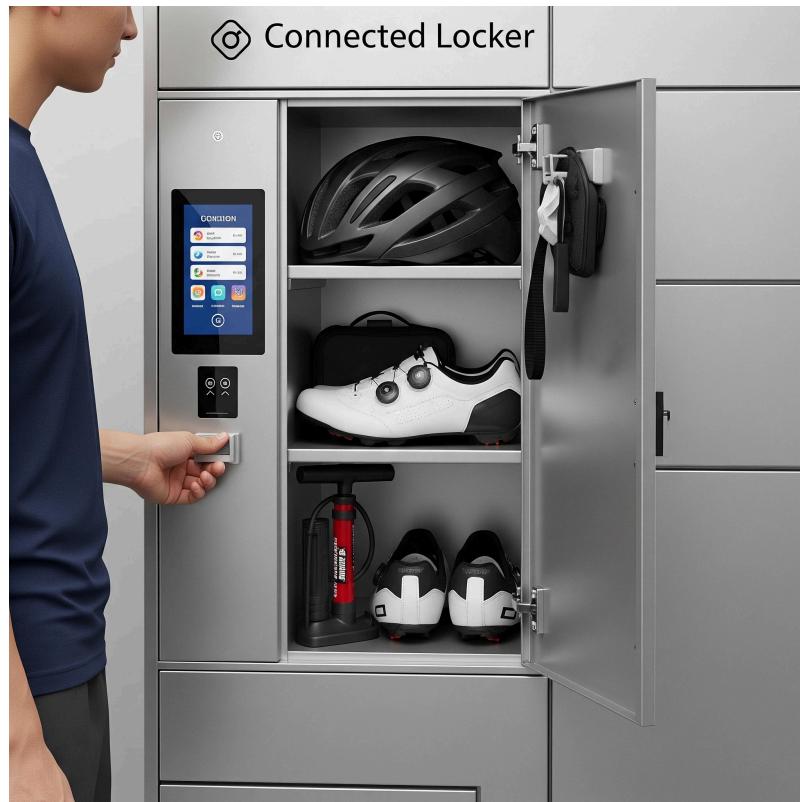


Connected Locker

Rapport de projet



Antoine Brousse, Soultane Sama-mola, Bastien Michel

Etablissement : La Salle Avignon / Avignon Université

SOMMAIRE

1 Introduction.....	4
1.1 Contexte du projet.....	4
1.2 Objectifs.....	5
1.3 Présentation de l'équipe.....	6
1.4 Méthodologie de travail.....	7
2 Étude préalable.....	8
2.1 Cahier des charges.....	8
2.2 Etude de marché.....	8
2.3 Choix techniques.....	9
3 Conception du projet.....	12
3.1 Architecture générale.....	12
3.2 Schémas et diagrammes.....	13
3.2.1 Diagramme de cas d'utilisation.....	13
3.2.2 Diagramme des exigences.....	14
3.3 Description des modules.....	15
3.3.1 IHM et Raspberry.....	15
3.3.2 Bloc casier.....	15
3.3.3 Serveur.....	15
3.3.4 Mécanique.....	15
4 Répartition du travail et contribution individuelle.....	16
4.1 Contribution de Soultane.....	16
4.1.1 Etude du cahier des charges.....	16
4.1.1.1 Authentification des utilisateurs.....	16
4.1.1.2 Base de donnée utilisateurs.....	19
4.1.1.3 Solution de transition vers Python/Flask.....	20
4.2 Contribution Antoine.....	21
4.2.1 Etude du cahier des charges.....	21
4.2.2 Recherche de pièces.....	21
4.2.3 Schéma structurel sous Proteus 8.....	22
4.2.4 Réalisation routage sur Ares.....	23
4.3 Contribution de Bastien.....	25
4.3.1 Développement de l'Interface Homme-Machine (IHM).....	25
4.3.2 Communication I2C.....	29

4.3.3 Gestion informatique et serveur.....	31
4.3.4 Électronique.....	33
4.3.5 Mécanique.....	34
4.3.6 Chef de projet.....	36
4.4 Collaboration et coordination dans l'équipe.....	37
5 Conclusion et perspectives.....	38
5.1 Bilan du projet.....	38
5.2 Améliorations possibles.....	39
5.2.1 Sécurité informatique.....	39
5.2.2 Sécurité électrique.....	39
5.2.3 Adressage Bus I2C.....	40
5.2.4 Mécanique.....	40
5.3 Conclusion.....	41
5.3.1 Conclusion de Soultane.....	41
5.3.2 Conclusion de Antoine.....	42
5.3.3 Conclusion de Bastien.....	43
6 Annexes.....	44
6.1 STB (Spécification Technique de Besoin).....	44
6.2 Etude de marché.....	50
6.3 Codes sources.....	54

1 Introduction

1.1 Contexte du projet

Avec le COVID, l'explosion de l'utilisation des vélos a chamboulé les pratiques dans les grandes villes. Les routes et les axes sont redessinés, la place est désormais laissée aux deux roues. Toutefois une problématique subsiste et inquiète encore bon nombre de nos vélocistes.

Où garer les vélos et comment les sécuriser ?

Les mairies se tournent légitimement vers les spécialistes du parkings et leur demandent de proposer des solutions concrètes et rapides. Les parkings mettent alors en place des stationnements sécurisés (local fermé accessible avec un badge) :



Pour obtenir celui-ci, l'utilisateur doit acheter un abonnement stationnement de vélo et attendre de recevoir son badge chez lui.

L'espace sécurisé contient des racks pour attacher les vélos, mais aussi une station de gonflage, un atelier de réparation et des casiers. Ces casiers sont actuellement difficilement attribuables et ne peuvent être facilement louables. C'est par une gestion de cadenas compliquée que le parking donne un accès au client. L'objet de notre solution est de proposer une alternative connecté à cette gestion de casier.

1.2 Objectifs

L'usager a accès au local fermé via un accès badge RFID.



Ce même badge devra être utilisé pour ouvrir le casier. En option, une connexion NFC pourra aussi être envisagée.

Une fois le casier ouvert, l'usager peut poser ses affaires et le refermer.

Un casier peut aussi contenir une prise électrique qui est alimentée sur demande par l'utilisateur depuis un site internet ou une appli mobile.



Le casier se verrouille automatiquement lorsqu'il est refermé. Si jamais un usager oublie son badge à l'intérieur du casier, il devra alors taper un code à 6 chiffres (lié à son compte) pour le déverrouiller. Il pourra aussi l'ouvrir depuis internet ou son application mobile et une connexion 4G sera nécessaire pour la communication smartphone (disponible dans certains parkings).

Le Back office proposé devra à tout moment pouvoir connaître l'état (ouvert/fermé/en charge/non en charge, capteur température) du casier.

Il sera aussi possible de consulter les informations du client (code/N° de casier).

Un capteur permet de mesurer la température et de remonter une alerte si la chaleur est trop élevée et que la prise est alimentée.

Les missions du système casier connecté sont de proposer une gestion simplifiée des casiers connectés pour les parkings à vélos, avec un accès sécurisé via RFID ou application mobile, et un suivi en temps réel de l'état des casiers.

- Conception du système
- Développement de l'application mobile et/ou du site internet
- Développement et intégration des casiers connectés
- Mise en place de la communication réseau
- Support et maintenance

1.3 Présentation de l'équipe

Le projet a été mené par une équipe composée de trois membres aux compétences complémentaires, permettant une répartition efficace des tâches et une bonne dynamique de travail tout au long du développement.

- **Soultane Samamola** – Responsable informatique

Soultane Samamola a été chargée du développement logiciel du projet. Il a assuré la programmation des différents modules nécessaires au fonctionnement du système, ainsi que l'intégration entre l'interface homme-machine (IHM) et les éléments matériels. Il a également participé aux tests fonctionnels et à la mise au point du code pour garantir la fiabilité et la performance des applications logicielles.

- **Antoine Brousse** – Responsable électronique / Responsable mécanique

Antoine Brousse a assuré la conception et le développement des cartes électroniques du projet, depuis l'analyse des besoins jusqu'aux phases de tests et de validation. Il a également supervisé les aspects mécaniques du système, la fabrication du boîtier et l'intégration physique des composants, tout en veillant à la compatibilité structurelle de l'ensemble.

- **Bastien Michel** – Chef de projet / Conception IHM / Documentation

En tant que chef de projet, Bastien Michel a assuré la coordination globale du projet, le suivi de l'avancement des tâches et la bonne répartition du travail au sein de l'équipe. Il a également pris en charge la conception de l'interface homme-machine (IHM), en veillant à l'ergonomie et à la cohérence graphique, ainsi que la rédaction de la documentation technique et du rapport de projet.

Cette répartition claire des responsabilités a permis à l'équipe de mener à bien le projet dans les délais impartis, tout en assurant une qualité technique et documentaire conforme aux attentes.

1.4 Méthodologie de travail

Pour mener à bien ce projet en trinôme, nous avons adopté une méthodologie de travail collaborative et structurée. Dès le début, nous avons défini un cadre clair de communication, de répartition des tâches et de suivi de l'avancement.

Afin d'organiser efficacement notre travail, nous avons utilisé ClickUp.



Il s'agit d'un outil de gestion de projet en ligne, celui-ci nous a permis de :

- Créer un tableau de bord dédié au projet, avec une vue d'ensemble des tâches à réaliser.
- Répartir les tâches entre les membres de l'équipe en fonction des compétences et disponibilités de chacun.
- Suivre l'état d'avancement de chaque tâche à l'aide de colonnes "À faire", "En cours", et "Terminée".
- Commenter et échanger directement sur chaque tâche, ce qui a facilité la communication asynchrone, en particulier en dehors des heures de travail en présentiel.
- Planifier des objectifs hebdomadaires pour garder un rythme constant tout au long du projet.

Nous avons également organisé des réunions régulières, souvent en début ou en fin de séance, afin de faire un point rapide sur l'avancement, résoudre les éventuels blocages techniques, et ajuster les priorités.

Cette méthode de gestion nous a permis de travailler de manière fluide et organisée, tout en assurant un bon équilibre dans la charge de travail entre les membres du groupe.

2 Étude préalable

2.1 Cahier des charges

Dans un premier temps, nous avons rédigé la STB (Spécification Technique de Besoin) disponible en annexe, un document essentiel qui nous a permis de poser les bases du projet.

Cette étape a consisté à analyser et formaliser les attentes du client ou des utilisateurs finaux, en identifiant de manière claire les besoins fonctionnels, les contraintes techniques, les objectifs de performance, ainsi que les limites du système à développer. La rédaction de cette STB nous a ainsi permis de cadrer le projet, de partager une vision commune au sein de l'équipe, et de faciliter la prise de décisions techniques lors des phases suivantes.

Annexe : [6.1 STB \(Spécification Technique de Besoin\)](#)

2.2 Etude de marché

Par la suite, nous avons mené une étude de marché afin d'évaluer l'environnement concurrentiel, les solutions existantes et les tendances technologiques liées à notre projet.

Cette étape nous a permis d'identifier les produits similaires déjà disponibles, d'analyser leurs caractéristiques techniques, leurs avantages et leurs limites, mais également de repérer les opportunités d'innovation ou d'amélioration.

Enfin, elle a constitué un appui solide pour justifier nos choix techniques et stratégiques dans la suite du projet.

Annexe : [6.2 Etude de marché](#)

2.3 Choix techniques

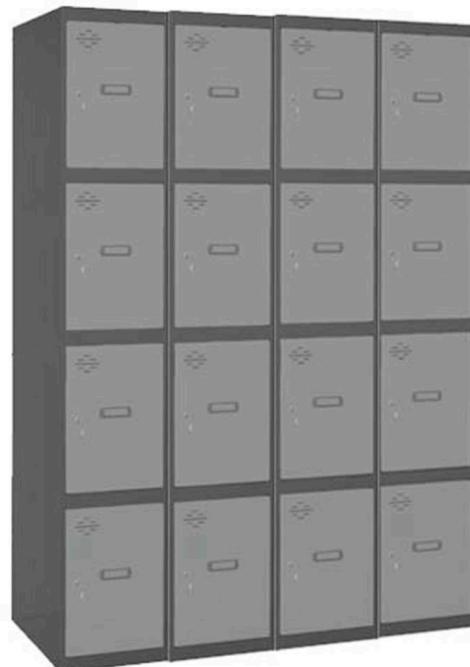
Les choix techniques ont été faits en tenant compte des objectifs du projet, des contraintes matérielles et logicielles, ainsi que des compétences de chaque membre du trinôme.

Nous avons sélectionné des langages et des outils que nous maîtrisons déjà partiellement et que nous avions vus durant la formation afin de gagner en efficacité tout en consolidant nos connaissances. Parmi eux :

- C++ pour le développement principal des Esp32, en raison de sa simplicité d'utilisation et de sa large documentation.
- Python choisi pour sa compatibilité avec notre projet et ses performances.
- Sqlite
- HTML/CSS/JAVA/PWA

Nous disposons déjà d'un casier physique existant, ce qui a constitué une base concrète pour le développement de notre projet de casier connecté.

Cela nous a permis de nous concentrer principalement sur l'intégration des composants électroniques et du système de contrôle, sans avoir à concevoir ou fabriquer la structure du casier. Ce choix nous a fait gagner du temps et nous a permis de nous focaliser sur les aspects techniques du projet (verrouillage, communication, interface, etc.).



