations personnelles telles que l'adresse e-mail et le mot de passe. Ils peuvent également consulter l'historique des modifications apportées aux poubelles et gérer les paramètres de compte.

2.3. Gestion des Administrateurs

La gestion des administrateurs permet de contrôler et de gérer les personnes ayant des droits étendus dans l'application "Mo Belle". Les administrateurs peuvent ajouter, modifier et supprimer des personnels.

2.3.1. Ajouter un Administrateur

Les administrateurs peuvent ajouter de nouveaux personnels en fournissant les informations nécessaires telles que le nom, l'adresse e-mail et les autorisations d'accès.

2.3.2. Modifier les Informations d'un Personnel

Les administrateurs peuvent mettre à jour les informations d'un personnel existant, y compris son nom, son adresse e-mail, ses autorisations d'accès, etc.

2.3.3. Supprimer un Personnel

Les administrateurs peuvent supprimer un personnel du système s'il n'est plus nécessaire ou s'il y a des changements dans la structure de gestion.

2.4. Gestion des Capteurs

La gestion des capteurs permet aux administrateurs de maintenir une liste complète et à jour des capteurs utilisés dans l'application "Mo Belle". Les utilisateurs peuvent ensuite accéder à cette liste pour vérifier l'état et les niveaux de remplissage des poubelles.

2.4.1. Ajouter un Nouveau Capteur

Les administrateurs peuvent ajouter de nouveaux capteurs à la liste disponible sur l'application "Mo Belle". Ils peuvent fournir des informations telles que le type de capteur (ultrasonique, poids HX711), l'emplacement de la poubelle, etc.

2.4.2. Modifier les Informations d'un Capteur

Les administrateurs peuvent mettre à jour les informations existantes d'un capteur, telles que son type, son emplacement, ses paramètres, etc. Cela peut être utile en cas de correction d'erreurs ou de changements dans les détails du capteur.

2.4.3. Supprimer un Capteur

Si un capteur n'est plus en service ou doit être retiré du système, les administrateurs peuvent le supprimer de la liste des capteurs disponibles.

2.5. Gestion des Itinéraires de Ramassage

La gestion des itinéraires de ramassage permet aux administrateurs de planifier et d'optimiser les itinéraires pour le ramassage des poubelles.

2.5.1. Planifier un Itinéraire de Ramassage

Les administrateurs peuvent planifier les itinéraires de ramassage en fonction des données de remplissage des poubelles. Cela permet de créer des trajets optimisés pour la collecte des déchets.

2.5.2. Mettre à Jour les Itinéraires

Les administrateurs peuvent mettre à jour les itinéraires de ramassage en fonction des changements de données ou des nouvelles informations sur les poubelles.

2.6. Visualisation sur Carte

La visualisation sur carte permet aux utilisateurs de consulter l'emplacement des poubelles et de vérifier leur état en temps réel.

2.6.1. Afficher la Carte des Emplacements des Poubelles

L'application affiche une carte interactive montrant l'emplacement des poubelles avec des indicateurs de leur niveau de remplissage.

2.6.2. Mettre à Jour les Informations sur la Carte

Les informations de la carte sont mises à jour en temps réel pour refléter les données les plus récentes sur l'état des poubelles.

2.6.3. Filtrer les Poubelles sur la Carte

Les utilisateurs peuvent filtrer les poubelles sur la carte en fonction de critères spécifiques tels que le niveau de remplissage, le type de déchets, etc.

3- Spécification des besoins non fonctionnel

3.1. Performance et Scalabilité

3.1.1. Temps de Réponse

L'application doit offrir un temps de réponse rapide, inférieur à 2 secondes pour l'affichage des données des poubelles et la mise à jour des informations sur la carte. Cela garantit une expérience utilisateur fluide et réactive.

3.1.2. Scalabilité

Le système doit être capable de gérer une augmentation du nombre de poubelles et d'utilisateurs sans dégradation des performances. Cela inclut la capacité à ajouter de nouveaux capteurs et de nouvelles zones de collecte sans nécessiter de refonte majeure de l'architecture.

3.2. Sécurité

3.2.1. Protection des Données

Les données des utilisateurs et des poubelles doivent être protégées contre les accès non autorisés. Cela inclut la mise en œuvre de mesures de cryptage pour les données sensibles et l'utilisation de protocoles sécurisés pour la communication des données.

3.2.2. Authentification et Autorisation

L'accès aux fonctionnalités spécifiques du système doit être restreint aux personnels autorisés via un système d'authentification robuste. Les personnels doivent être capables de se connecter pour voir les caractéristiques des poubelles et les poubelles à vider, sans pouvoir modifier les données. Les utilisateurs classiques, quant à eux, peuvent uniquement consulter les poubelles et leur emplacement sans nécessiter d'authentification.

3.3. Fiabilité

3.3.1. Disponibilité

L'application doit être disponible et fonctionnelle 99,9% du temps, avec des temps d'arrêt planifiés pour la maintenance limitée à des périodes de faible activité. Cela garantit une disponibilité continue pour les utilisateurs et les administrateurs.

3.3.2. Tolérance aux Pannes

Le système doit être conçu pour continuer à fonctionner même en cas de défaillance d'un composant individuel. Les mécanismes de sauvegarde et de récupération automatique doivent être mis en place pour minimiser les interruptions de service.

3.4. Maintenabilité

3.4.1. Documentation

L'application doit être accompagnée d'une documentation complète et à jour. Cela inclut des guides d'installation, des manuels utilisateur, des spécifications techniques et des instructions pour la maintenance.

3.4.2. Modularité

Le code doit être structuré de manière modulaire pour faciliter les mises à jour et les modifications futures. Les différentes fonctionnalités (comme la gestion des poubelles, l'affichage de la carte, etc.) doivent être séparées en modules indépendants.

3.5. Compatibilité

3.5.1. Compatibilité Multi-Plateforme

L'application doit être compatible avec les principaux systèmes d'exploitation mobiles (iOS et Android). Elle doit également être accessible via un navigateur web pour offrir une flexibilité maximale aux utilisateurs.

3.5.2. Intégration avec des Capteurs IoT

Le système doit être capable de s'intégrer avec une variété de capteurs IoT, y compris les capteurs ultrasoniques et les capteurs de poids HX711, pour recueillir les données de remplissage des poubelles.

3.6. Expérience Utilisateur (UX)

3.6.1. Interface Intuitive

L'interface utilisateur doit être intuitive et facile à utiliser, même pour les utilisateurs sans expertise technique. Cela inclut des menus clairs, des boutons bien étiquetés et des instructions simples.

3.6.2. Design Réactif

L'application doit être conçue pour fonctionner de manière optimale sur différents appareils et tailles d'écran, offrant une expérience utilisateur cohérente et agréable, qu'il s'agisse de smartphones, de tablettes ou d'ordinateurs de bureau.

4- Présentation des cas d'utilisations

4.1. Présentations des acteurs

Acteur : Utilisateur

Description : Un utilisateur est une personne qui accède à l'application de gestion des poubelles automatiques sans avoir besoin de s'authentifier. Il peut consulter la liste des poubelles et afficher la carte pour voir l'emplacement des poubelles.

Acteur : Personnel

Description : Le personnel est un utilisateur qui possède un compte sur l'application et peut s'authentifier pour accéder à des fonctionnalités supplémentaires. Il peut voir les caractéristiques des poubelles et les poubelles à vider. Cependant, il n'a pas la permission de modifier les données.

Acteur : Administrateur

Description : L'administrateur est une personne responsable de la gestion complète de l'application. Il dispose de privilèges spéciaux qui lui permettent d'ajouter, modifier ou supprimer des poubelles. Il peut également gérer les comptes du personnel et accéder à des rapports détaillés sur l'utilisation et la performance des poubelles.

4.2. Descriptions des cas d'utilisations

4.2.1. Administrateur

Cas n° 1	Ajouter du personnel
Acteur	Administrateur
Objectif	Permettre à l'administrateur d'ajouter de nouveaux membres du personnel dans le système.
Pré-condition	L'administrateur est authentifié et accède à la section de gestion du personnel.
Post-condition	Le nouveau membre du personnel est ajouté et peut se connecter au système.
Scénario nominal	<ol style="list-style-type: none">1. L'administrateur accède à la page d'ajout de personnel.2. L'administrateur remplit le formulaire avec les informations nécessaires (nom, adresse e-mail, rôle, etc.).3. L'administrateur soumet le formulaire.4. Le système vérifie les informations et ajoute le nouveau membre du personnel.5. Le système affiche un message de confirmation de l'ajout réussi.
Scénario alternatif	<p>2a. Si les informations sont incomplètes ou incorrectes :</p> <ul style="list-style-type: none">• Le système affiche un message d'erreur et invite l'administrateur à corriger les informations.

Cas n° 2	Voir tous les personnels
Acteur	Administrateur
Objectif	Permettre à l'administrateur de consulter la liste de tout le personnel enregistré dans le système.
Pré-condition	L'administrateur est authentifié et accède à la section de gestion du personnel.
Post-condition	L'administrateur visualise la liste de tout le personnel.
Scénario nominal	<ol style="list-style-type: none">1. L'administrateur accède à la page de liste du personnel.2. L'administrateur consulte la liste avec les détails de chaque membre du personnel (nom, rôle, etc.).3. Le système affiche la liste complète du personnel.
Scénario alternatif	<p>2a. Si la liste du personnel est vide :</p> <ul style="list-style-type: none">• Le système affiche un message indiquant qu'il n'y a aucun membre du personnel disponible.

Cas n° 3	Ajouter des poubelles
Acteur	Administrateur
Objectif	Permettre à l'administrateur d'ajouter de nouvelles poubelles dans le système.
Pré-condition	L'administrateur est authentifié et accède à la section de gestion des poubelles.

Post-condition	La nouvelle poubelle est ajoutée dans le système.
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur accède à la page d'ajout de poubelles. 2. L'administrateur remplit le formulaire avec les informations nécessaires (type de poubelle, emplacement, etc.). 3. L'administrateur soumet le formulaire. 4. Le système vérifie les informations et ajoute la nouvelle poubelle. 5. Le système affiche un message de confirmation de l'ajout réussi.
Scénario alternatif	2a. Si les informations sont incomplètes ou incorrectes : <ul style="list-style-type: none"> • Le système affiche un message d'erreur et invite l'administrateur à corriger les informations.

