Projet IoT : Système de présence automatique par carte RFID

Matériel nécessaire pour la première version (prototype de base)

Membres du groupe

- **Bouba Ahmed - Lkhalidi Mohammed**

Objectif du projet

Ce projet vise à **automatiser la gestion des présences** dans un environnement académique grâce à une **double technologie de vérification** : la **RFID** et la reconnaissance par **empreinte digitale**. Chaque étudiant peut utiliser sa carte RFID personnelle ou son empreinte pour s'authentifier et enregistrer sa présence automatiquement, avec transmission des données via Wi-Fi vers une base de données centralisée.

Matériel nécessaire

Composant	Description	Qté	Prix (DH)
ESP32	Microcontrôleur avec Wi-Fi intégré	1	≈ 160
Lecteur RFID RC522	Module de lecture RFID 13.56MHz	1	≈ 35
Cartes RFID	Cartes/porte-clés 13.56MHz	2	≈ 35
Capteur d'empreinte	Module de lecture d'empreintes	1	≈ 180
Écran LCD (16x2 - I2C)	Affiche les informations de l'étudiant	1	≈ 45
Buzzer	Émet un son lors de la détection	1	≈ 10
Câbles Dupont (M-F)	Fils de connexion multicolores	20	≈ 20
Breadboard	Plaque d'essai 400 points	1	≈ 20
Alimentation 5V	Câble USB ou adaptateur	1	≈ 70
LEDs	Indicateurs visuels	2	≈ 10
Résistances(220 ohm)	Résistances	2	≈ 10
Total estimé du matériel			**≈ 585 dh**

Table 1: Liste du matériel nécessaire pour le système de présence IoT

Objectifs spécifiques :

1. **Concevoir et réaliser le module de capture (Hardware)** : Développer un système embarqué basé sur l'**ESP32** intégrant un **lecteur RFID (RC522)** et un **capteur d'empreinte digitale** pour une identification rapide et fiable des étudiants.
2. **Assurer la notification locale** : Intégrer un **écran LCD** au module ESP32 pour afficher immédiatement le statut de pointage (ex: "Présence enregistrée : [Nom]") et corriger les erreurs d'identification.
3. **Mettre en place la communication IoT** : Configurer la connexion **Wi-Fi** de l'ESP32 pour envoyer les données de présence structurées (**JSON** contenant UID/Empreinte, Heure, Salle, Module) vers le Cloud.
4. **Structurer le Backend Cloud** : Utiliser **Firebase Realtime Database** pour le stockage, la gestion et la synchronisation en temps réel des données de présence et des métadonnées des utilisateurs (noms, classes, emplois du temps).
5. **Développer l'application de supervision (Dashboard)** : Concevoir une interface Web administrative (**ReactJS**) capable de se connecter à Firebase pour afficher les présences **en temps réel**, générer l'historique, les statistiques et exporter les rapports de fréquentation pour les professeurs.

Principe de fonctionnement

1. L'étudiant présente sa **carte RFID** (au RC522) ou utilise le **capteur d'empreinte digitale**.
2. Le module de lecture (RC522 ou capteur biométrique) transmet l'identifiant unique (UID/Empreinte ID) à l'**ESP32**.
3. L'**ESP32** traite l'information, vérifie la validité du pointage, et établit la connexion Wi-Fi.
4. Le module affiche immédiatement le statut de confirmation ou d'erreur sur l'**écran LCD**.
5. Les données de présence (UID/ID, Heure, Salle, Module) sont transmises via JSON à la base de données **Firebase Realtime Database**.
6. **Firebase** enregistre la présence et synchronise l'information en temps réel.
7. Les professeurs peuvent consulter et gérer les données de présence actualisées via le **Dashboard ReactJS** (interface web/mobile).

Résultats attendus / Livrable final :

1. Système fonctionnel détectant automatiquement les élèves.
2. Tableau de présence consultable en ligne.

Compétences visées :

1. Programmation microcontrôleur et communication série.
2. Utilisation de capteurs et modules **RFID**.
3. Intégration **IoT** et gestion de données.
4. Documentation technique.

Étapes de réalisation :

1. Connecter le module **RFID** à **l'ESP32** et tester la lecture des cartes.
2. Programmer la carte pour détecter les cartes et récupérer l'**ID** unique.
3. Configurer la connexion **Wi-Fi** et le transfert des données.
4. Tester l'affichage des **présences en temps réel**.
5. Documenter le montage et rédiger le **mini-rapport technique**.