Choix des composants et outils de développement :

- **ESP32** : Ce microcontrôleur a été retenu pour sa puissance de calcul, son faible coût et sa large compatibilité avec nos capteurs et composants. Il sera intégré dans chaque casier afin d'assurer la connectivité.



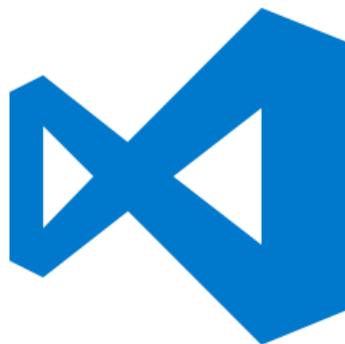
- **Raspberry Pi** associé à un écran tactile 7" : Ce duo a été sélectionné pour servir d'interface utilisateur principale à l'entrée du système de casiers, mais aussi pour assurer les fonctions de serveur central et de gestion de la base de données.



Pour assurer le bon fonctionnement et la supervision des casiers, plusieurs capteurs ont été sélectionnés :

- **Capteur de fin de course (GTIWUNG)** : utilisé pour détecter la fermeture effective de la porte du casier.
- **Capteur de courant** : permet de mesurer la consommation électrique de la prise intégrée dans le casier, utile pour surveiller l'usage ou détecter une anomalie.
- **Capteur NFC (PN532)** : employé pour l'identification et l'ouverture du casier via une carte ou badge NFC.
- **Gâche électrique (ATOPLEE, 12 V 2 A)** : verrou à ressort en acier au carbone (73 × 58 × 13 mm), utilisé comme mécanisme de verrouillage du casier. Elle permet un verrouillage fiable, tout en assurant une ouverture contrôlée électroniquement à distance.

- **Environnement de développement :** Nous avons utilisé Visual Studio Code pour son interface intuitive, ses nombreuses extensions et sa compatibilité avec divers langages et bibliothèques. Proteus a également été utilisé pour la conception et la simulation des cartes électroniques.



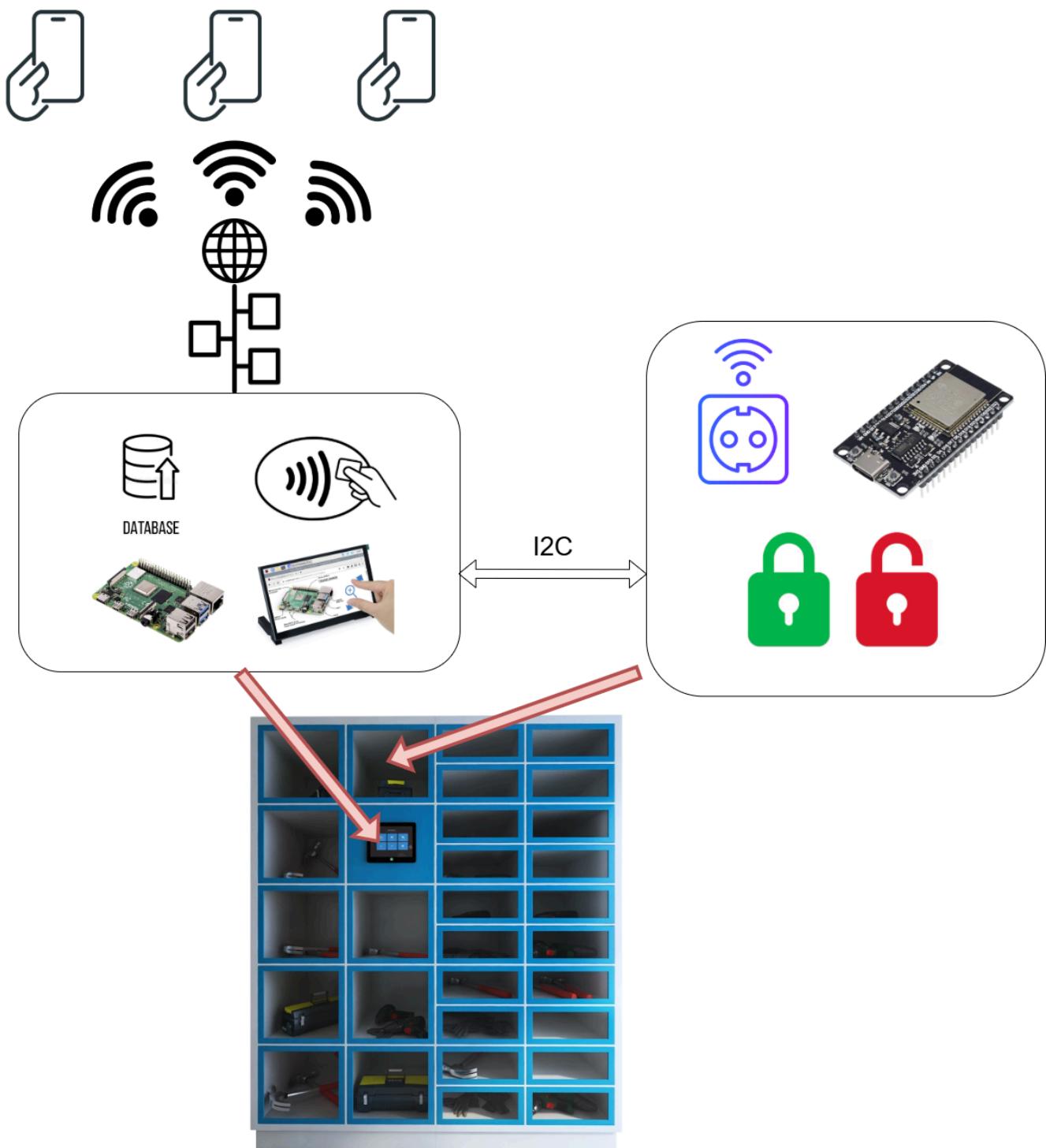
- **Système de versionnage :** Même si nous n'avons pas utilisé Git de façon poussée, nous avons conservé des versions manuelles régulières de notre travail pour prévenir toute perte de données.



Ces choix techniques nous ont permis d'allier simplicité, efficacité et fiabilité dans la conception du projet, tout en restant dans le cadre des compétences et des ressources disponibles.

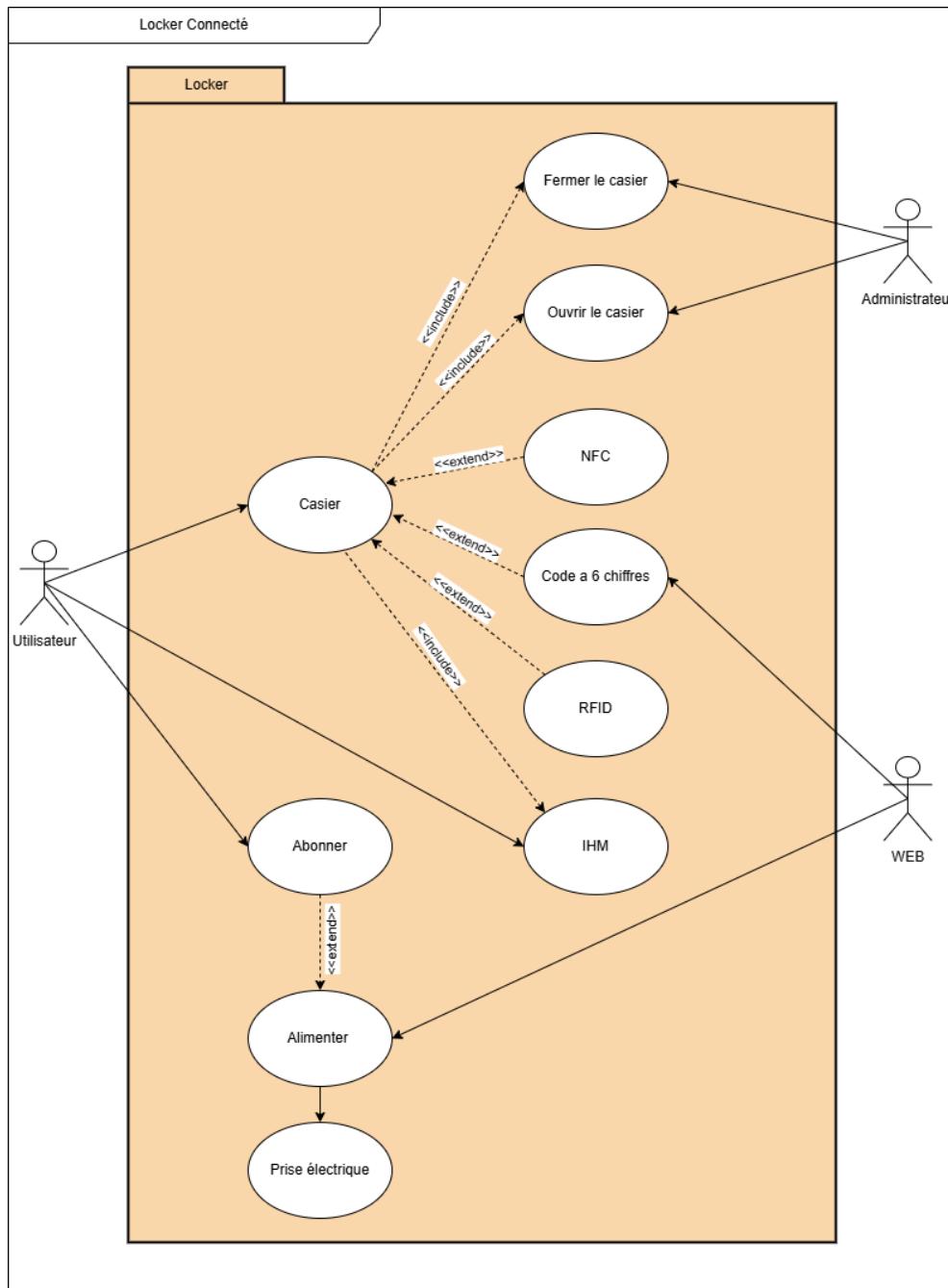
3 Conception du projet

3.1 Architecture générale



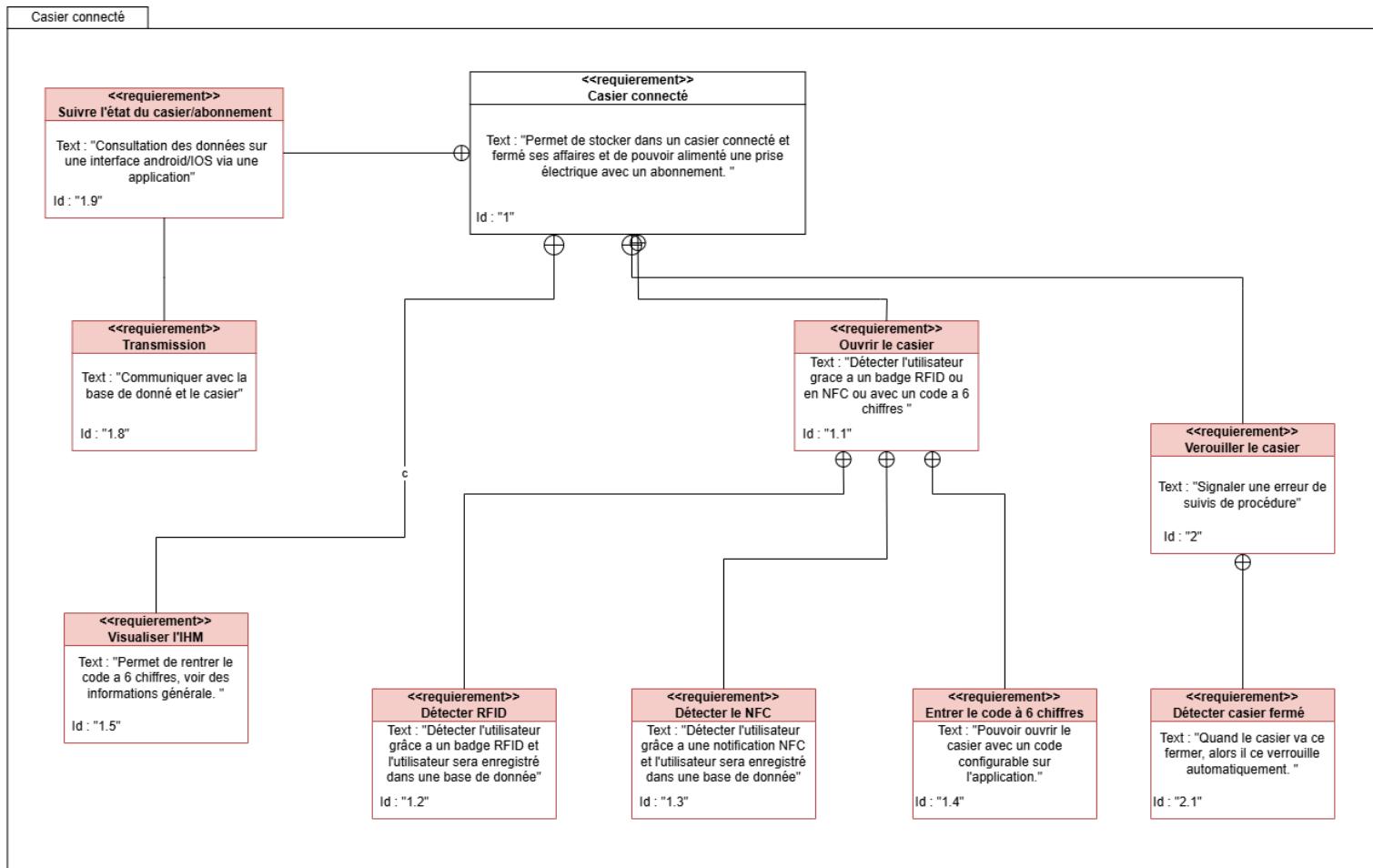
3.2 Schémas et diagrammes

3.2.1 Diagramme de cas d'utilisation



Ce diagramme des cas d'utilisation représente les différentes interactions entre les utilisateurs (administrateur et utilisateur classique) et le système de casiers connectés, en illustrant les principales fonctionnalités accessibles comme l'authentification, l'ouverture des casiers, etc.