Cas n° 4	Voir toutes les poubelles
Acteur	Administrateur
Objectif	Permettre à l'administrateur de consulter la liste de toutes les poubelles enregistrées dans le système.
Pré-condition	L'administrateur est authentifié et accède à la section de gestion des poubelles.
Post-condition	L'administrateur visualise la liste de toutes les poubelles.
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur accède à la page de liste des poubelles. 2. L'administrateur consulte la liste avec les détails de chaque poubelle (type, emplacement, etc.). 3. Le système affiche la liste complète des poubelles.
Scénario alternatif	2a. Si la liste des poubelles est vide : <ul style="list-style-type: none"> • Le système affiche un message indiquant qu'il n'y a aucune poubelle disponible.

Cas n° 5	Voir les poubelles à vider
Acteur	Administrateur
Objectif	Permettre à l'administrateur de voir quelles poubelles doivent être vidées.
Pré-condition	L'administrateur est authentifié et accède à la section de gestion des poubelles.
Post-condition	L'administrateur visualise les poubelles qui doivent être vidées.
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur accède à la page des poubelles à vider. 2. L'administrateur consulte la liste des poubelles avec des niveaux de remplissage élevés. 3. Le système affiche la liste des poubelles à vider.
Scénario alternatif	2a. Si aucune poubelle n'a besoin d'être vidée : <ul style="list-style-type: none"> • Le système affiche un message indiquant qu'il n'y a aucune poubelle à vider.

Cas n° 6	Voir la carte
Acteur	Administrateur
Objectif	Permettre à l'administrateur de voir l'emplacement des poubelles sur une carte.
Pré-condition	L'administrateur est authentifié et accède à la section de la carte.
Post-condition	L'administrateur visualise les emplacements des poubelles sur la carte.
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur accède à la page de la carte des poubelles. 2. L'administrateur consulte les emplacements des poubelles sur la carte. 3. Le système affiche la carte avec les emplacements des poubelles.
Scénario alternatif	2a. Si aucun emplacement de poubelle n'est disponible : <ul style="list-style-type: none"> • Le système affiche un message indiquant qu'il n'y a aucun emplacement de poubelle disponible.

4.2.2. Personnel

Cas n° 1	Voir toutes les poubelles
Acteur	Personnel
Objectif	Permettre au personnel de consulter la liste de toutes les poubelles enregistrées dans le système.
Pré-condition	Le personnel est authentifié et accède à la section de gestion des poubelles.
Post-condition	Le personnel visualise la liste de toutes les poubelles.
Scénario nominal	<ol style="list-style-type: none"> 1. Le personnel accède à la page de liste des poubelles. 2. Le personnel consulte la liste avec les détails de chaque poubelle (type, emplacement, etc.). 3. Le système affiche la liste complète des poubelles.
Scénario alternatif	2a. Si la liste des poubelles est vide : <ul style="list-style-type: none"> • Le système affiche un message indiquant qu'il n'y a aucune poubelle disponible.

Cas n° 2	Voir les poubelles à vider
Acteur	Personnel
Objectif	Permettre au personnel de voir quelles poubelles doivent être vidées.
Pré-condition	Le personnel est authentifié et accède à la section de gestion des poubelles
Post-condition	Le personnel visualise les poubelles qui doivent être vidées.
Scénario nominal	<ol style="list-style-type: none"> 1. Le personnel accède à la page des poubelles à vider. 2. Le personnel consulte la liste des poubelles avec des niveaux de remplissage élevés. 3. Le système affiche la liste des poubelles à vider.
Scénario alternatif	2a. Si aucune poubelle n'a besoin d'être vidée :

	<ul style="list-style-type: none"> Le système affiche un message indiquant qu'il n'y a aucune poubelle à vider.
--	--

Cas n° 3	Voir la carte
Acteur	Personnel
Objectif	Permettre au personnel de voir l'emplacement des poubelles sur une carte.
Pré-condition	Le personnel est authentifié et accède à la section de la carte.
Post-condition	Le personnel visualise les emplacements des poubelles sur la carte.
Scénario nominal	<ol style="list-style-type: none"> Le personnel accède à la page de la carte des poubelles. Le personnel consulte les emplacements des poubelles sur la carte. Le système affiche la carte avec les emplacements des poubelles.
Scénario alternatif	2a. Si aucun emplacement de poubelle n'est disponible : <ul style="list-style-type: none"> Le système affiche un message indiquant qu'il n'y a aucun emplacement de poubelle disponible.

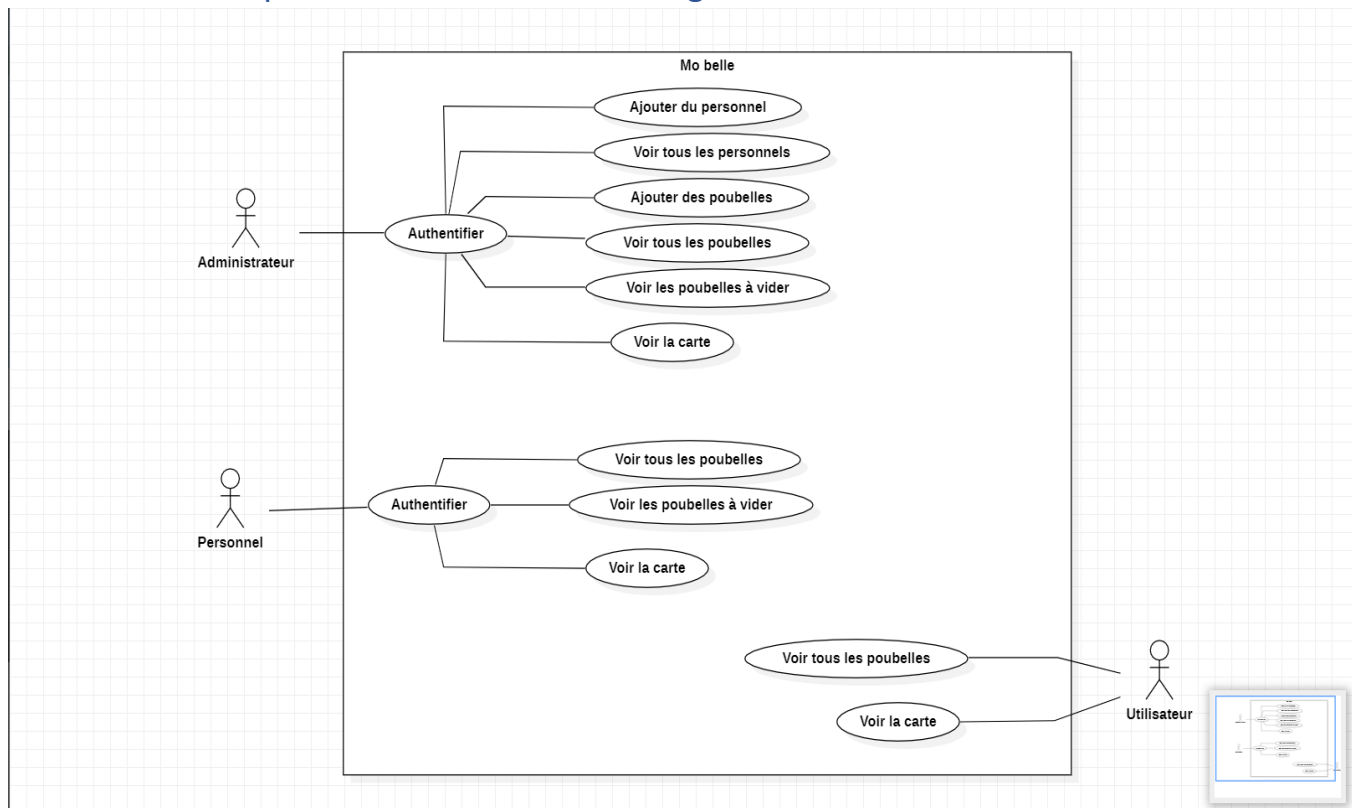
4.2.3. Utilisateur

Cas n° 1	Voir toutes les poubelles disponible
Acteur	Utilisateur
Objectif	Permettre à l'utilisateur de consulter la liste de toutes les poubelles enregistrées dans le système.
Pré-condition	L'utilisateur accède à la section de gestion des poubelles.
Post-condition	L'utilisateur visualise la liste de toutes les poubelles.
Scénario nominal	<ol style="list-style-type: none"> L'utilisateur accède à la page de liste des poubelles. L'utilisateur consulte la liste avec les détails de chaque poubelle (type, emplacement, etc.). Le système affiche la liste complète des poubelles.
Scénario alternatif	2a. Si la liste des poubelles est vide : <ul style="list-style-type: none"> Le système affiche un message indiquant qu'il n'y a aucune poubelle disponible.

Cas n° 2	Voir la carte
Acteur	Utilisateur
Objectif	Permettre à l'utilisateur de voir l'emplacement des poubelles sur une carte.
Pré-condition	L'utilisateur accède à la section de la carte.
Post-condition	L'utilisateur visualise les emplacements des poubelles sur la carte
Scénario nominal	<ol style="list-style-type: none"> L'utilisateur accède à la page de la carte des poubelles.

	<ol style="list-style-type: none"> 2. L'utilisateur consulte les emplacements des poubelles sur la carte. 3. Le système affiche la carte avec les emplacements des poubelles.
Scénario alternatif	<ol style="list-style-type: none"> 2a. Si aucun emplacement de poubelle n'est disponible : <ul style="list-style-type: none"> • Le système affiche un message indiquant qu'il n'y a aucun emplacement de poubelle disponible.

4.3. Descriptions des cas d'utilisations global



5- Conclusion

Pour résumer, cette partie a permis de définir de manière précise les besoins et exigences pour le développement du projet "Mo Belle". Les spécifications fonctionnelles et non fonctionnelles établies fournissent une base solide pour la conception et la mise en œuvre du système de gestion des poubelles automatiques, garantissant ainsi une solution efficace et adaptée aux attentes des utilisateurs.

Chapitre 4 : Conception du système

1- Introduction

Dans cette section, nous abordons la conception et la mise en œuvre du système "Mo Belle". Nous détaillons les choix technologiques, les architectures logicielles et matérielles, ainsi que les méthodes de développement adoptées pour réaliser un système de gestion des poubelles automatisé et intelligent.

2- Modélisation dynamique

2.1. Diagramme de séquence

2.1.1. Administrateur

Se Connecter

1. **Action de l'administrateur** : L'administrateur envoie une requête de connexion au système.
2. **Vérification des identifiants** :
 - **Système → Base de Données** : Le système transmet les identifiants de l'administrateur à la base de données pour vérification.
 - **Base de Données → Système** : La base de données vérifie les identifiants et renvoie le résultat au système.
3. **Confirmation d'identifiants** :
 - **Système → Administrateur** : Si les identifiants sont corrects, le système informe l'administrateur que l'accès est accordé.