3.2.2 Diagramme des exigences



Le diagramme des exigences a été réalisé en amont du développement afin de formaliser les besoins fonctionnels et techniques du projet.

Ce diagramme permet d'identifier clairement ce que le système doit accomplir, aussi bien du point de vue de l'utilisateur que du point de vue technique.

On y retrouve, par exemple, des exigences comme la connexion sécurisée des utilisateurs via un code PIN.

Ce travail d'analyse a permis d'organiser les tâches, de mieux répartir les responsabilités au sein de l'équipe, et de s'assurer que toutes les fonctionnalités essentielles soient prises en compte dans la phase de conception.

3.3 Description des modules

Avant la réalisation des différents modules, une phase d'étude a été menée afin de répartir efficacement les tâches entre les membres du groupe.

Grâce à l'utilisation de l'outil de gestion de projet ClickUp, chaque membre a pu se voir attribuer des éléments à analyser, comprendre ou concevoir, en lien avec les modules à développer.

Cette organisation a permis d'optimiser la collaboration et de s'assurer que chaque partie du projet soit bien étudiée avant sa mise en œuvre technique.

3.3.1 IHM et Raspberry

L'interface homme-machine (IHM), développée en HTML/CSS/JavaScript et affichée via un navigateur sur un Raspberry Pi 4, permet aux utilisateurs de s'identifier afin d'ouvrir un casier. Le Raspberry Pi communique avec le serveur Flask au moyen de requêtes HTTP POST, avec une gestion des sessions et de l'authentification

3.3.2 Bloc casier

Le bloc casier est contrôlé par un ESP32 connecté au Raspberry Pi via le bus I2C. Lorsqu'un utilisateur s'authentifie avec succès, une commande est envoyée depuis le Raspberry Pi vers l'ESP32 pour déverrouiller ou verrouiller le casier. Le système est conçu pour garantir que chaque utilisateur accède uniquement à son propre casier à l'aide d'un code PIN à 6 chiffres.

3.3.3 Serveur

Le serveur est développé en Python avec le framework Flask. Il gère l'authentification des utilisateurs, la modification des codes PIN, l'enregistrement de nouveaux comptes et la gestion de session via cookies. Il utilise une base de données SQLite pour stocker les informations utilisateurs (nom, mot de passe, PIN). Le serveur est également compatible avec les requêtes cross-origin grâce à Flask-CORS avec prise en charge des sessions.

3.3.4 Mécanique

La partie mécanique comprend principalement deux pièces réalisées en impression 3D à l'aide de Tinkercad.

La première pièce est un support destiné à maintenir l'écran du Raspberry Pi en façade, posé sur le casier central pour une meilleure accessibilité de l'IHM.

La seconde pièce permet de fixer proprement la carte électronique contenant l'ESP32, placée à l'arrière d'une prise électrique intégrée dans chaque casier.

4 Répartition du travail et contribution individuelle

4.1 Contribution de Soultane

4.1.1 Etude du cahier des charges

4.1.1.1 Authentification des utilisateurs

Cette partie du projet a pour objectif principal de concevoir et de mettre en place un **système d'authentification utilisateur** à la fois simple, fonctionnel et sécurisé. Le système doit permettre aux utilisateurs de s'**inscrire**, de **se connecter** à un espace sécurisé et de **se déconnecter** proprement, en respectant les bonnes pratiques de sécurité.

Dans un premier temps, le développement a été réalisé en **local**, afin de faciliter les phases de test et de débogage. Pour cela, nous avons utilisé un **environnement de développement local**, constitué principalement de **phpMyAdmin** (via un serveur local XAMPP) pour l'administration de la base de données, ainsi que les langages **PHP**, **HTML** et **CSS** pour la conception de l'application web.

- **PHP** a été utilisé pour la **logique serveur**, notamment pour le traitement des formulaires, la gestion des sessions et l'interaction avec la base de données.
- **HTML** a servi à définir la **structure des pages web** (formulaires d'inscription et de connexion).
- **CSS** a permis d'assurer une **mise en forme soignée et responsive**, garantissant une bonne expérience utilisateur sur différents types d'écrans (ordinateurs, tablettes, smartphones).

Page de Connexion

Connexion

soultane

.....

Se connecter

Pas encore inscrit ?

M'inscrire

Page d'Inscription

Inscription

Doe

John

....

8684

8684



M'inscrire

J'ai déjà un compte

Se connecter

Bonjour Doe, vous êtes maintenant connecté.

Connexion réussie !

Vous pouvez maintenant accéder à votre espace personnel.

[Se déconnecter](#)

4.1.1.2 Base de donnée utilisateurs

La gestion des données utilisateurs est assurée par une **base de données MySQL**, hébergée localement et administrée à l'aide de **phpMyAdmin**. Ce dernier facilite la création des tables, l'insertion et la consultation des enregistrements, ainsi que le suivi du bon fonctionnement de l'application. La **table des utilisateurs** stocke les informations essentielles telles que le nom d'utilisateur, le mot de passe, ce dernier étant **haché** à l'aide de la fonction `password_hash()` de PHP pour garantir la confidentialité des données et le PIN pour les casiers.

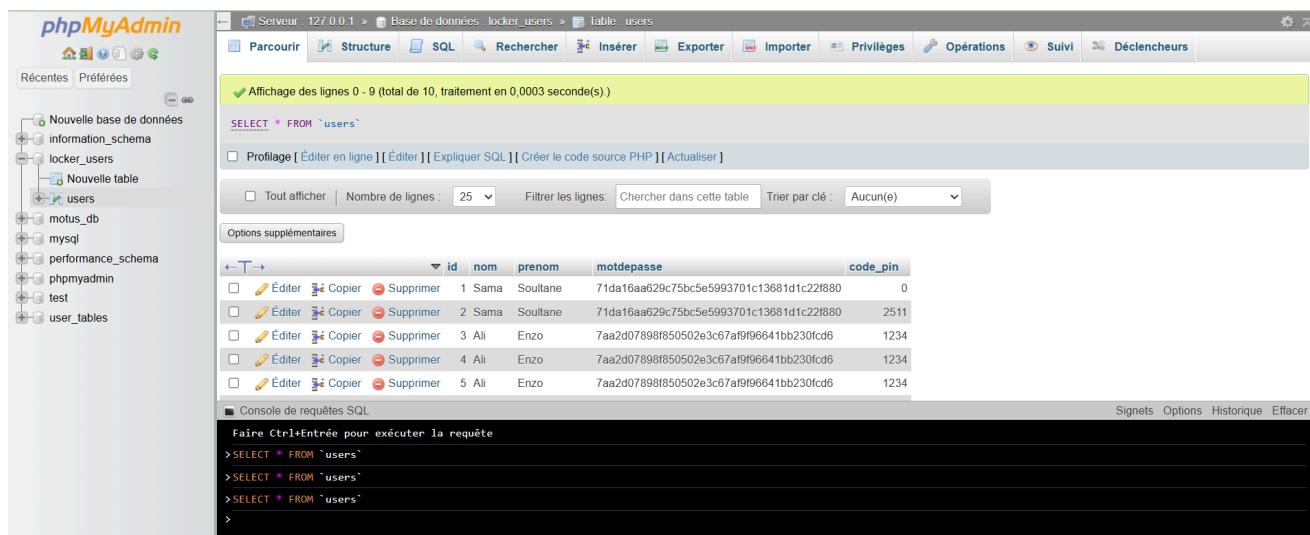


Table : users

Affichage des lignes 0 - 9 (total de 10, traitement en 0,0003 seconde(s))

SELECT * FROM `users`

	id	nom	prenom	motdepasse	code_pin
<input type="checkbox"/>	1	Sama	Soultane	71da16aa629c75bc5e5993701c13681d1c22f880	0
<input type="checkbox"/>	2	Sama	Soultane	71da16aa629c75bc5e5993701c13681d1c22f880	2511
<input type="checkbox"/>	3	Ali	Enzo	7aa2d07898f850502e3c67af9f96641bb230fcf6	1234
<input type="checkbox"/>	4	Ali	Enzo	7aa2d07898f850502e3c67af9f96641bb230fcf6	1234
<input type="checkbox"/>	5	Ali	Enzo	7aa2d07898f850502e3c67af9f96641bb230fcf6	1234

Faire Ctrl+Entrée pour exécuter la requête

```
>SELECT * FROM `users`
>SELECT * FROM `users`
>SELECT * FROM `users`
```

4.1.1.3 Solution de transition vers Python/Flask

Après cette phase de test, la prochaine phase est une transition vers une solution Python/Flask comme évoqué plus haut.

4.2 Contribution Antoine

4.2.1 Etude du cahier des charges

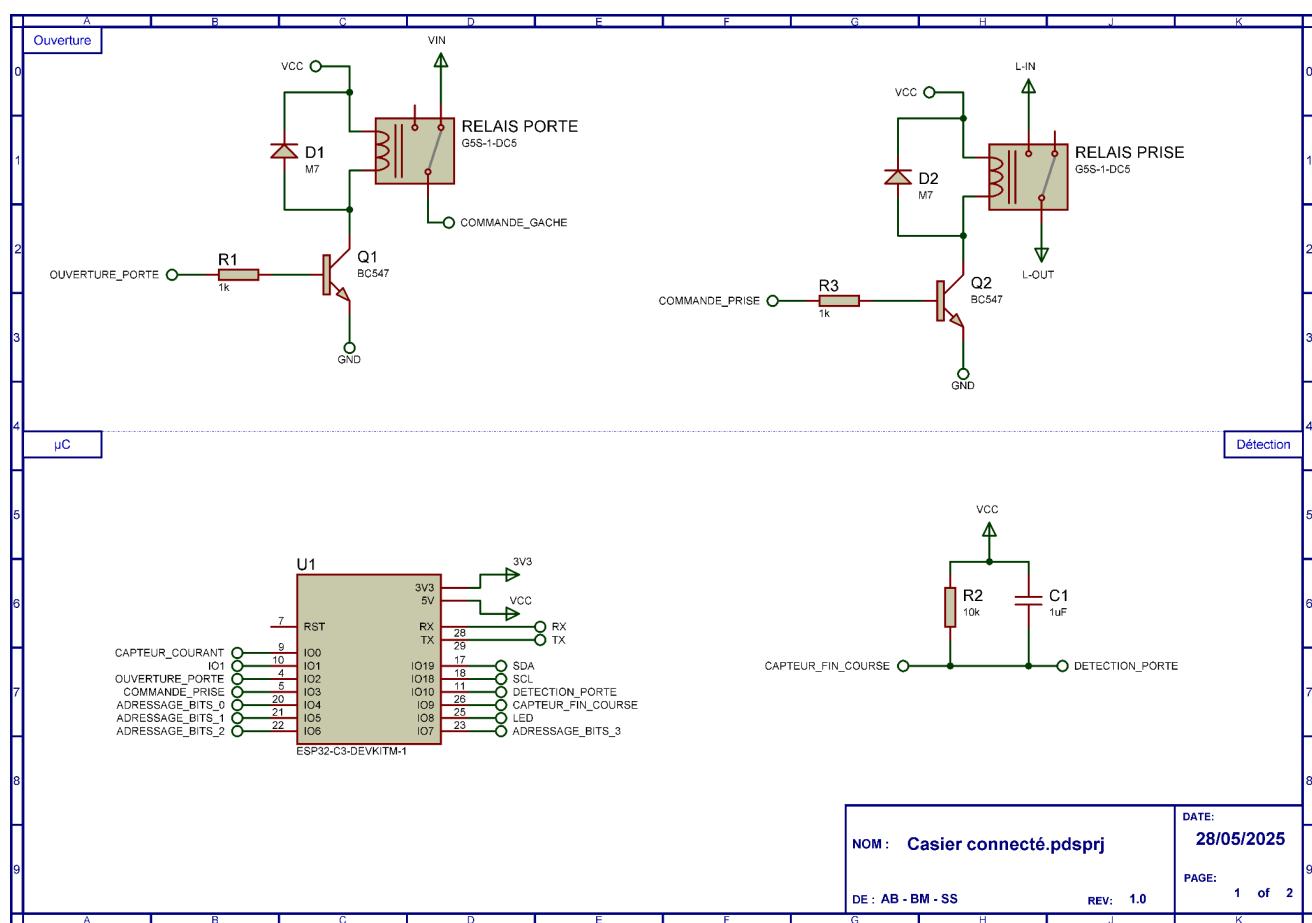
La phase initiale et fondamentale de ce projet a consisté en une étude approfondie du cahier des charges. Ce document, véritable feuille de route, a été minutieusement analysé pour en extraire toutes les exigences et contraintes. J'ai identifié les fonctions principales et secondaires que le système devait impérativement remplir, telles que la gestion du casier en général, la "location" de la prise électrique. Au-delà des fonctionnalités, j'ai également relevé les contraintes techniques. Cette analyse m'a permis de définir un ensemble de spécifications détaillées qui ont guidé toutes les décisions techniques ultérieures, assurant ainsi que le produit final serait en adéquation avec les attentes initiales.

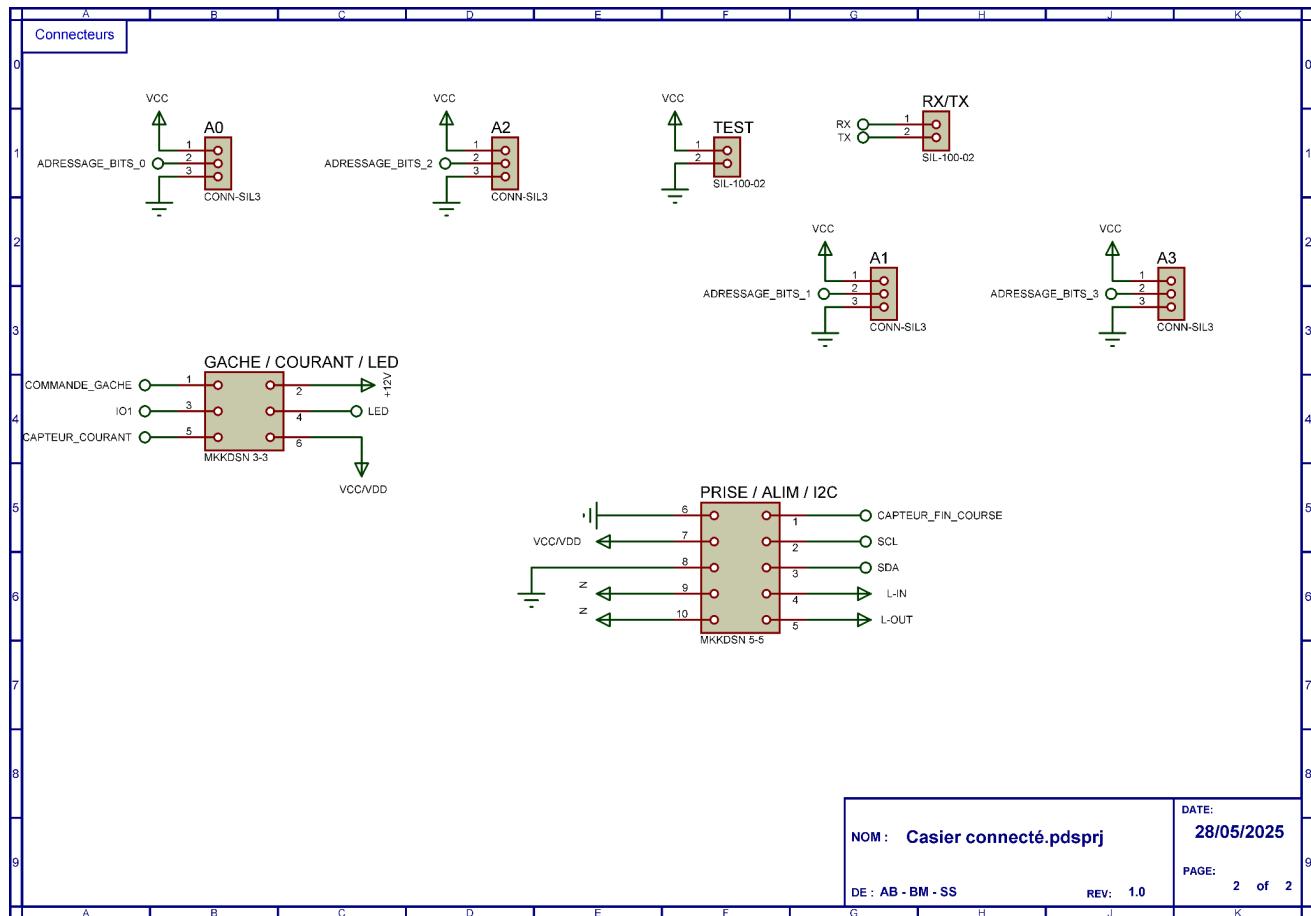
4.2.2 Recherche de pièces

Une fois les spécifications techniques clairement établies à partir du cahier des charges, l'étape suivante a été la recherche et la sélection des composants nécessaires à la matérialisation du projet. Cette phase a impliqué l'exploration de différentes fiches techniques de fabricants et de distributeurs pour identifier les références adéquates. Pour chaque fonction requise (microcontrôleur, interrupteur fin de course, relais, etc.), j'ai évalué plusieurs options en me basant sur des critères précis. La disponibilité des composants sur le marché, leur coût unitaire, mais aussi leurs caractéristiques techniques spécifiques (tensions de fonctionnement, courants maximaux, fréquences, encombrement, etc.) ont été nécessaires. Il a parfois fallu faire des choix stratégiques, pesant les avantages et inconvénients de différentes solutions pour trouver le meilleur compromis entre performance, fiabilité et contraintes budgétaires.

4.2.3 Schéma structurel sous Proteus 8

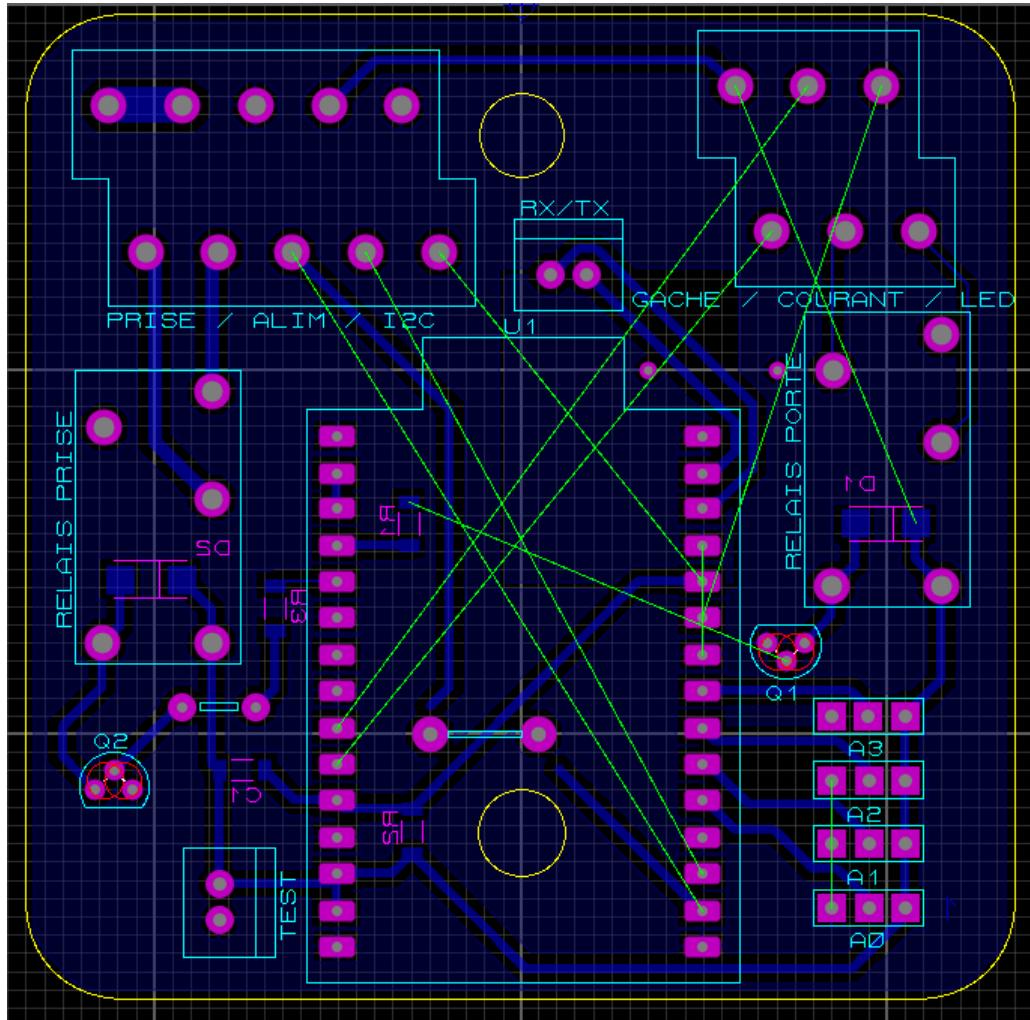
Avec la sélection des composants validée, j'ai procédé à la conception du schéma structurel de mon système à l'aide du logiciel Proteus 8. Cette étape représente la première matérialisation visuelle du circuit. J'ai commencé par placer chaque composant identifié lors de la phase de recherche sur l'espace de travail. Ensuite, j'ai établi toutes les interconnexions électriques entre eux, en respectant la logique de fonctionnement du circuit et les fiches techniques de chaque composant. Cela inclut le raccordement des alimentations, des masses, des lignes de données (par exemple, les bus I2C), des signaux de contrôle, et des entrées/sorties.





4.2.4 Réalisation routage sur Ares

Après la validation du schéma structurel, l'étape suivante a été la conception physique du circuit imprimé (PCB) via le module Ares de Proteus 8. Il s'agit de transformer le schéma logique en un plan de fabrication concret pour la carte électronique. Cette phase a débuté par le positionnement des composants sur le circuit, en tenant compte de contraintes d'encombrement, d'optimisation des longueurs de pistes pour les signaux critiques, de dissipation thermique, et d'accessibilité pour l'assemblage et la maintenance. Ensuite, j'ai tracé les pistes de cuivre qui connectent électriquement les broches des composants, en respectant scrupuleusement les règles de conception pour les PCB (largeur des pistes en fonction du courant, espacement entre les pistes, gestion des plans de masse, etc.). Cette étape demande une grande précision et une vision spatiale, car un routage mal optimisé peut entraîner des problèmes de performance (parasites, surchauffe) ou rendre la fabrication complexe. L'objectif était d'obtenir un routage propre, compact et fonctionnel, prêt pour une éventuelle fabrication de la carte.



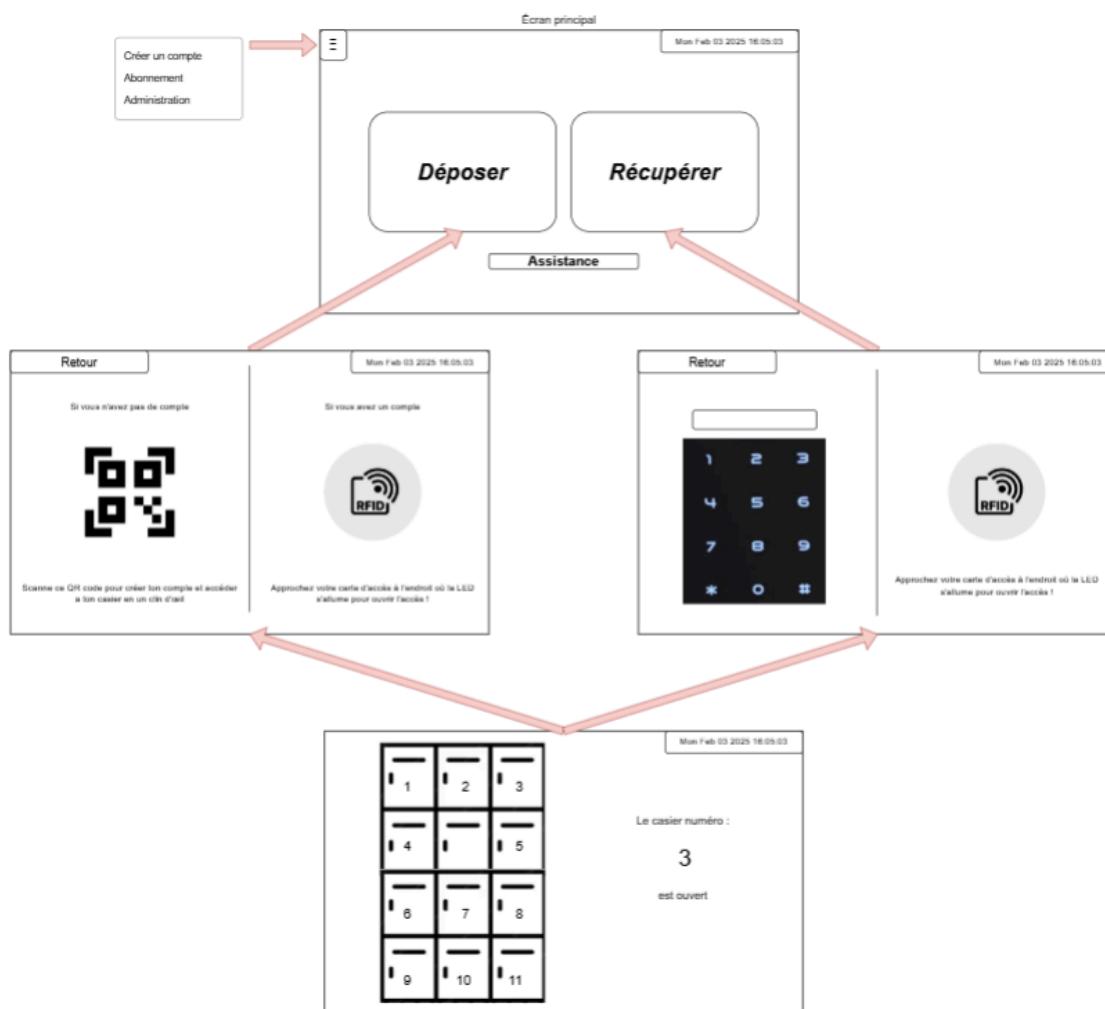
4.3 Contribution de Bastien

4.3.1 Développement de l'Interface Homme-Machine (IHM)

Dans le cadre de ce projet de casier connecté, ma principale responsabilité a été la conception et le développement de l'Interface Homme-Machine (IHM), un élément central du système permettant à l'utilisateur d'interagir avec le casier de manière intuitive et sécurisée.