Ajouter Personnel

1. **Demande d'ajout** :
 - **Administrateur → Système** : L'administrateur envoie une requête pour ajouter un nouveau personnel.
2. **Enregistrement des informations** :
 - **Système → Base de Données** : Le système transmet les informations du nouveau personnel à la base de données.
 - **Base de Données → Système** : La base de données enregistre les informations et confirme l'ajout.
3. **Confirmation d'ajout** :

- **Système → Administrateur** : Le système informe l'administrateur que le personnel a été ajouté avec succès.

Ajouter Poubelle

1. Demande d'ajout :

- **Administrateur → Système** : L'administrateur envoie une requête pour ajouter une nouvelle poubelle.

2. Enregistrement des informations :

- **Système → Base de Données** : Le système transmet les informations de la nouvelle poubelle à la base de données.
- **Base de Données → Système** : La base de données enregistre les informations et confirme l'ajout.

3. Confirmation d'ajout :

- **Système → Administrateur** : Le système informe l'administrateur que la poubelle a été ajoutée avec succès.

Voir les Poubelles

1. Demande de visualisation :

- **Administrateur → Système** : L'administrateur envoie une requête pour visualiser les poubelles.

2. Récupération des données :

- **Système → Base de Données** : Le système demande les données des poubelles à la base de données.
- **Base de Données → Système** : La base de données renvoie les données des poubelles au système.

3. Affichage des données :

- **Système → Administrateur** : Le système affiche les données des poubelles à l'administrateur.

Voir Carte

1. Demande de visualisation :

- **Administrateur → Système** : L'administrateur envoie une requête pour visualiser la carte des poubelles.

2. Récupération des données :

- **Système → Base de Données** : Le système demande les données nécessaires pour afficher la carte à la base de données.
- **Base de Données → Système** : La base de données envoie les données nécessaires au système.

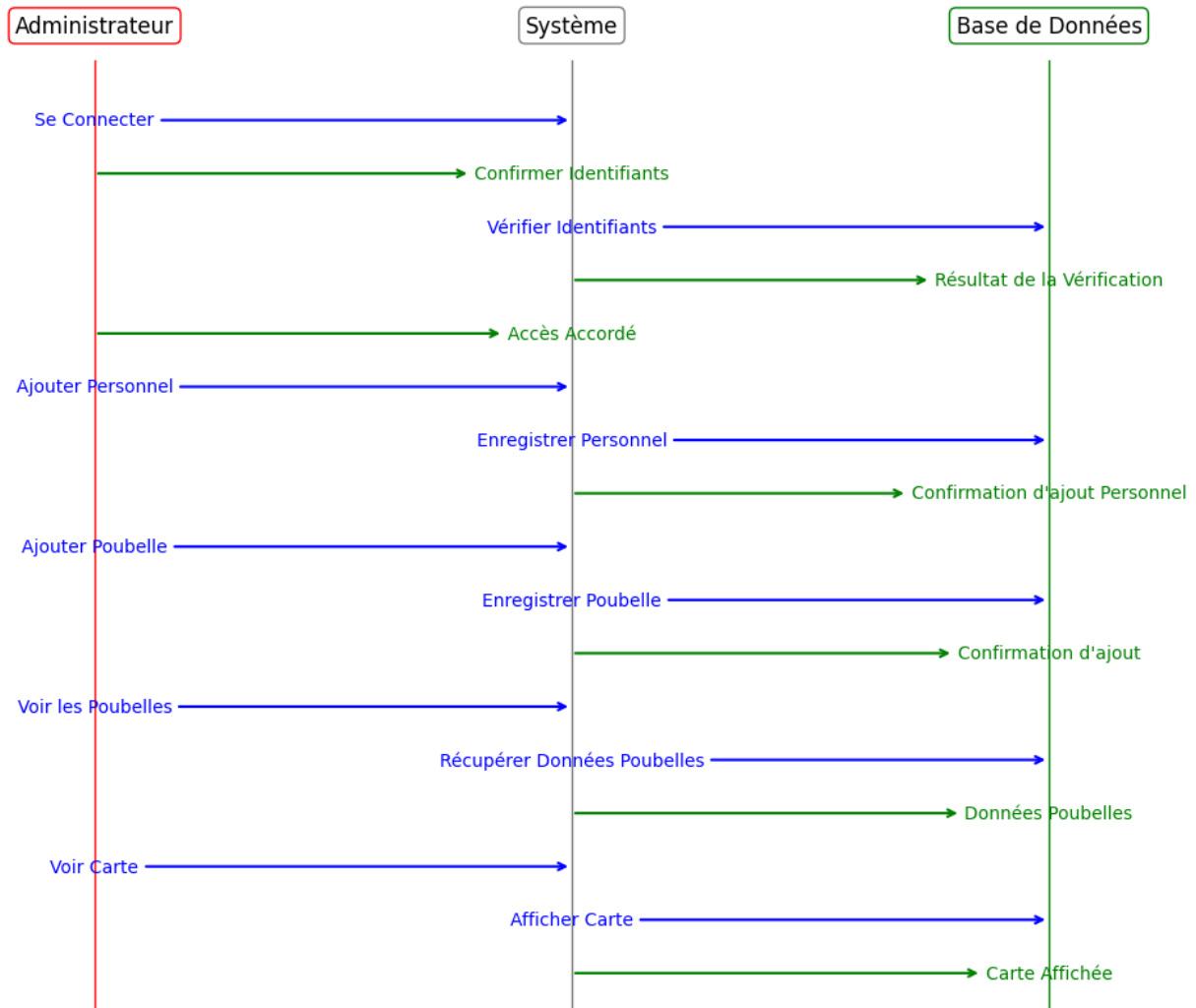
3. Affichage de la carte :

- **Système → Administrateur** : Le système affiche la carte des poubelles à l'administrateur.

Points Clés du Diagramme :

- **Interactivité** : Les interactions entre l'administrateur, le système et la base de données sont clairement définies, montrant le flux de données et les réponses à chaque étape.
- **Vérification des données** : Chaque action administrative implique une vérification et une confirmation des données, assurant l'intégrité et la fiabilité des informations gérées.
- **Retour d'information** : Le système fournit des retours d'information à l'administrateur après chaque action, garantissant une bonne communication et une gestion efficace des opérations.

Diagramme de Séquence pour l'Administrateur



2.1.2. Personnel

Se Connecter

1. **Action du personnel** : Le personnel envoie une requête de connexion au système.
2. **Vérification des identifiants** :
 - **Système → Base de Données** : Le système transmet les identifiants du personnel à la base de données pour vérification.
 - **Base de Données → Système** : La base de données vérifie les identifiants et renvoie le résultat au système.
3. **Confirmation d'identifiants** :
 - **Système → Personnel** : Si les identifiants sont corrects, le système informe le personnel que l'accès est accordé.

Voir les Poubelles

1. **Demande de visualisation** :
 - **Personnel → Système** : Le personnel envoie une requête pour visualiser les poubelles.
2. **Récupération des données** :
 - **Système → Base de Données** : Le système demande les données des poubelles à la base de données.
 - **Base de Données → Système** : La base de données renvoie les données des poubelles au système.
3. **Affichage des données** :
 - **Système → Personnel** : Le système affiche les données des poubelles au personnel.

Voir les Poubelles à Vider

1. **Demande de visualisation** :
 - **Personnel → Système** : Le personnel envoie une requête pour voir les poubelles à vider.
2. **Récupération des données** :
 - **Système → Base de Données** : Le système demande les données des poubelles à vider à la base de données.

- **Base de Données → Système** : La base de données renvoie les données des poubelles à vider au système.

3. Affichage des données :

- **Système → Personnel** : Le système affiche les données des poubelles à vider au personnel.

Voir Carte

1. Demande de visualisation :

- **Personnel → Système** : Le personnel envoie une requête pour visualiser la carte des poubelles.

2. Récupération des données :

- **Système → Base de Données** : Le système demande les données nécessaires pour afficher la carte à la base de données.
- **Base de Données → Système** : La base de données envoie les données nécessaires au système.

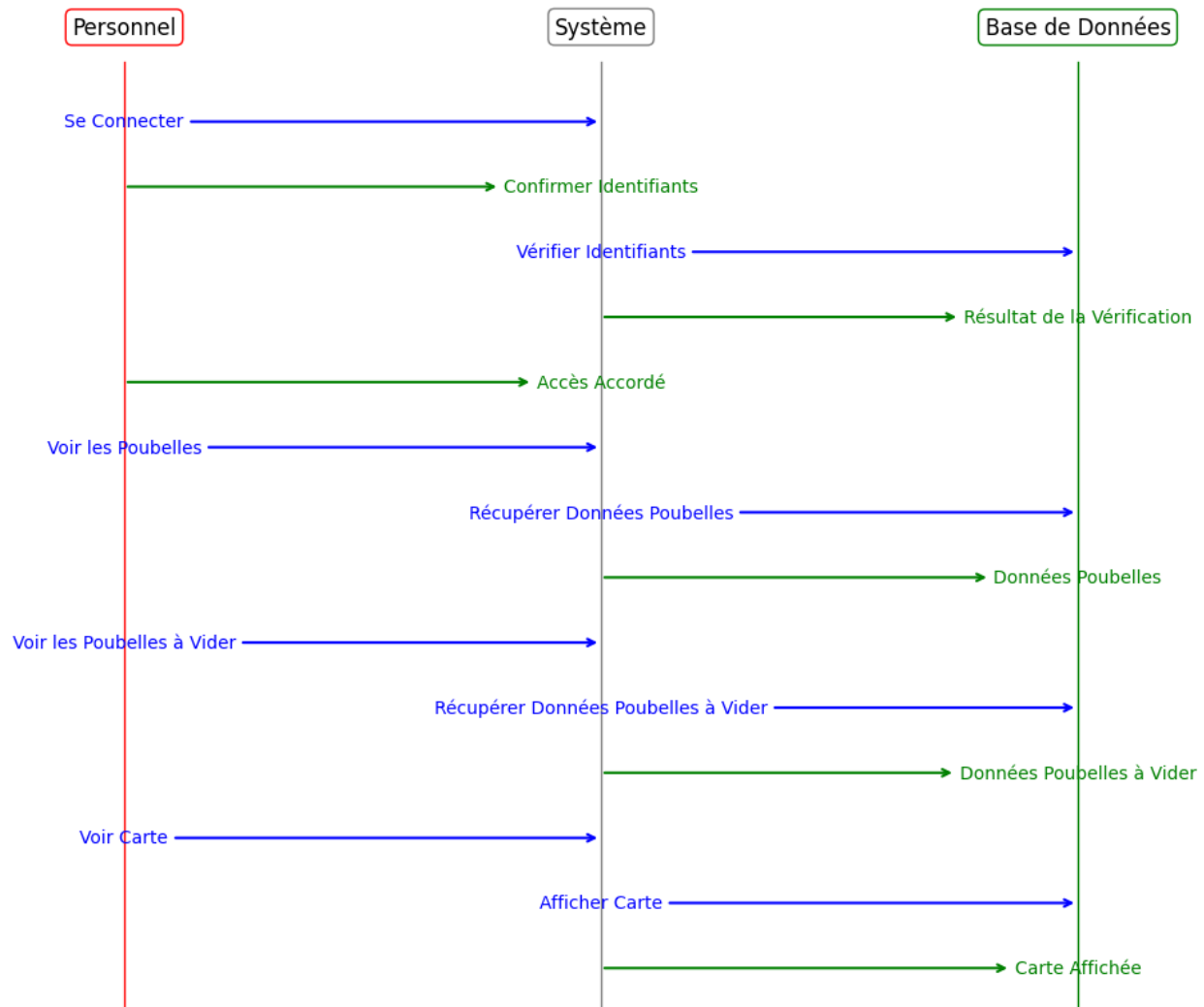
3. Affichage de la carte :

- **Système → Personnel** : Le système affiche la carte des poubelles au personnel.