Avant de commencer le développement, j'ai analysé les besoins fonctionnels de l'IHM. L'interface devait permettre à l'utilisateur de s'identifier (par exemple via un code PIN), de déverrouiller le casier si l'accès était autorisé, et éventuellement d'afficher des messages d'erreur ou de confirmation.

Dans un premier temps, j'ai réalisé un schéma sans programmation afin de visualiser à quoi ressemblerait mon interface utilisateur (IHM) :



Ce schéma m'a permis de débuter la phase de programmation avec une idée claire et précise de l'apparence souhaitée pour l'interface.

J'ai choisi d'utiliser les langages HTML, CSS et JavaScript pour la partie visuelle de l'interface (IHM), car ils permettaient une mise en œuvre simple et efficace, tout en offrant une bonne flexibilité pour les interactions.

Pour la partie backend, j'ai utilisé le langage Python avec le framework Flask, qui permet de gérer les requêtes côté serveur et la communication avec la base de données SQLite.

Conception de l'interface

J'ai conçu une interface claire et responsive, en tenant compte des contraintes matérielles (taille de l'écran, interaction tactile, affichage dans un environnement scolaire). L'IHM comportait notamment :

Une page d'accueil permettant à l'utilisateur de choisir entre déposer ses affaires ou les récupérer :



Un clavier numérique virtuel pour la saisie du code PIN lors de la récupération des affaires, avec possibilité d'utiliser un badge NFC en alternative :

Menu principal
mardi 10 juin 2025 - 19:09:33

Récuperer

Veuillez suivre les instructions ci-dessous pour récupérer vos affaires.

Entrez votre code :

1	2	3
4	5	6
7	8	9
✖	0	✓

Approchez votre carte d'accès à l'endroit où la LED s'allume pour ouvrir l'accès !

Un QR code redirigeant vers la page web de l'application mobile pour créer un compte, et un badge NFC permettant l'ouverture des casiers :

Menu principal
mardi 10 juin 2025 - 19:11:05

Déposer

Veuillez suivre les instructions ci-dessous pour déposer vos affaires en toute sécurité.

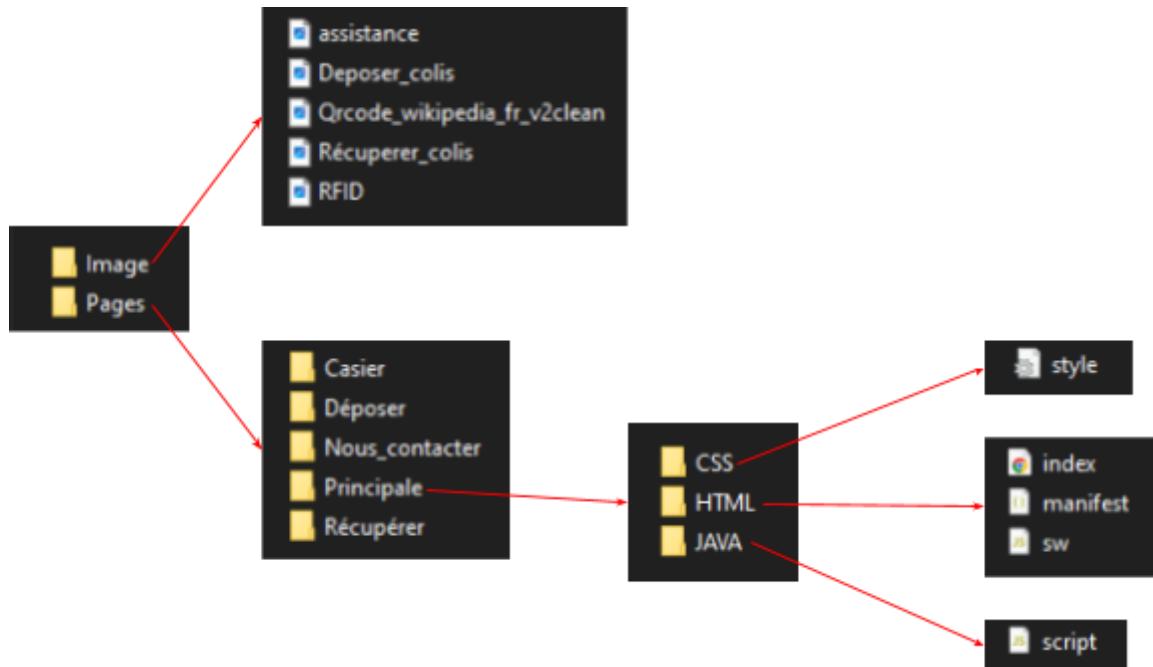
Si vous n'avez pas de compte

Scanne ce QR code pour créer ton compte et accéder à ton casier en un clin d'œil

Si vous avez un compte

Approchez votre carte d'accès à l'endroit où la LED s'allume pour ouvrir l'accès !

Pour la partie programmation de l'IHM mon dossier se présente comme ceci :



J'utilise un fichier HTML pour définir la structure de ma page, c'est-à-dire les différents éléments qui la composent (titres, boutons, champs de saisie, etc.).

Pour la mise en forme, j'utilise le CSS, qui me permet d'harmoniser le style sur l'ensemble des pages et d'offrir une expérience utilisateur plus agréable et cohérente.

Enfin, j'intègre du JavaScript pour rendre la page interactive, notamment pour gérer les actions de l'utilisateur (comme la saisie du code PIN, l'ouverture d'un casier, ou l'affichage de messages dynamiques).

Une fois l'interface développée, j'ai travaillé sur la liaison entre l'IHM et le système embarqué. Grâce à une architecture simple en requêtes HTTP ou en envoi de commandes via des scripts sur le Raspberry Pi, l'IHM permettait de transmettre les actions de l'utilisateur au microcontrôleur (ex. : demande d'ouverture du casier)."

```

function validateCode() {
    fetch("http://localhost:5000/verify-pin", {
        method: "POST",
        headers: { "Content-Type": "application/json" },
        body: JSON.stringify({ pin: codeSaisi })
    })
}
  
```

Comme illustré ici, une requête HTTP est envoyée pour vérifier si le code PIN saisi correspond à celui stocké dans la base de données (dont le fonctionnement est détaillé plus loin).

JavaScript est également utilisé pour des fonctionnalités plus simples, comme l'affichage de l'heure et la date sur l'interface :

```
// --- Affichage de la date et l'heure ---
function updateDateTime() {
    const now = new Date();
    const options = {
        weekday: 'long', year: 'numeric', month: 'long', day: 'numeric'
    };
    const dateStr = now.toLocaleDateString('fr-FR', options);
    const timeStr = now.toLocaleTimeString('fr-FR');
    document.getElementById("dateHeure").textContent = `${dateStr} - ${timeStr}`;
}
```

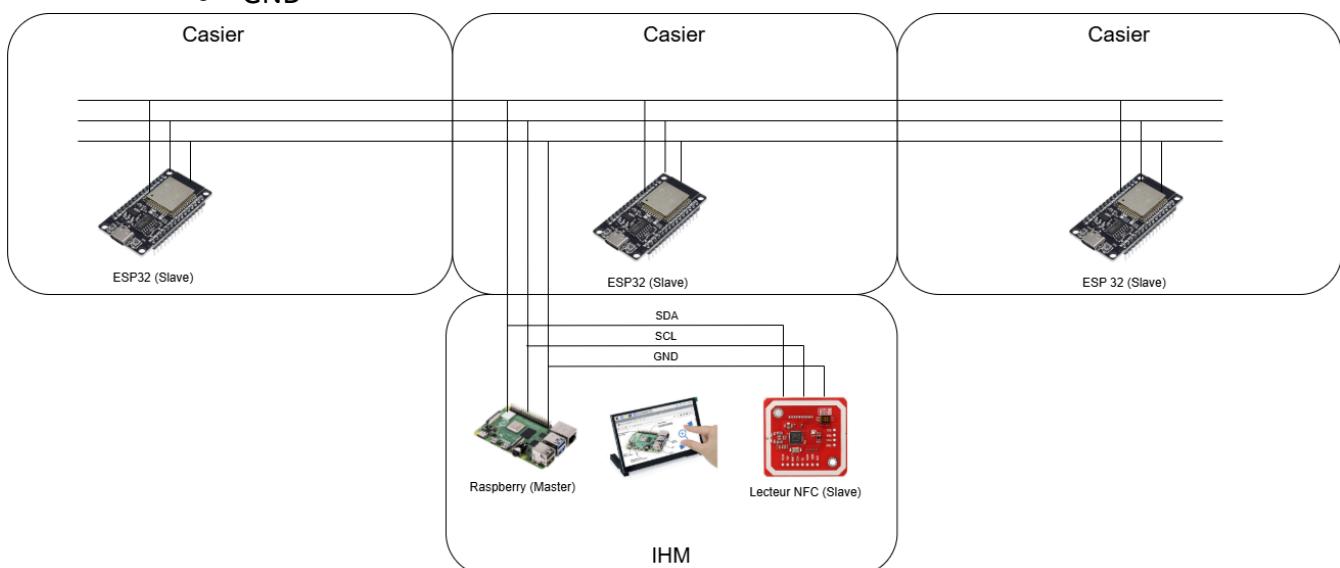
Le code est disponible en Annexes : [6.3 Codes sources](#)

4.3.2 Communication I2C

Dans le cadre de notre projet, la communication entre le Raspberry Pi, un ou plusieurs ESP32 (un par casier) et le lecteur de badge NFC a été réalisée via le bus I2C (Inter-Integrated Circuit). Cette liaison a permis au Raspberry Pi d'envoyer des commandes à l'ESP32, notamment pour déclencher certaines actions physiques telles que l'ouverture ou la fermeture d'un casier et de recevoir l'ID des badges qui été scanné devant le lecteur NFC.

Sur le Raspberry Pi, l'interface I2C a été activée via raspi-config. Le câblage a été réalisé à l'aide des broches GPIO correspondantes :

- SDA (GPIO 2)
- SCL (GPIO 3)
- GND



Un script Python a été développé pour gérer l'envoi de commandes I2C depuis le Raspberry. Ce script est intégré dans une API REST construite avec Flask. Lorsqu'un utilisateur entre un code PIN valide depuis l'IHM (interface HTML), une requête est envoyée à un serveur Flask (comme mentionné précédemment). Celui-ci, une fois l'utilisateur authentifié, envoie une commande via I2C à l'ESP32.

L'ESP32 a été programmé avec Visual Studio Code en C++, en utilisant la bibliothèque WireSlave, qui permet à l'ESP32 de fonctionner en tant qu'esclave I2C. Un identifiant d'adresse esclave (par exemple 0x08) a été attribué pour que le Raspberry puisse le cibler précisément. (Dans la partie "[5.2 Améliorations possibles](#)" nous parlerons de l'adressage automatique des ESP32 pour faciliter l'installations des casier)

Voici les étapes réalisées sur l'ESP32 :

- Initialisation du mode esclave I2C avec WireSlave.begin(SDA, SCL, address)
- Mise en place d'un buffer de réception pour les commandes envoyées
- Décodage des données reçues pour déclencher une action (ex. : ouvrir un casier via un servomoteur ou relais)

Interaction avec l'interface utilisateur

Lorsqu'un utilisateur entre un code PIN sur l'interface web hébergée par le Raspberry Pi, celui-ci interroge la base de données SQLite. Si le code est valide, la logique côté serveur appelle un script I2C pour envoyer l'ordre à l'ESP32. Cette chaîne de traitement assure une interaction sécurisée et centralisée entre le serveur, l'IHM et le microcontrôleur.

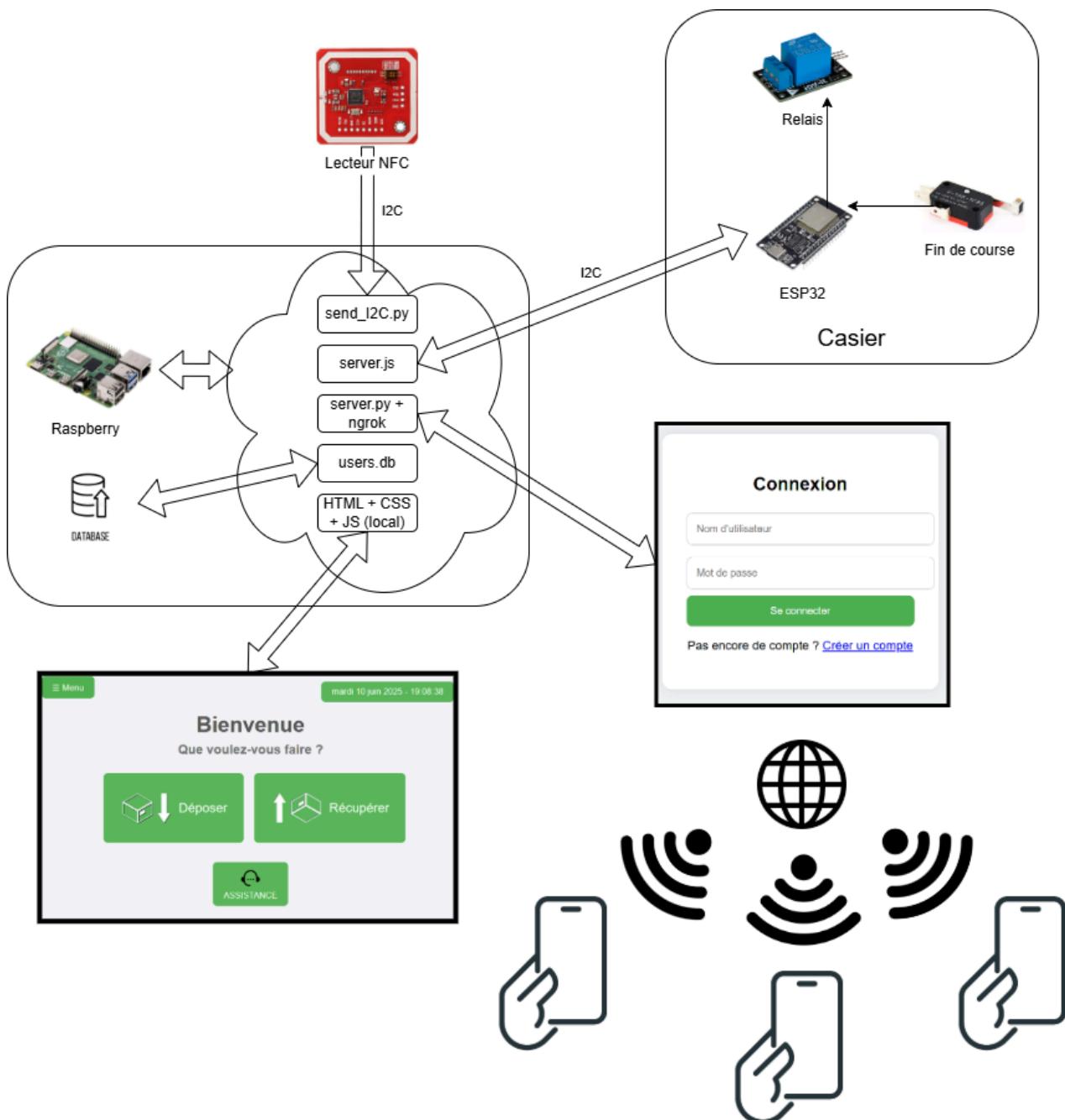
Sécurité et vérifications

Pour éviter les collisions ou les interférences sur le bus I2C, les connexions ont été vérifiées à l'oscilloscope. Le code a également été conçu pour vérifier la disponibilité du bus et gérer les erreurs de communication.

Le code de l'ESP32 est disponible en Annexe : [6.3 Codes sources](#)

4.3.3 Gestion informatique et serveur

Dans cette partie, et en raison de l'absence de certains membres de l'équipe, j'ai pris en charge l'intégration et la mise en communication des différents modules informatiques du projet. Pour mieux illustrer l'architecture logicielle mise en place, j'ai réalisé un synoptique représentant l'ensemble des composants du système :



Lecture des tags NFC — send_i2c.py

Ce script Python permet de lire les tags NFC scannés par un lecteur et d'envoyer les informations récupérées via le bus I²C au Raspberry Pi. Le Raspberry Pi agit comme maître I²C et reçoit les données. Ce processus permet de transmettre de manière sécurisée les identifiants des utilisateurs détectés par le lecteur NFC.

Contrôle des relais (casier) — server.js

Ce serveur Node.js permet de piloter l'ESP32, qui gère l'ouverture et la fermeture des casiers, détecte leur état (ouvert/fermé) et contrôle la prise électrique. Lorsqu'un code PIN est validé, le serveur envoie un signal au relais correspondant pour déverrouiller le casier. Le script Node.js reste à l'écoute de requêtes HTTP, permettant de recevoir des commandes depuis l'interface web ou d'autres modules du système.

Serveur web Flask — server.py

Ce script lance un serveur web Python avec Flask. Il gère :

- L'authentification des utilisateurs (login, enregistrement, vérification du code PIN)
- La gestion des sessions
- Les modifications de code PIN
- L'interface utilisateur HTML via le dossier static/
- Les échanges avec la base de données SQLite

Il constitue la colonne vertébrale du système, centralisant toutes les logiques applicatives du côté serveur.

Mise en ligne avec Ngrok

Ngrok est utilisé pour exposer le serveur Flask en ligne via un tunnel sécurisé. Cela permet d'accéder à l'interface web à distance depuis un smartphone ou un autre ordinateur sans configurer de redirection de ports.

Base de données — users.db

Le système utilise une base de données SQLite, simple et légère, parfaitement adaptée pour un usage embarqué sur un Raspberry Pi. Cette base est stockée localement sous le fichier users.db et sert à gérer l'ensemble des informations utilisateurs du système.

La base de données contient une table principale appelée users avec les colonnes suivantes :

- id : identifiant unique (clé primaire, auto-incrémenté)
- username : nom d'utilisateur (chaîne de caractères, unique)
- password : mot de passe associé (non chiffré dans ce projet, mais peut être amélioré avec du hash)
- pin : code PIN à 6 chiffres utilisé pour ouvrir le casier

Les interactions avec la base sont réalisées via le module sqlite3 en Python dans le fichier server.py, notamment pour :

- Vérifier les identifiants lors du login
- Enregistrer de nouveaux utilisateurs via la page register
- Associer ou modifier un code PIN
- Vérifier l'unicité d'un code PIN avant modification
- Identifier l'utilisateur en fonction du tag NFC reçu via I²C

Cette base de données joue un rôle essentiel dans la sécurité et la gestion centralisée des accès, permettant d'associer un code utilisateur à un casier spécifique contrôlé électroniquement.

4.3.4 Électronique

Pour la partie électronique, j'ai activement collaboré avec Antoine, en lui apportant mon aide sur plusieurs aspects techniques, en particulier le routage. Grâce à l'expérience que j'ai pu acquérir au cours de mon alternance en bureau d'études, j'ai pu lui faire bénéficier de mes connaissances dans la conception de cartes électroniques.

J'ai notamment pris en charge le placement stratégique des composants, en tenant compte à la fois des contraintes d'encombrement. J'ai également réalisé une partie importante du routage, en optimisant les chemins pour limiter les interférences avec le 230V, simplifier le suivi des pistes et faciliter le processus de fabrication.

L'objectif principal était de produire une carte compacte, fonctionnelle et fiable, en respectant les bonnes pratiques de conception électronique.

Cette collaboration m'a permis de mettre en pratique mes compétences, tout en consolidant ma capacité à travailler en équipe sur des projets concrets.

Pour davantage de détails concernant la conception de la carte électronique, je vous invite à consulter la partie rédigée par Antoine, qui revient plus en profondeur sur les choix techniques et les étapes de développement : [4.2 Contribution Antoine](#)

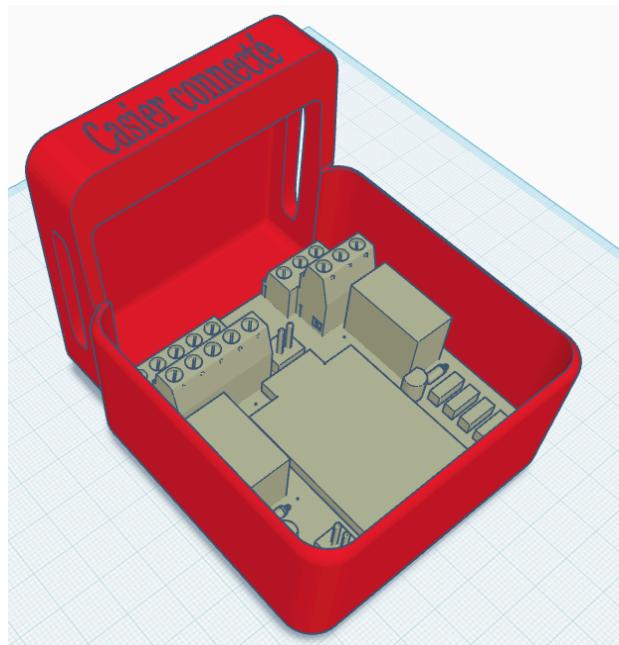
4.3.5 Mécanique

J'ai également apporté mon soutien à Antoine sur la partie mécanique du projet. Dans un premier temps, j'ai conçu un support destiné à accueillir l'écran ainsi que le Raspberry Pi, en utilisant le logiciel de modélisation 3D Tinkercad. Cette pièce a été pensée pour être à la fois compacte et facilement intégrable sur le casier central. J'ai pris en compte les contraintes d'encombrement ainsi que la facilité d'accès aux ports de connexion.

Un autre élément important intégré dans cette pièce est le lecteur NFC, que j'ai volontairement dissimulé derrière une fine couche de plastique afin qu'il reste fonctionnel tout en étant protégé et esthétiquement intégré à l'ensemble. Pour indiquer clairement sa présence et guider l'utilisateur, j'ai ajouté sur le modèle 3D un texte gravé ainsi qu'un logo NFC, directement visibles sur la face avant du support. Ce soin apporté à l'ergonomie et à l'aspect visuel contribue à une meilleure expérience utilisateur tout en respectant les contraintes techniques du projet.



J'ai également contribué à la conception d'un second modèle 3D, destiné à accueillir la carte électronique à l'arrière des prises intégrées aux casiers. Ce boîtier permet de dissimuler proprement la partie électrique du projet tout en assurant sa protection.



4.3.6 Chef de projet

Dans le cadre de la gestion du projet, j'ai utilisé l'outil ClickUp qui s'est révélé particulièrement efficace pour structurer notre travail en équipe. Il m'a permis de créer un découpage clair des tâches à accomplir, d'attribuer ces dernières aux différents membres de l'équipe, et de suivre l'avancement de chacun. Grâce à cette organisation, nous avons pu maintenir une certaine visibilité sur le projet malgré les difficultés rencontrées.

En effet, bien que la collaboration ait parfois été entravée par un manque de rigueur et d'implication de certains membres, j'ai veillé à ce que les échéances soient respectées et à ce que les livrables soient finalisés à temps. Cette capacité d'adaptation face aux imprévus et aux aléas humains a été un véritable exercice de gestion de projet en conditions réelles.

Nous avons ainsi pu aboutir à un prototype fonctionnel du casier connecté, intégrant l'ensemble des modules matériels et logiciels prévus, ce qui représente une réelle réussite au regard du contexte.

En parallèle, j'ai pris l'initiative de rédiger la quasi-totalité du rapport de projet, veillant à ce qu'il reflète fidèlement le travail réalisé, aussi bien sur le plan technique qu'organisationnel. Seules les sections spécifiques aux contributions individuelles des autres membres de l'équipe ont été laissées à leur charge. Cette rédaction m'a permis de faire le lien entre les différentes briques du projet et de renforcer ma capacité à synthétiser et structurer l'information.

4.4 Collaboration et coordination dans l'équipe

Malgré plusieurs absences importantes au sein de l'équipe au cours du projet, nous avons su maintenir une collaboration efficace et une bonne coordination grâce à une organisation rigoureuse.

L'outil ClickUp a joué un rôle central dans cette gestion. Il nous a permis de suivre l'état d'avancement du projet en temps réel, de répartir les tâches clairement et de communiquer facilement, même en l'absence de certains membres. Chaque membre pouvait consulter les tâches en attente, voir les priorités et ajouter des commentaires ou des remarques, ce qui a facilité le travail en différé.

Nous avons également veillé à tenir des réunions régulières dès que l'équipe était au complet afin de faire le point, redéfinir les priorités et redistribuer certaines tâches si nécessaire. Cela a renforcé l'esprit d'équipe et permis de garder une ligne directrice claire tout au long du développement.

Grâce à cette organisation et à la capacité d'adaptation de chacun, nous avons pu mener à bien un projet concret et fonctionnel, malgré les contraintes rencontrées en cours de route.

5 Conclusion et perspectives

5.1 Bilan du projet

Le projet de casier connecté a permis de concevoir un système combinant des compétences en électronique, informatique embarquée, développement logiciel et impression 3D. Il répond à un besoin concret : permettre à un utilisateur d'accéder à un casier sécurisé via un code PIN ou un badge NFC, tout en centralisant la gestion sur un Raspberry Pi.

L'architecture retenue repose sur une structure modulaire, facilitant l'adaptation à différents types de casiers et de câblages. L'intégration de différents modules (interface HTML, API Flask, communication I2C, contrôle via ESP32) a permis d'assurer une bonne interopérabilité entre les composants matériels et logiciels.

Du côté électronique, la réalisation de cartes sur mesure, l'attention portée à la CEM, au routage et au refroidissement, ainsi que l'utilisation de composants adaptés ont contribué à la robustesse du système.

En mécanique, les pièces ont été modélisées et imprimées en 3D pour s'intégrer parfaitement dans l'environnement du casier. Le support de l'écran, le logement de l'ESP32, ainsi que la discrétion du lecteur NFC témoignent d'une réflexion soignée sur l'ergonomie et l'esthétique.