Points Clés du Diagramme :

- **Interactivité** : Les interactions entre le personnel, le système et la base de données sont clairement définies, montrant le flux de données et les réponses à chaque étape.
- **Vérification des données** : Chaque action du personnel implique une vérification et une confirmation des données, assurant l'intégrité et la fiabilité des informations gérées.
- **Retour d'information** : Le système fournit des retours d'information au personnel après chaque action, garantissant une bonne communication et une gestion efficace des opérations.

Diagramme de Séquence pour le Personnel



2.1.3. Utilisateur

Voir les Poubelles

1. Demande de visualisation :

- **Utilisateur → Système** : L'utilisateur envoie une requête pour visualiser les poubelles.

2. Récupération des données :

- **Système → Base de Données** : Le système demande les données des poubelles à la base de données.

- **Base de Données → Système** : La base de données renvoie les données des poubelles au système.

3. Affichage des données :

- **Système → Utilisateur** : Le système affiche les données des poubelles à l'utilisateur.

Voir Carte

1. Demande de visualisation :

- **Utilisateur → Système** : L'utilisateur envoie une requête pour visualiser la carte des poubelles.

2. Récupération des données :

- **Système → Base de Données** : Le système demande les données nécessaires pour afficher la carte à la base de données.
- **Base de Données → Système** : La base de données envoie les données nécessaires au système.

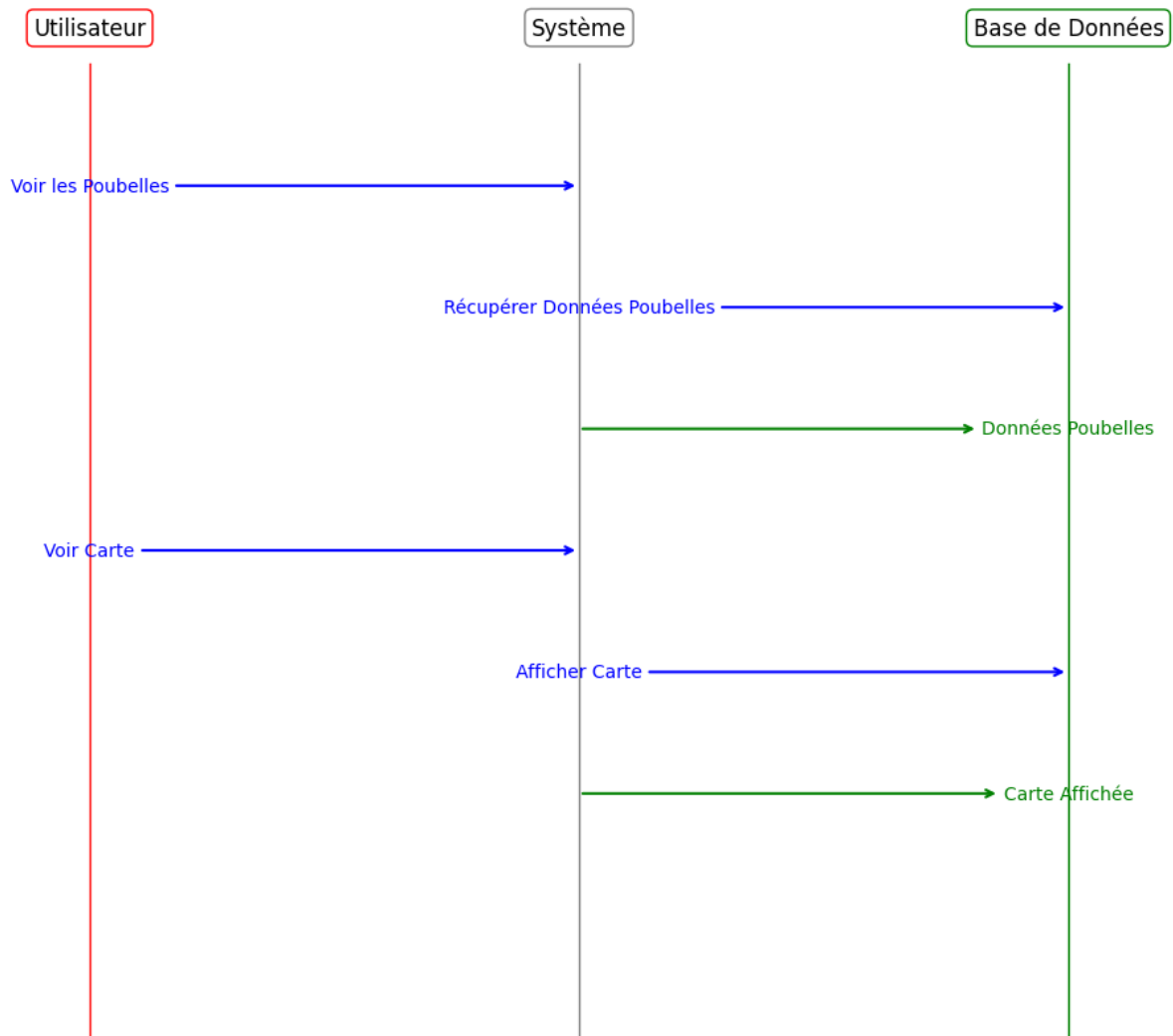
3. Affichage de la carte:

- **Système → Utilisateur** : Le système affiche la carte des poubelles à l'utilisateur.

Points Clés du Diagramme:

- **Interactivité** : Les interactions entre l'utilisateur, le système et la base de données sont clairement définies, montrant le flux de données et les réponses à chaque étape.
- **Vérification des données** : Chaque action de l'utilisateur implique une récupération et une présentation des données, assurant l'intégrité et la fiabilité des informations fournies.
- **Retour d'information** : Le système fournit des retours d'information à l'utilisateur après chaque action, garantissant une bonne communication et une expérience utilisateur fluide et intuitive.

Diagramme de Séquence pour l'Utilisateur



2.2. Diagramme d'état

2.2.1. Administrateur

Etat Initial:

- **Transition:** Démarrer
- **Description:** L'administrateur démarre l'application.

Etat Authentification:

- **Transitions:**

- ✓ **Authentifier** : Cette transition mène vers plusieurs états, selon l'action choisie par l'administrateur après authentification.
- **Description** : L'administrateur entre ses identifiants pour accéder au système. Une fois authentifié, il peut choisir diverses actions.

Etats Post-Authentification:

- **Ajouter du personnel:**
 - ✓ **Transition:** Personnel ajouté
 - ✓ **Description** : Permet à l'administrateur d'ajouter un nouveau membre du personnel. Après l'ajout, la transition retourne à l'état final.
- **Voir tous les personnels:**
 - ✓ **Transition:** Voir tous les personnels
 - ✓ **Description** : Permet à l'administrateur de visualiser la liste complète des membres du personnel. Après avoir consulté la liste, la transition retourne à l'état final.
- **Ajouter des poubelles:**
 - ✓ **Transition:** Poubelle ajoutée
 - ✓ **Description** : Permet à l'administrateur d'ajouter une nouvelle poubelle. Après l'ajout, la transition retourne à l'état final.
- **Voir toutes les poubelles:**
 - ✓ **Transition:** Voir toutes les poubelles
 - ✓ **Description** : Permet à l'administrateur de voir toutes les poubelles existantes. Après avoir consulté les poubelles, la transition retourne à l'état final.
- **Voir les poubelles à vider :**
 - ✓ **Transition** : Voir les poubelles à vider
 - ✓ **Description** : Permet à l'administrateur de visualiser les poubelles qui nécessitent d'être vidées. Après avoir consulté les poubelles à vider, la transition retourne à l'état final.
- **Voir la carte:**
 - ✓ **Transition:** Voir la carte

- ✓ **Description** : Permet à l'administrateur de voir la carte des emplacements des poubelles. Après avoir consulté la carte, la transition retourne à l'état final.

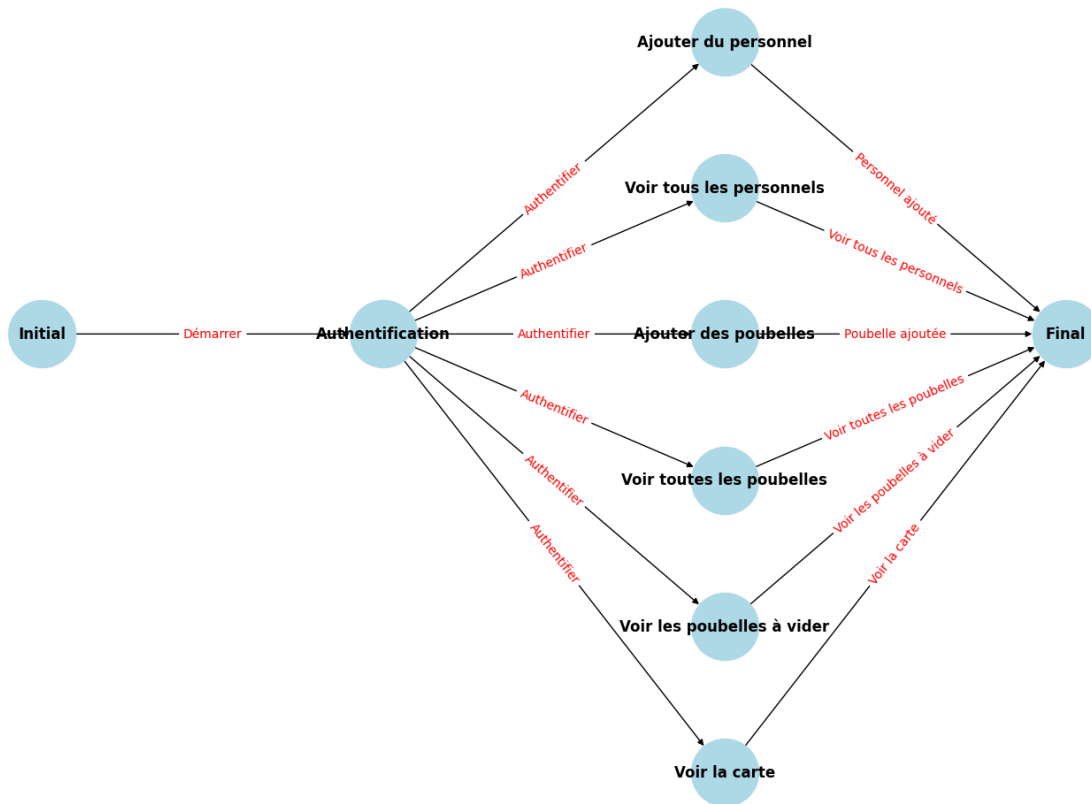
Etat Final:

- **Description** : Cet état marque la fin du processus courant, permettant de retourner à l'état initial ou d'autres actions selon les besoins de l'administrateur.