Malgré certaines difficultés rencontrées, notamment en termes d'organisation et de respect des délais, les objectifs fonctionnels principaux ont été atteints. Le casier connecté est opérationnel et constitue une base fiable pour une future industrialisation ou un déploiement à plus grande échelle.

5.2 Améliorations possibles

5.2.1 Sécurité informatique

En raison du temps limité, certains aspects du projet ont été partiellement négligés, notamment la sécurité du système.

Par exemple, les informations sensibles telles que les identifiants et mots de passe des utilisateurs sont actuellement stockées en clair dans une base de données locale sur le Raspberry Pi, sans aucune mesure de protection. Ce choix, bien qu'acceptable dans un cadre expérimental ou de démonstration, représente une faille importante dans un environnement réel.

Il s'agirait donc d'un point d'amélioration prioritaire pour une version finale ou une éventuelle industrialisation, en intégrant des pratiques de sécurisation comme le hachage des mots de passe, l'authentification sécurisée et le chiffrement des communications.

5.2.2 Sécurité électrique

Dans notre projet, l'ensemble des prises électriques des casiers est actuellement alimenté à partir d'une seule prise d'alimentation. Cependant, selon la norme NF C 15-100, il n'est autorisé de raccorder qu'un maximum de 8 prises sur un disjoncteur de 16 A.

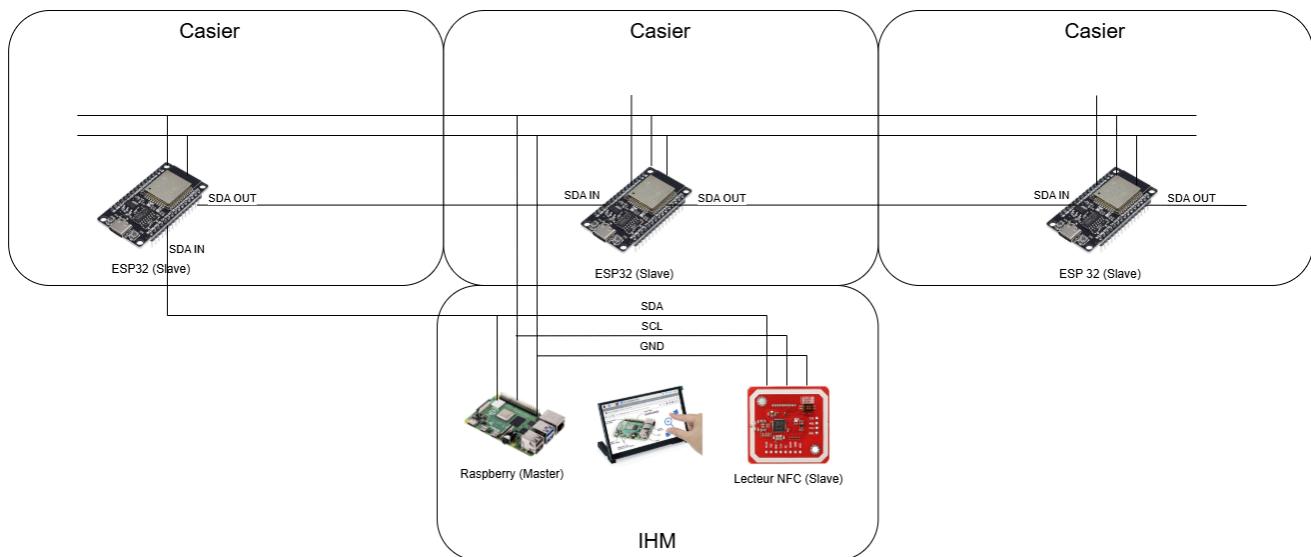
Ainsi, pour garantir la conformité réglementaire et assurer une sécurité électrique optimale, il serait nécessaire de repenser l'alimentation électrique en cas d'extension du système à plus de 8 casiers.

Une solution consisterait à raccorder directement le système au tableau électrique principal et à prévoir des disjoncteurs séparés à l'intérieur du casier, afin de répartir correctement la charge et respecter les prescriptions de la norme.

5.2.3 Adressage Bus I2C

Concernant le bus I2C, chaque ESP32 connecté au système doit posséder une adresse unique afin d'être reconnu individuellement et de pouvoir échanger des données avec le Raspberry Pi. Actuellement, l'ajout d'un nouvel ESP32 nécessite de définir manuellement son adresse, aussi bien sur l'ESP32 lui-même que dans le code du Raspberry Pi.

Cette méthode, bien que fonctionnelle, limite la scalabilité du système. Une amélioration possible consisterait à chaîner les ESP32 en série, de manière à ce que chaque module attribue automatiquement une adresse au suivant lors de son initialisation. Cela permettrait d'automatiser l'adressage des modules et de simplifier leur intégration dans le système.



5.2.4 Mécanique

Au niveau mécanique, plusieurs améliorations pourraient être envisagées pour optimiser l'intégration des composants dans les casiers :

- Le support de l'écran pourrait être modifié afin qu'il soit entièrement maintenu par la structure du casier. Cela permettrait de masquer l'arrière de l'écran et d'obtenir une finition plus propre et professionnelle.
- Le support de la prise électrique pourrait également être revu pour renforcer la fixation entre la carte électronique, le boîtier et la prise. Un meilleur système d'ancrage offrirait plus de robustesse et faciliterait l'installation ainsi que la maintenance.

5.3 Conclusion

5.3.1 Conclusion de Soultane

Dans un premier temps, nous avons développé une version fonctionnelle en PHP, avec une interface HTML/CSS et une base de données MySQL administrée via phpMyAdmin. Cette phase nous a permis de comprendre les fondements de l'authentification, la gestion des sessions et la sécurisation des données utilisateur (notamment via le hachage des mots de passe).

Par la suite, afin de gagner en souplesse et en évolutivité, nous avons migré vers une architecture basée sur Python et le framework Flask. Cette nouvelle version intègre la gestion des utilisateurs (inscription, connexion, modification de code PIN), la prise en charge des sessions via cookies, ainsi que la compatibilité CORS pour permettre les échanges entre différents domaines. Les données sont désormais stockées de manière légère et efficace à l'aide d'une base SQLite.

Cette progression nous a permis de comparer deux approches différentes en termes de développement web, d'identifier les avantages et les limites de chacune, et de mettre en pratique des notions essentielles comme la sécurité, la modularité, et la gestion d'état utilisateur dans une application web moderne.

Ce projet nous a permis de mettre en œuvre un système complet d'authentification utilisateur, en passant par différentes phases de conception, de test et d'amélioration technologique. Le gain ou la plus value que ce projet m'a apporté personnellement réside beaucoup plus dans la partie électronique, la communication des objets ou le téléversement du code sur les modules sont des choses que je connaissais.

5.3.2 Conclusion de Antoine

Ce projet a été une excellente opportunité de mettre en pratique et d'approfondir les connaissances en conception électronique et en gestion de projet que nous avons pu apprendre tout au long de l'année. J'ai appris à naviguer entre les contraintes techniques, les exigences fonctionnelles et les choix de composants, tout en veillant à la cohérence globale du système. Les défis rencontrés, qu'il s'agisse d'optimiser le placement des éléments sur le circuit ou d'assurer la bonne interconnexion des différents modules, ont été des moments clés pour développer ma capacité à résoudre des problèmes de manière autonome et rigoureuse.

En somme, ce travail m'a permis d'acquérir une vision complète du cycle de vie d'un produit électronique. Il a renforcé ma compréhension de l'importance d'une méthodologie structurée et de la précision à chaque étape du processus de développement.

5.3.3 Conclusion de Bastien

Cette année d'alternance et la réalisation de ce projet m'ont permis de consolider de nombreuses compétences, tant sur le plan technique que personnel. En particulier, j'ai pu approfondir mes connaissances en électronique, notamment grâce à la conception et au routage de cartes électroniques, domaine dans lequel mon expérience en entreprise m'a été précieuse. Cette expertise m'a permis de soutenir efficacement mon équipe dans les choix techniques et les phases critiques du développement matériel.

D'un point de vue logiciel, j'ai également pris une part active dans l'intégration des différentes briques logicielles du projet : IHM en HTML/CSS/JavaScript, communication via API REST en Python avec Flask, gestion I2C, scripts Node.js pour le pilotage des ESP32, etc. Cette polyvalence m'a permis d'assurer la cohérence du système et de faire le lien entre les composants électroniques et les interfaces utilisateurs.

En parallèle, j'ai apporté une contribution significative à la conception mécanique à travers la modélisation de pièces 3D sur Tinkercad, telles que le support pour l'écran tactile, l'intégration discrète du lecteur NFC ou encore les boîtiers pour les cartes électroniques dans les casiers.

Ce projet a également été une véritable expérience de gestion de projet. En utilisant l'outil ClickUp, j'ai pu organiser les tâches, répartir les rôles et compenser les manques d'implication de certains membres. Cela m'a demandé de faire preuve de rigueur, d'autonomie et de prise d'initiative, notamment en assurant la rédaction principale du rapport de projet.

Enfin, je peux dire que les enseignements reçus en formation m'ont été très utiles, particulièrement dans les domaines de l'électronique et de la gestion de projet. Cette mise en pratique concrète m'a permis de mieux comprendre les enjeux réels du métier d'ingénieur et de confirmer mon orientation professionnelle.

6 Annexes

6.1 STB (Spécification Technique de Besoin)

La Salle 9 Rue Notre Dame des 7 douleurs Avignon 04 90 14 56 56 antoinebrousse.bitsn@gmail.com michel.bastien.pro@gmail.com	Licence Pro Systèmes Numériques	Session 2025
Locker Connecté		
		
Partenaire professionnel : BRISSY Sylvain	Étudiants chargés du projet : MICHEL Bastien BROUSSE Antoine	Professeurs ou Tuteurs responsables : BEAUMONT Jérôme (EC), LUQUET Christophe (EC)
Reprise d'un projet existant : NON / OUI		

LP SN UCE 2

Projet technique : Locker connecté

Aix-Marseille

Présentation générale du système

Avec le COVID, l'explosion de l'utilisation des vélos a chamboulé les pratiques dans les grandes villes. Les routes et les axes sont redessinés, la place est désormais laissée aux deux roues. Toutefois une problématique subsiste et inquiète encore bon nombre de nos vélocistes.

Où garer les vélos et comment les sécuriser ?

Les mairies se tournent légitimement vers les spécialistes du parkings et leur demandent de proposer des solutions concrètes et rapides. Les parkings mettent alors en place des stationnements sécurisés (local fermé accessible avec un badge). Pour obtenir celui-ci, l'utilisateur doit acheter un abonnement stationnement de vélo et attendre de recevoir son badge chez lui.

L'espace sécurisé contient des racks pour attacher les vélos, mais aussi une station de gonflage, un atelier de réparation et des casiers. Ces casiers sont actuellement difficilement attribuables et ne peuvent être facilement louables. C'est par une gestion de cadenas compliquée que le parking donne un accès au client. L'objet de notre solution est de proposer une alternative connecté à cette gestion de casier.

Expression du besoin

L'usager a accès au local fermé via un accès badge RFID. Ce même badge devra être utilisé pour ouvrir le casier. En option, une connexion NFC pourra aussi être envisagée. Une fois le casier ouvert, l'usager peut poser ses affaires et le refermer. Un casier peut aussi contenir une prise électrique qui est alimentée sur demande par l'utilisateur depuis un site internet ou une appli mobile.

Le casier se verrouille automatiquement lorsqu'il est refermé. Si jamais un usager oublie son badge à l'intérieur du casier, il devra alors taper un code à 6 chiffres (lié à son compte) pour le déverrouiller. Il pourra aussi l'ouvrir depuis internet ou son application mobile et une connexion 4G sera nécessaire pour la communication smartphone (disponible dans certains parkings).

LP SN UCE 2

Projet technique : Locker connecté

Aix-Marseille

Le Back office proposé devra à tout moment pouvoir connaître l'état (ouvert/fermé/en charge/non en charge, capteur température) du casier.

Il sera aussi possible de consulter les informations du client (code/N° de casier).

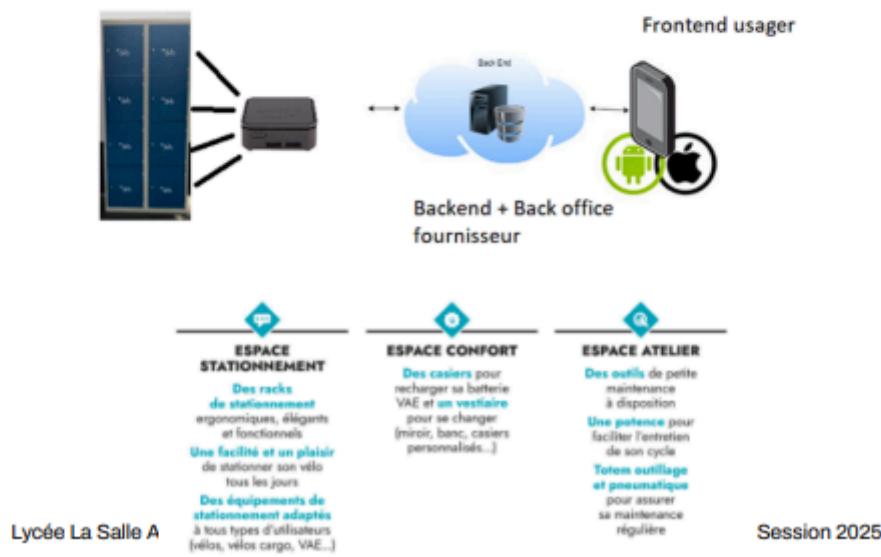
Un capteur permet de mesurer la température et de remonter une alerte si la chaleur est trop élevée et que la prise est alimentée.