Points Clés du Diagramme:

- **Clarté des transitions** : Chaque transition est clairement définie, montrant les actions possibles de l'administrateur à chaque étape.
- **Retour à l'état final** : Après chaque action (ajout de personnel ou de poubelles, visualisation des listes et de la carte), l'administrateur retourne à l'état final, indiquant la fin de l'action courante.
- **Sécurité et contrôle** : L'authentification est une étape centrale, garantissant que seules les personnes autorisées peuvent accéder aux fonctionnalités administratives.

Diagramme d'état pour l'administrateur



2.2.2. Personnel

Etat Initial:

- **Transition** : Démarrer
- **Description** : Le personnel commence l'interaction avec le système.

Etat Authentification :

- **Transition** : Authentifier
- **Description** : Le personnel s'authentifie pour accéder aux fonctionnalités.

Etat Voir les poubelles :

- **Transition** : Voir les poubelles
- **Description** : Le personnel consulte la liste des poubelles disponibles. Une fois cette action terminée, la transition se fait vers l'état final.

Etat Voir les poubelles à vider :

- **Transition** : Voir les poubelles à vider
- **Description** : Le personnel consulte les poubelles qui doivent être vidées. Après cette action, la transition se fait également vers l'état final.

Etat Voir la carte :

- **Transition** : Voir la carte
- **Description** : Le personnel consulte la carte des poubelles. Après cette action, la transition se fait également vers l'état final.

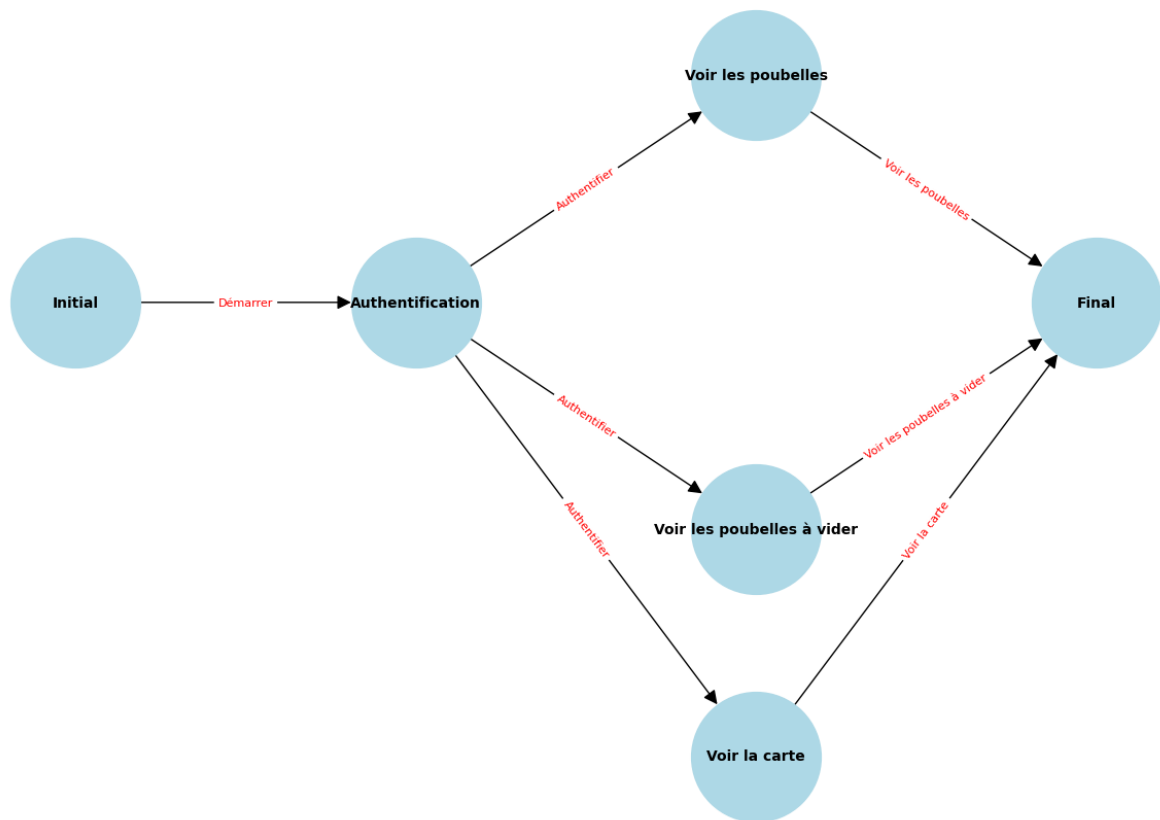
Etat Final :

- **Description** : Cet état marque la fin de l'interaction actuelle du personnel, lui permettant de retourner à l'état initial pour d'autres actions si nécessaire.

Points Clés du Diagramme :

- **Interactivité** : Les interactions entre le personnel, le système et la base de données sont clairement définies, montrant le flux de données et les réponses à chaque étape.
- **Vérification des données** : Chaque action du personnel implique une vérification et une confirmation des données, assurant l'intégrité et la fiabilité des informations gérées.
- **Retour d'information** : Le système fournit des retours d'information au personnel après chaque action, garantissant une bonne communication et une gestion efficace des opérations.

Diagramme d'état pour le personnel



2.2.3. Utilisateur

1. État Initial :

- **Transition :** Démarrer
- **Description :** L'utilisateur commence l'interaction avec le système.

2. État Voir les poubelles :

- **Transition :** Voir les poubelles
- **Description :** L'utilisateur consulte la liste des poubelles disponibles. Une fois cette action terminée, la transition se fait vers l'état final.

3. État Voir la carte :

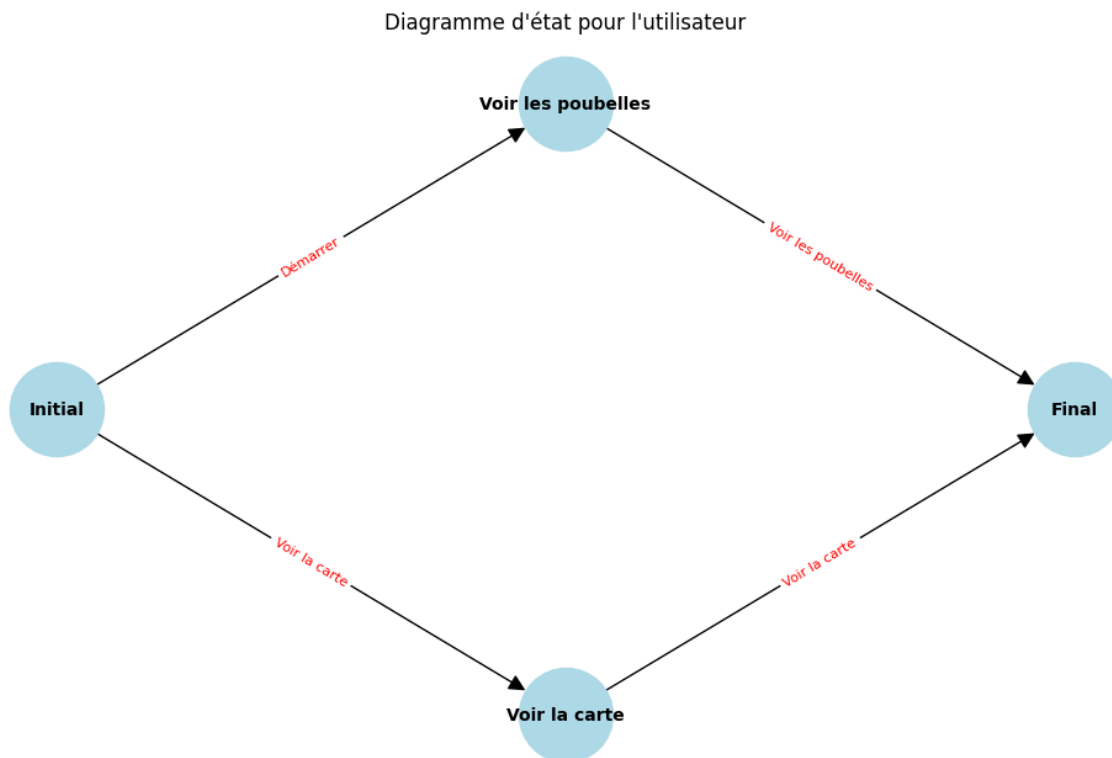
- **Transition :** Voir la carte
- **Description :** L'utilisateur consulte la carte des poubelles. Après cette action, la transition se fait également vers l'état final.

4. État Final :

- **Description** : Cet état marque la fin de l'interaction actuelle de l'utilisateur, lui permettant de retourner à l'état initial pour d'autres actions si nécessaire.