Missions du système

Les missions du système casier connecté sont de proposer une gestion simplifiée des casiers connectés pour les parkings à vélos, avec un accès sécurisé via RFID ou application mobile, et un suivi en temps réel de l'état des casiers.

- Étude et spécifications fonctionnelles
- Conception du système
- Développement de l'application mobile et/ou du site internet
- Développement et intégration des casiers connectés
- Mise en place de la communication réseau
- Support et maintenance

Description structurelle du système



LP SN UCE 2

Projet technique : Locker connecté

Aix-Marseille

Contraintes imposées

Un local sécurisé à vélo pourra contenir jusqu'à 16 casiers tous gérés par un module centralisant la connexion internet et les liaisons aux casiers.

La plupart des parkings n'ont pas de réseau cellulaire stable, la solution devra alors passer en ethernet.

L'usager a accès au local fermé via un accès badge RFID. Ce même badge devra être utilisé pour ouvrir le casier.

Module IHM/Application

L'IHM (Interface Homme-Machine) ou l'application mobile du projet Connected Locker permet à l'utilisateur de gérer ses casiers de manière simple et intuitive. Grâce à l'application, l'utilisateur peut réserver un casier, l'ouvrir via son smartphone (en utilisant la technologie RFID ou NFC), et suivre l'état du casier (ouvert/fermé, en charge, température). En cas d'oubli du badge, un code à six chiffres peut être utilisé pour déverrouiller le casier. L'application permet également de contrôler l'alimentation de la prise électrique des casiers, en allumant ou éteignant la prise à distance. L'interface sera conçue pour être fluide et facile à utiliser, avec des notifications en cas d'anomalie ou d'alertes (par exemple, température trop élevée dans le casier).

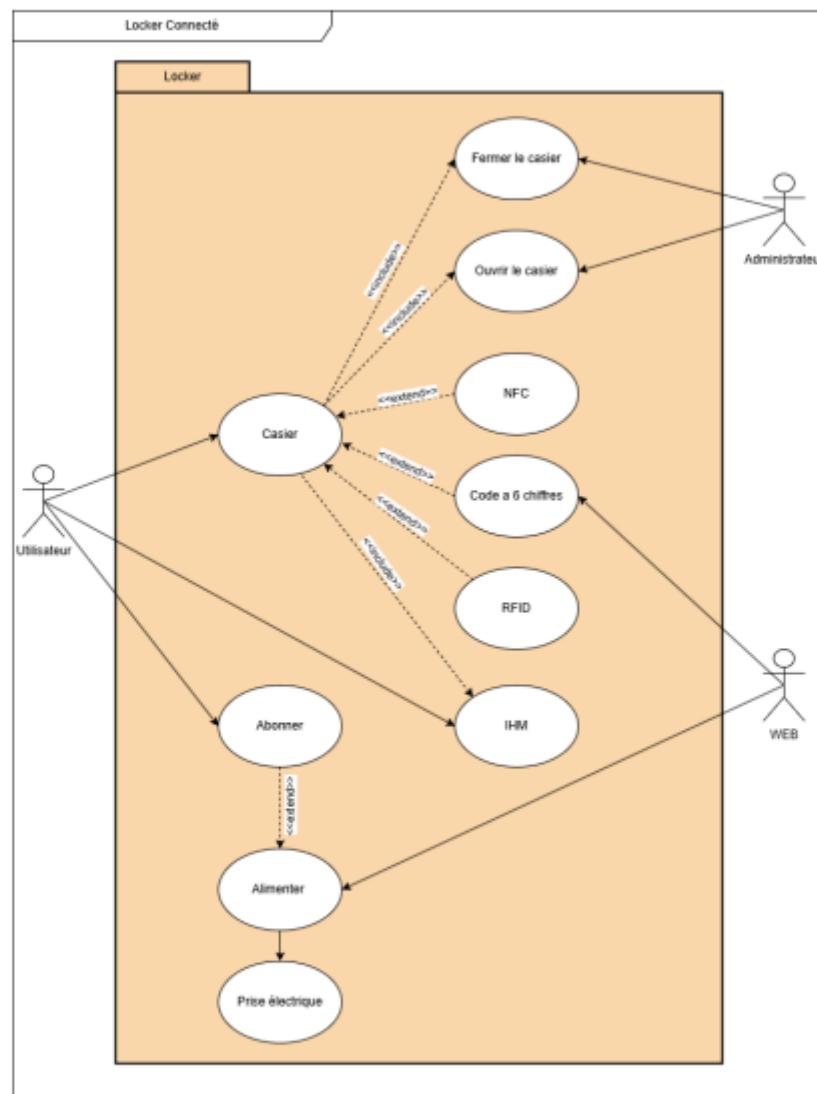
Module Casier

Le casier connecté est un espace sécurisé destiné au stockage des effets personnels des utilisateurs dans les parkings à vélos. Il est équipé d'un système de verrouillage intelligent, permettant un accès via un badge RFID ou, en option, par technologie NFC. Le casier se verrouille automatiquement une fois refermé, et si l'utilisateur oublie son badge à l'intérieur, il peut le déverrouiller à distance via une application mobile ou un code à six chiffres. Certains casiers sont aussi dotés de prises électriques, que l'utilisateur peut activer à distance pour recharger ses appareils. Enfin, des capteurs de température intégrés permettent de surveiller la sécurité du casier et d'envoyer des alertes en cas de surchauffe.

LP SN UCE 2

Projet technique : Locker connecté

Aix-Marseille



LP SN UCE 2

Projet technique : Locker connecté

Aix-Marseille

Inventaire des matériels et outils logiciels

Les ressources matérielles

Désignation	Caractéristiques techniques	Acquisition	Existant
Gâche mécanique	à définir	x	
Écran	à définir	x	
Casier	Casier en métal, bois		x
Prise électrique	à définir	x	

Les ressources logicielles

Désignation	Caractéristiques
Simulation électronique	PROTEUS 8.3 (module ISIS)
Ihm	Base de données, interface utilisateur
Application	Sous Android/IOS

Planification prévisionnelle

Date de début du projet	Semaine 4
Rendu des spécifications fonctionnelles/non fonctionnelles	Semaine 6
Revue à mi-parcours de projet	Semaine 13
Revues finales du projet	Semaine 26

6.2 Etude de marché

CONNECTED LOCKER

Qu'est ce qu'un casier connecté ?

Le casier connecté est une armoire de stockage dont la gestion est assurée par un logiciel informatique, généralement connecté au réseau internet. Sa gestion peut se faire à distance, par le biais d'un smartphone ou d'un ordinateur.



Le casier connecté possède de nombreux emplacements de stockage dont l'accès est sécurisé au moyen d'un code à usage unique. En effet, il est essentiel de protéger les colis et les marchandises qui y sont rangés contre le vandalisme et les ouvertures erronées.

La prise en main d'un casier connecté est intuitive. L'utilisateur se sert d'un écran tactile pour naviguer parmi les différentes fonctionnalités du casier. Il y insère son code secret pour déverrouiller la case qui lui a été allouée.

Une fois que l'utilisateur récupère son colis et referme le casier, la case de stockage se referme automatiquement. Ce compartiment est alors disponible pour un nouvel utilisateur.

1. Contexte et Présentation du Marché

Les casiers connectés sont des dispositifs de stockage électronique, accessibles à distance via des applications mobiles ou des interfaces en ligne. Ils sont utilisés dans différents secteurs : entreprises, campus universitaires, gymnases, espaces publics, commerces, etc. Ces systèmes permettent aux utilisateurs de stocker des objets de manière sécurisée, de manière autonome, et sont souvent intégrés avec des solutions de paiement ou de gestion d'accès.

2. Segmentations du Marché

a. Par type de casiers :

Casiers de transport (ex. colis, marchandises) : Ces casiers sont principalement utilisés pour la livraison de colis (ex. Amazon Locker, Casier Pickup).

Casiers personnels : Pour les objets personnels, souvent trouvés dans les entreprises, écoles, salles de sport.

Casiers de stockage pour entreprises : Pour la gestion des ressources et équipements dans des espaces partagés.

b. Par secteur d'application :

Logistique & E-commerce : Utilisation croissante des casiers pour les livraisons automatiques de colis.

Espaces publics & Commerces de détail : L'intégration de casiers connectés pour faciliter les services d'achat en ligne ou la collecte de commandes.

Entreprises & Bureaux : Casiers intelligents utilisés dans des environnements de travail pour le stockage sécurisé d'objets personnels.

Éducation & Campus universitaires : Pour stocker des affaires personnelles, de manière plus sécurisée et pratique.

3. Technologies Clés

IoT (Internet of Things) : L'Internet des objets permet aux casiers d'être gérés à distance via des applications mobiles ou des plateformes web. Ces technologies assurent un suivi des accès et la gestion en temps réel.

Blockchain et Sécurité des données : Certaines entreprises explorent l'usage de la blockchain pour sécuriser l'accès et le suivi des casiers.

Intelligence Artificielle et Analyse des données : Utilisation de l'IA pour prévoir les comportements utilisateurs et optimiser les ressources des casiers.

Paiement sans contact / NFC : Intégration de paiements sans contact pour les casiers accessibles par une carte ou téléphone mobile.

4. Analyse de la Demande

a. Évolution de la Demande :

Le marché des casiers connectés connaît une forte croissance, notamment avec l'essor du e-commerce et des services de livraison. Les consommateurs recherchent des solutions de plus en plus pratiques et sécurisées pour récupérer leurs colis. De plus, l'utilisation des casiers intelligents dans des environnements comme les écoles, les entreprises et les espaces publics est en expansion.

b. Facteurs de Croissance :

Commodité : Les casiers connectés offrent un accès facile et rapide à des produits ou objets, ce qui attire un large public.

Sécurité : Ils permettent de garantir une sécurité renforcée pour les objets stockés, avec des accès personnalisés et traçables.

Logistique Last-Mile : Les entreprises de livraison cherchent à réduire les coûts et améliorer l'efficacité avec des solutions de stockage autonomes pour leurs clients.

c. Facteurs Limitants :

Coût d'installation initial : Le coût des technologies intégrées peut freiner l'adoption, notamment pour les petites entreprises ou collectivités locales.

Problèmes d'accèsibilité : Certains utilisateurs peuvent être réticents à adopter ces technologies, surtout s'ils ne sont pas familiers avec les applications mobiles.

5. Concurrence et Acteurs Principaux

Amazon Locker : Le géant du e-commerce est l'un des acteurs majeurs dans le secteur des casiers connectés pour la livraison de colis.

InPost : Fournisseur de casiers de retrait pour les colis en Pologne, très présent en Europe.

Willow : Start-up spécialisée dans les casiers connectés pour les entreprises et espaces publics.

Casiers à vélo & smart lockers : D'autres acteurs se concentrent sur des solutions plus spécifiques, comme les casiers pour les vélos ou dans les parkings d'entreprise.

6. Tendances et Perspectives du Marché

Prolifération des casiers dans les villes intelligentes : Le concept de ville intelligente s'appuie sur l'optimisation de la gestion des espaces publics, et les casiers connectés sont une partie de cette vision.

Partenariats avec des commerces locaux : Pour les petites entreprises, l'implantation de casiers connectés pour récupérer des achats en ligne ou pour le stockage temporaire pourrait se développer, ce qui serait une alternative pratique aux services de livraison à domicile.

Systèmes d'échange entre utilisateurs : Des solutions de casiers interconnectés permettant à des utilisateurs de partager des casiers ou des espaces de stockage, notamment dans les entreprises ou les écoles.

7. Opportunités

Développement dans les zones urbaines à forte densité : Les casiers connectés peuvent aider à désengorger les points de collecte traditionnels et offrir une nouvelle alternative dans les grandes villes.

Automatisation du secteur logistique : La mise en place de casiers pour les livraisons peut aider à réduire le coût des opérations de livraison et la congestion urbaine.

Conclusion

Le marché des casiers connectés connaît un développement rapide et offre de nombreuses opportunités, notamment dans le secteur de la logistique, des espaces publics, et des entreprises. Cependant, la concurrence y est forte et les barrières à l'entrée pour de petites structures sont liées à des coûts d'installation et à l'adoption technologique. Les entreprises souhaitant se lancer dans ce domaine devront se concentrer sur des solutions innovantes, sécurisées et faciles d'accès pour capter une clientèle de plus en plus exigeante en termes de flexibilité et de commodité.

6.3 Codes sources

L'ensemble du code source du projet (incluant les scripts pour le serveur Flask, les pages HTML/CSS/JS de l'interface utilisateur, ainsi que les programmes embarqués pour l'ESP32) est disponible sur un dépôt GitHub. Ce dépôt centralise tous les fichiers nécessaires à la compréhension, la reproduction et l'amélioration du système. Il est structuré de manière à séparer clairement les différentes parties du projet (interface, serveur, code ESP32, base de données).

Lien GitHub : <https://github.com/SNDOC-Michel-Bastien/Casier-connect-.git>