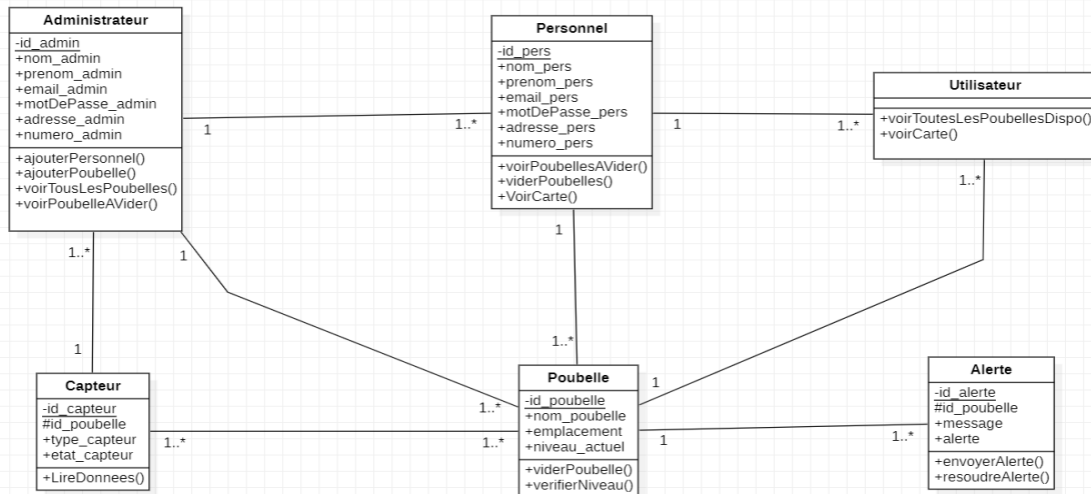
Points Clés du Diagramme :

- **Interactivité** : Les interactions entre l'utilisateur et le système sont clairement définies, montrant le flux de données et les réponses à chaque étape.
- **Accès Direct** : L'utilisateur peut accéder directement aux fonctionnalités sans avoir besoin de s'authentifier, simplifiant ainsi l'interaction.
- **Retour d'information** : Le système fournit des retours d'information à l'utilisateur après chaque action, garantissant une bonne communication et une gestion efficace des opérations.

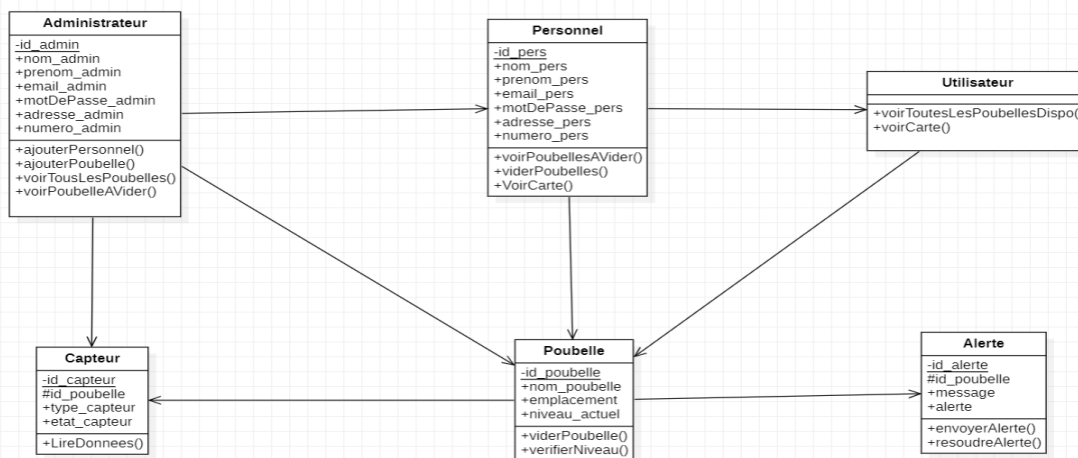


3- Modélisation statique

3.1. Diagramme de classe



3.2. Diagramme de relation



3.3. Dictionnaire de données

Nom de la colonne	Type de données	Taille	Description	Clé	Nom de la table
id_admin	N	3	Identifiant	Primaire	Administrateur
nom_admin	AN	50	Nom		
prenom_admin	AN	50	Prenom		
email_admin	AN	100	Email		
motDePasse_admin	AN	100	Mot de Passe		
adresse_admin	AN	100	Adresse		
numero_admin	N	11	Numero		

Nom de la colonne	Type de données	Taille	Description	Clé	Nom de la table
id_pers	N	3	Identifiant	Primaire	Personnel
nom_pers	AN	50	Nom		
prenom_pers	AN	50	Prenom		
email_pers	AN	100	Email		
motDePasse_pers	AN	100	Mot de Passe		
adresse_pers	AN	100	Adresse		
numero_pers	N	11	Numero		

Nom de la colonne	Type de données	Taille	Description	Clé	Nom de la table
id_capteur	N	3	Identifiant_cap	Primaire	Capteur
id_poubelle	AN	5	Identifiant_pou	Etrangère	
type_capteur	AN	20	Type		
etat_capteur	AN	20	Etat		

Nom de la colonne	Type de données	Taille	Description	Clé	Nom de la table
id_poubelle	AN	5	Identifiant_poubelle	Primaire	Poubelle
nom_poubelle	AN	50	Nom_poubelle		
emplacement	AN	50	Emplacement		
niveau_actuel	N	10	Niveau actuel		

Nom de la colonne	Type de données	Taille	Description	Clé	Nom de la table
Id_alerte	N	5	Identifiant_alerte	Primaire	Alerte
Id_poubelle	AN	5	Identifiant_poubelle	Etrangère	
message	AN	500	Message		

alerte	AN	500	Alerte		
--------	----	-----	--------	--	--

4- Conclusion

En conclusion, cette section a présenté une vue d'ensemble complète de la conception et de la mise en œuvre du système "Mo Belle". Les choix technologiques et méthodologiques expliqués ici jettent les bases solides nécessaires pour le développement efficace et l'intégration réussie de notre solution de gestion intelligente des déchets.

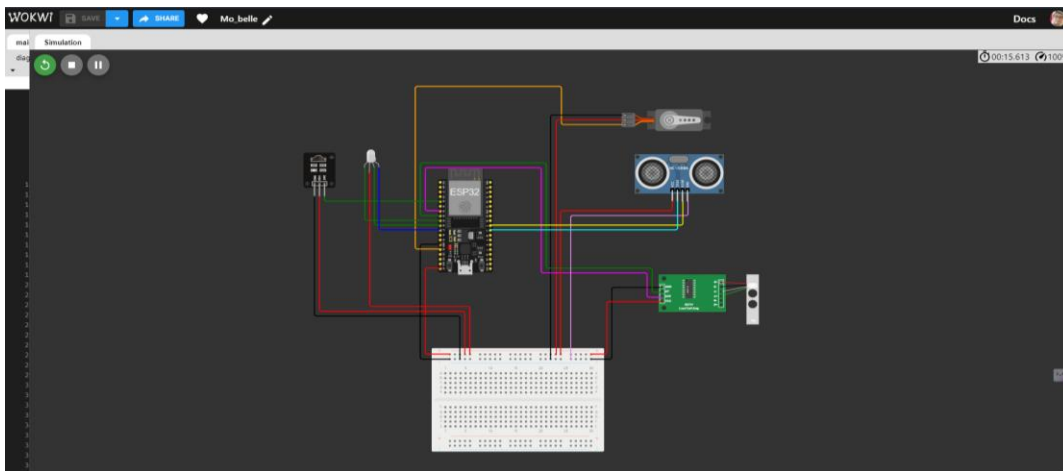
Chapitre 5 : Réalisation du système

1- Introduction

Dans cette partie, nous examinerons les résultats obtenus et l'analyse des performances du système "Mo Belle". Nous évaluerons l'efficacité de notre solution en termes de précision, de réactivité et de fiabilité, tout en identifiant les aspects à améliorer pour garantir une gestion optimale des déchets.

2- Wokwi

Wokwi est un simulateur en ligne pour projets électroniques, parfait pour le prototypage avec des microcontrôleurs comme l'ESP32. Pour notre projet de gestion des poubelles automatiques, il permet de tester et valider les capteurs et les modules sans matériel physique. Wokwi aide à simuler les interactions et le comportement du système en temps réel. Cela améliore la fiabilité et l'efficacité avant le déploiement réel.




3- Principales interfaces graphiques

Bienvenue à l'administration et au personnel de Mo Belle. Pour continuer, cliquez sur "Développeur" pour vous authentifier. Une route propre dépend autant de l'efficacité du service de nettoyage que de l'éducation des passants.



Seuls les administrateurs et le personnel autorisé ont accès à cette page d'authentification. La connexion est obligatoire.



Mo Belle

Email

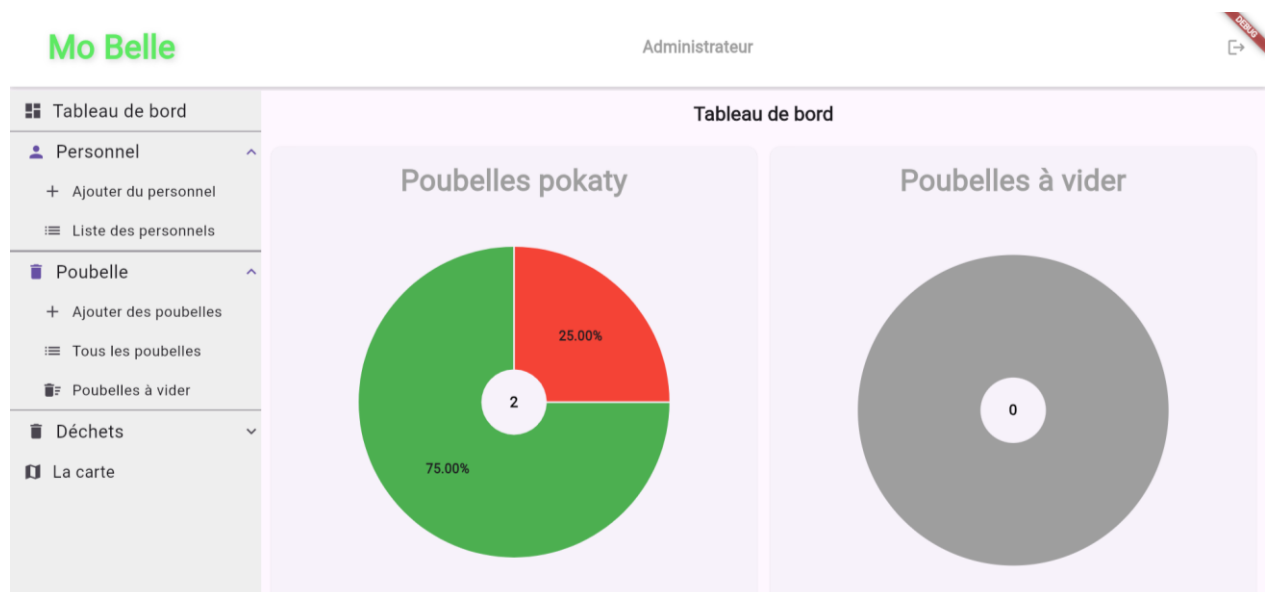
Mot de passe [Mot de passe oublié](#)

Se connecter

2.1. Administrateur

2.1.1. Tableau de bord

Le tableau de bord présente la répartition des déchets dans les poubelles : 75% de déchets organiques (en vert) et 25% de déchets chimiques (en rouge). À noter qu'aucune poubelle n'est actuellement à vider.



2.1.2. Ajouter du personnel

Le formulaire d'ajout du personnel permet de renseigner les informations nécessaires : nom, prénom, email, adresse, numéro de téléphone, rôle et mot de passe. Cliquez sur "Ajouter" pour enregistrer le nouveau membre du personnel.

The screenshot shows the 'Ajouter du personnel' form in the Mo Belle application. The interface includes a sidebar with navigation options: 'Tableau de bord', 'Personnel' (selected), 'Ajouter du personnel', and 'Liste des personnels'. The main content area is titled 'Ajouter du personnel' and contains several input fields: 'Nom', 'Prénom', 'Email', 'Adresse', 'Numéro de téléphone', 'Rôle' (a dropdown menu), and 'Mot de passe' (with a toggle for visibility). A green 'Ajouter' button is at the bottom of the form. The top of the page shows the 'Mo Belle' logo, the user role 'Administrateur', and a red 'nouveau' badge.

2.1.3. Liste des personnels

Le tableau liste les membres du personnel avec leurs informations : nom, prénom, email, adresse et numéro de téléphone. Une action de suppression est disponible pour chaque membre via l'icône en rouge.

The screenshot shows the 'Liste des personnels' table in the Mo Belle application. The interface is similar to the previous one, with the 'Liste des personnels' option selected in the sidebar. The table has the following columns: 'Nom', 'Prénom', 'Email', 'Adresse', 'Numéro de téléphone', and 'Action'. There is one row of data for a person named RANDRIA. The 'Action' column contains a red trash icon for deleting the record. The top of the page shows the 'Mo Belle' logo, the user role 'Administrateur', and a red 'nouveau' badge.

Nom	Prénom	Email	Adresse	Numéro de téléphone	Action
RANDRIA	Samuela	nathancarlinot002@gmail.com	0203Q0020 Antanimalandy	58213984	

2.1.4. Ajouter des poubelles

Le formulaire permet d'ajouter des poubelles en renseignant le nom, le poids maximum (en kg), le volume (en m³) et la localisation. Cliquez sur "Ajouter" pour enregistrer la nouvelle poubelle.

The screenshot shows the 'Ajouter des poubelles' form in the Mo Belle application. The left sidebar contains a menu with 'Tableau de bord', 'Personnel', 'Poubelle' (selected), 'Ajouter des poubelles', 'Tous les poubelles', 'Poubelles à vider', 'Déchets', and 'La carte'. The main content area is titled 'Ajouter des poubelles' and contains a form with four input fields: 'Nom de la poubelle', 'Poids maximum (en kg)', 'Volume (en m³)', and 'Localisation'. A green 'Ajouter' button is at the bottom of the form. The top right of the page shows 'Administrateur' and a red 'nouveau' badge.

2.1.5. Les listes des poubelles disponibles

Le tableau affiche la liste des poubelles avec leurs détails : ID, nom, localisation, poids maximum (kg), volume (m³), poids utilisé, volume utilisé, pourcentage de déchets organiques et chimiques. Une action de suppression est disponible pour chaque poubelle via l'icône en rouge.

The screenshot shows the 'Les listes des poubelles' table in the Mo Belle application. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Les listes des poubelles' and contains a table with 11 columns: ID, Nom, Localisation, Poids (kg), Volume (m³), Poids utilisé, Volume utilisé, Organiques, Chimiques, and Action. The table lists three poubelles: Maria, Nathan, and Ben. Each row has a red trash icon in the Action column. The top right of the page shows 'Administrateur' and a red 'nouveau' badge.

ID	Nom	Localisation	Poids (kg)	Volume (m³)	Poids utilisé	Volume utilisé	Organiques	Chimiques	Action
mob001	Maria	Chambre 2	12	12	3.10	2.40	66.67%	33.33%	
mob002	Nathan	Chambre 1	10	10	4.30	5.50	80.00%	20.00%	
mob003	Ben	Garage	2	1	0	0	0%	0%	

2.1.6. Les poubelles à vider

Le tableau indique les poubelles à vider avec les informations suivantes : ID, nom, localisation, poids maximum (kg), volume (m³), poids utilisé, volume utilisé, pourcentage de

déchets organiques et chimiques. Une action de vidage est disponible pour chaque poubelle via le bouton "Vider".

Mo Belle

Administrateur

Tableau de bord

Personnel

Poubelle

Ajouter des poubelles
Tous les poubelles
Poubelles à vider

Déchets

La carte

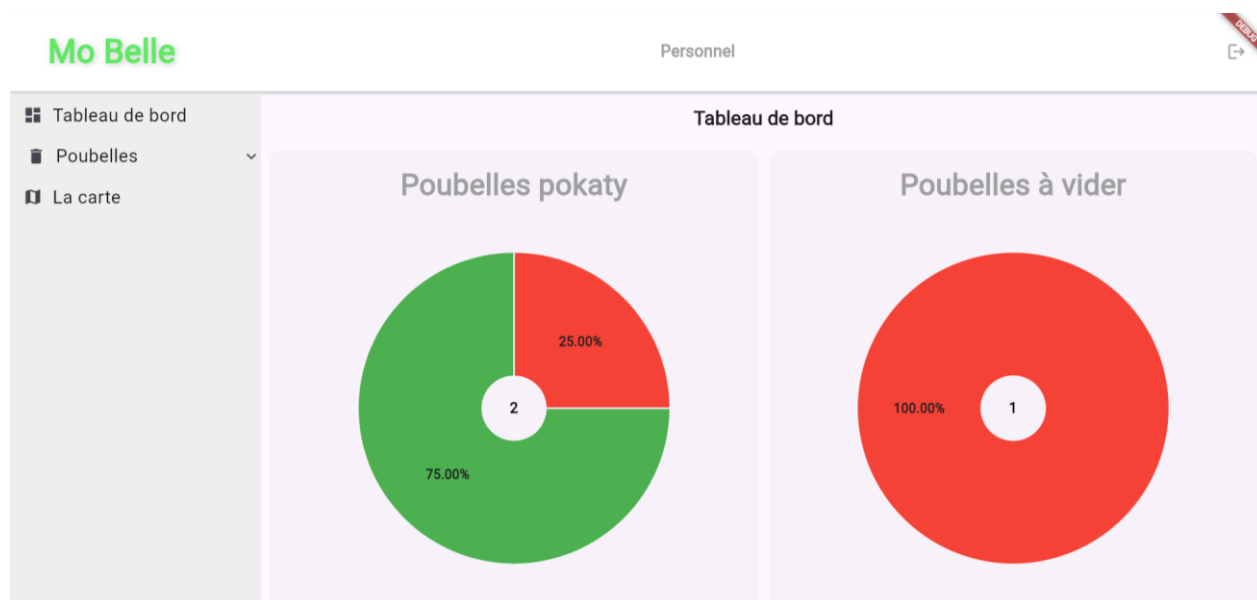
Les poubelles à vider

ID	Nom	Localisation	Poids (kg)	Volume (m³)	Poids utilisé	Volume utilisé	Organiques	Chimiques	Action
mob003	Ben	Garage	2	1	2	1	0%	100%	Vider

2.2. Personnel

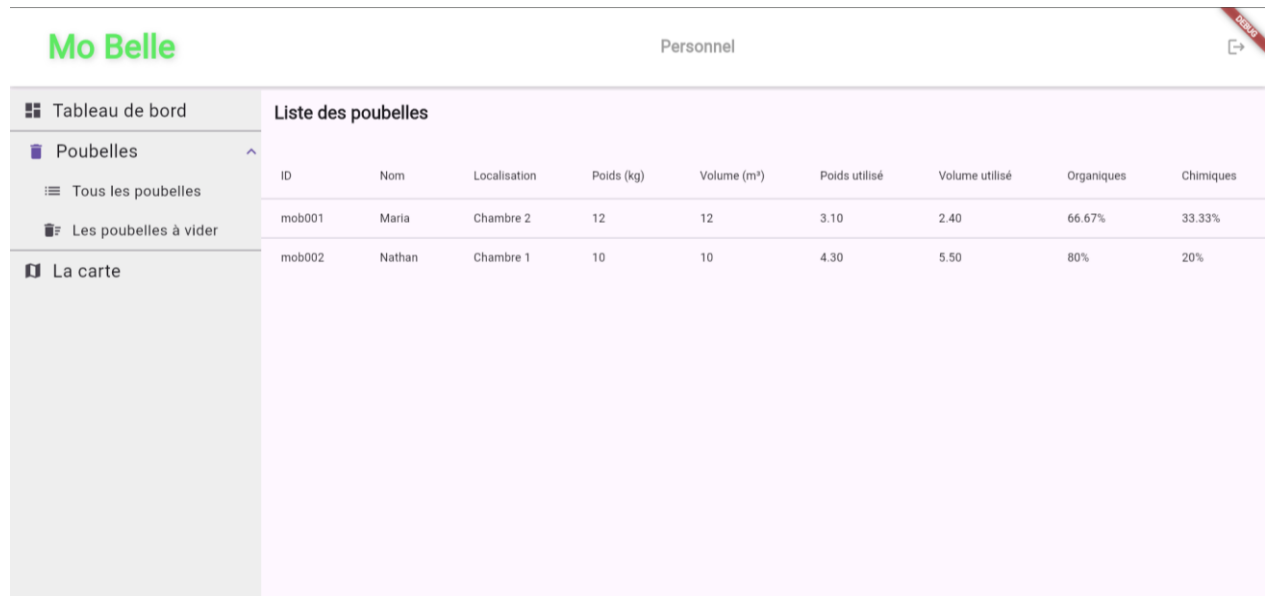
2.2.1. Tableau de bord

Le tableau de bord montre la répartition des déchets dans les poubelles : 75% de déchets organiques (en vert) et 25% de déchets chimiques (en rouge). Actuellement, une poubelle doit être vidée, représentant 100% des poubelles à vider.



2.2.2. Les listes des poubelles disponibles

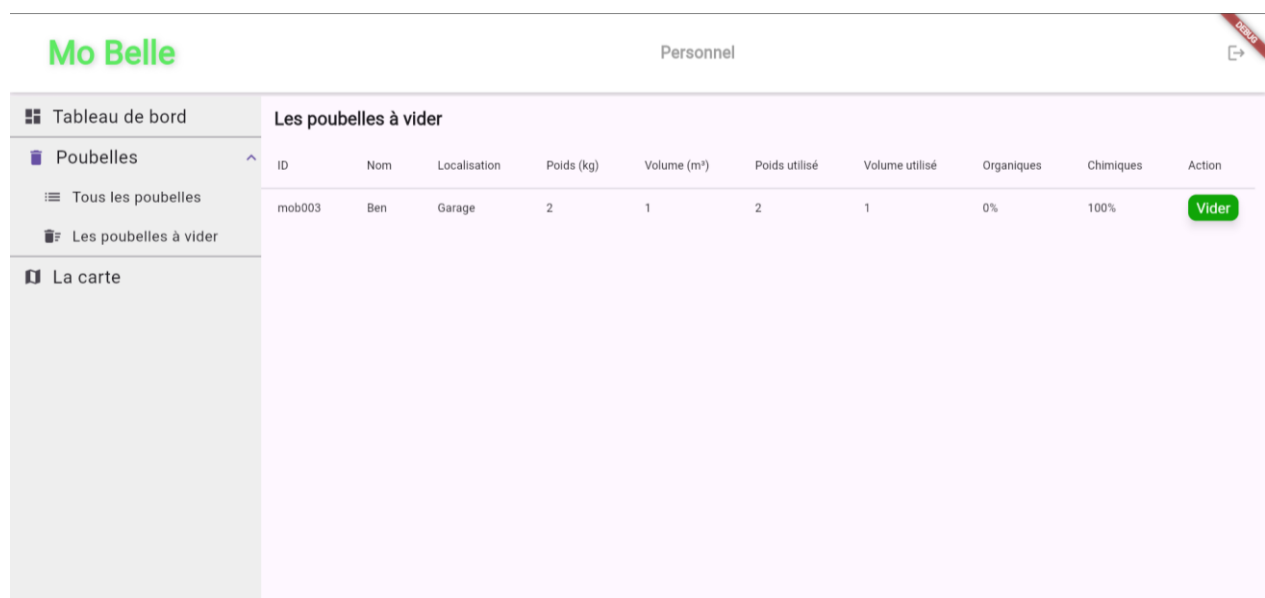
Le tableau affiche la liste des poubelles avec leurs détails : ID, nom, localisation, poids maximum (kg), volume (m³), poids utilisé, volume utilisé, pourcentage de déchets organiques et chimiques. Notons que les membres du personnel n'ont pas accès à la suppression des poubelles.



Mo Belle		Personnel							
Tableau de bord	Liste des poubelles								
Poubelles	ID	Nom	Localisation	Poids (kg)	Volume (m³)	Poids utilisé	Volume utilisé	Organiques	Chimiques
Tous les poubelles	mob001	Maria	Chambre 2	12	12	3.10	2.40	66.67%	33.33%
Les poubelles à vider	mob002	Nathan	Chambre 1	10	10	4.30	5.50	80%	20%
La carte									

2.2.3. Les poubelles à vider

Le tableau indique les poubelles à vider avec les informations suivantes : ID, nom, localisation, poids maximum (kg), volume (m³), poids utilisé, volume utilisé, pourcentage de déchets organiques et chimiques. Une action de vidage est disponible pour chaque poubelle via le bouton "Vider".



Mo Belle		Personnel								
Tableau de bord	Les poubelles à vider									
Poubelles	ID	Nom	Localisation	Poids (kg)	Volume (m³)	Poids utilisé	Volume utilisé	Organiques	Chimiques	Action
Tous les poubelles	mob003	Ben	Garage	2	1	2	1	0%	100%	Vider
Les poubelles à vider										
La carte										

2.3. Utilisateur

2.3.1. Accueil

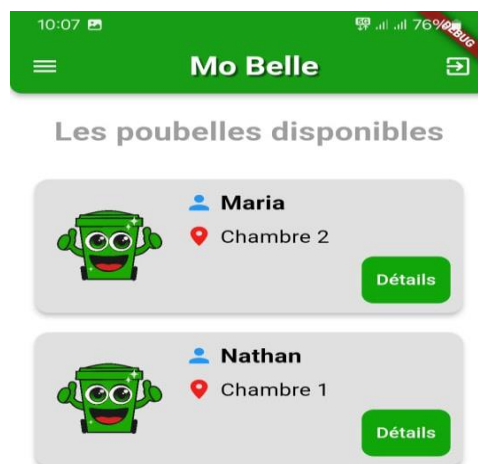
L'accueil utilisateur affiche un message inspirant : "Une route propre ne dépend pas seulement de l'efficacité du service de nettoyage, mais aussi de l'éducation des personnes qui y passent." Un bouton "Les poubelles disponibles" permet de consulter les poubelles accessibles.



2.3.2. Les poubelles disponibles

L'écran affiche les poubelles disponibles avec leur localisation respective. Pour chaque poubelle, un bouton "Détails" permet d'obtenir plus d'informations :

- **Maria** : Chambre 2
- **Nathan** : Chambre 1



2.3.3. Détails

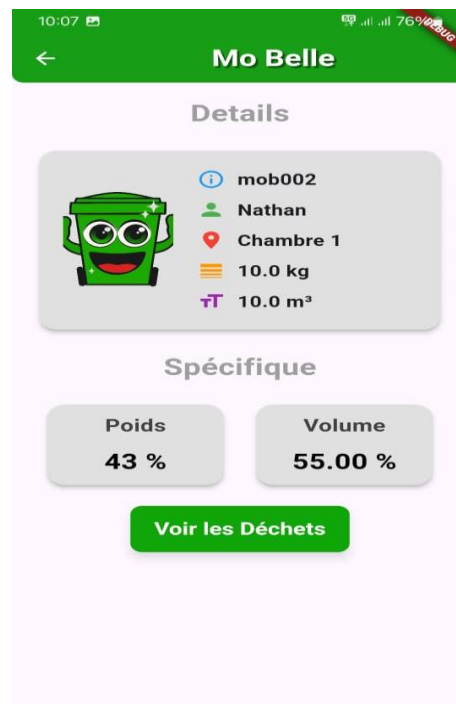
L'écran des détails affiche les informations spécifiques de la poubelle :

- **ID** : mob002
- **Nom** : Nathan
- **Localisation** : Chambre 1
- **Poids maximum** : 10.0 kg
- **Volume** : 10.0 m³

Détails spécifiques :

- **Poids utilisé** : 43%
- **Volume utilisé** : 55%

Un bouton "Voir les Déchets" permet de consulter les types de déchets présents dans cette poubelle.



2.3.4. Voir les déchets

L'écran des déchets affiche les détails des objets dans la poubelle :

- **Boîte en carton** : Masse 0.1 kg, Volume 0.2 m³, Type : Chimique
- **Bananes** : Masse 0.1 kg, Volume 0.4 m³, Type : Organique
- **Tomates** : Masse 0.2 kg, Volume 0.1 m³, Type : Organique
- **Œufs** : Masse 0.1 kg, Volume 0.5 m³, Type : Organique
- **Chaussures** : Masse 2.0 kg, Volume 1.0 m³, Type : Chimique
- **Oranges** : Masse 1.0 kg, Volume 1.0 m³, Type : Organique
- **Bouteille en plastique** : Masse 2.0 kg, Volume 3.0 m³, Type : Chimique

Un bouton "Mise à jour" permet d'actualiser les informations des déchets.



4- Conclusion

Pour conclure, cette section a mis en lumière les performances et l'efficacité du système "Mo Belle". Les résultats obtenus montrent un potentiel significatif pour améliorer la gestion des déchets grâce à l'automatisation intelligente, tout en soulignant les domaines où des améliorations peuvent encore être apportées pour atteindre une performance optimale.

Chapitre 6 : Evaluation

1- Introduction

Dans cette section, nous nous penchons sur l'évaluation et les résultats du projet "Mo Belle". Nous analyserons les performances du système mis en place, les retours des utilisateurs, et les impacts environnementaux et économiques de cette solution innovante. Cette évaluation permet de mesurer l'efficacité du projet et d'identifier les axes d'amélioration pour les développements futurs.

2- Problème rencontré

2.1. Intégration des Capteurs

Nous avons rencontré des difficultés lors de l'intégration des différents capteurs (ultrason, infrarouge et HX711) avec l'ESP32. Les interférences et les incompatibilités entre les capteurs ont nécessité plusieurs ajustements et calibrations pour obtenir des lectures précises et fiables.

2.2. Connexion Wi-Fi Instable

La connexion Wi-Fi de l'ESP32 s'est avérée instable par moments, surtout lors des tentatives de connexion automatique après une déconnexion. Ce problème a affecté la continuité des communications entre le microcontrôleur et le réseau, nécessitant la mise en place de mécanismes de reconnexion robustes.

2.3. Sécurité des Données

Assurer la sécurité des données transmises et stockées, notamment les informations collectées par les capteurs et les actions exécutées par le système, nous a nécessité la mise en place de protocoles de sécurité robustes pour prévenir toute tentative d'accès non autorisé ou de manipulation des données.

3- Amélioration dans les futures

3.1. Optimiser la Consommation d'Énergie

Nous devons travailler sur l'optimisation de la consommation d'énergie de notre système. Cela pourrait inclure l'intégration de modes de sommeil pour l'ESP32 et les capteurs, ainsi que l'utilisation de sources d'alimentation plus efficaces ou de panneaux solaires pour prolonger l'autonomie du dispositif.

3.2. Améliorer la Précision des Capteurs

Nous devons continuer à améliorer la précision des capteurs, en particulier ceux utilisés pour mesurer la distance et le poids. Cela peut inclure l'utilisation de capteurs de meilleure qualité, l'optimisation des algorithmes de calibration et la réduction des interférences environnementales.

3.3. Renforcer la Sécurité des Données

Nous devons renforcer les mesures de sécurité pour protéger les données transmises et stockées par notre système. Cela inclut la mise en œuvre de protocoles de chiffrement plus robustes, l'authentification des appareils et la surveillance continue des activités pour détecter et prévenir toute tentative d'accès non autorisé.

4- Conclusion

Pour résumer, l'évaluation du projet "Mo Belle" démontre des résultats positifs en termes d'efficacité et de satisfaction des utilisateurs. Les objectifs initiaux ont été atteints, offrant une solution durable et intelligente pour la gestion des déchets urbains. Les retours d'expérience et les données collectées serviront de base pour les améliorations continues, assurant ainsi la pérennité et l'optimisation du système.

Conclusion générale

