

Rapport de Projet

Industrie 4.0 et Usine Future

Système de contrôle d'accès automatisé via ESP32 et RFID

Encadré par : M. Hosni

Filière : Génie Industriel, Intelligence Artificielle et Data Science

Année universitaire : 2024/2025

Réalisé par :

DRIEF Nisrine

BOUTARHAT Alae

BELLMIR Yahya

CHEGDATI Chouaib

AOUSSAR Wissal

HJIRT Soufia

EL RHAZI Kaoutar

ISSOUMMOUR Amal

HATTAKI Kawter

Table des matières

0.1	Introduction	3
0.1.1	Contexte	3
0.1.2	Objectifs du projet	3
0.1.3	Contraintes rencontrées	4
0.2	Architecture technique	5
0.2.1	Composants Matériels	5
0.2.2	Composants Logiciels	5
0.2.3	Fonctionnement Global (Workflow)	6
0.3	Schéma du montage utilisé	7
0.3.1	Inventaire des composants	7
0.3.2	Explication du fonctionnement	7
0.4	Logiciels et outils de développement	8
0.4.1	Environnement local avec XAMPP	8
0.4.2	Les fichiers PHP	8
0.4.3	Base de données MySQL	9
0.4.4	Résumé du fonctionnement logiciel local	9
0.5	Pipeline de Développement	9
0.5.1	Développement Côté Serveur	9
0.5.2	Programmation de l'ESP32	9
0.5.3	Intégration et Tests	10
0.5.4	Configuration Matérielle et Câblage	10
0.6	Résultats et conclusions	10
0.6.1	Résultats obtenus	10
0.6.2	Conclusions	10

0.1 Introduction

0.1.1 Contexte

Dans le contexte de l'Industrie 4.0 et de l'usine du futur, la sécurisation intelligente des accès représente un enjeu stratégique majeur. Le présent projet s'inscrit pleinement dans cette dynamique en proposant une solution moderne et intégrée de contrôle d'accès automatisé. Il vise à concevoir et implémenter un système de gestion d'ouverture de porte basé sur la technologie RFID, en exploitant les capacités d'un microcontrôleur ESP32 dans un environnement IoT sécurisé et connecté.

Le système repose sur la lecture d'identifiants RFID via l'ESP32, lesquels sont ensuite vérifiés par un serveur distant hébergeant une base de données MySQL. En cas d'autorisation, une commande est envoyée pour activer un mécanisme d'ouverture de porte, garantissant ainsi un accès contrôlé et traçable. Une interface web développée en PHP permet l'administration des badges, la consultation des journaux d'accès, et l'interaction distante avec le système.

Ce projet constitue une solution concrète et évolutive pour la gestion des accès physiques dans les infrastructures industrielles ou académiques. Il illustre l'interopérabilité entre le matériel embarqué, les technologies web et les bases de données, et s'intègre dans une perspective de digitalisation des processus. La présente documentation décrit de manière détaillée les objectifs du projet, les choix technologiques, l'architecture système, ainsi que l'implémentation logicielle et matérielle retenue.

0.1.2 Objectifs du projet

L'objectif principal de ce projet est de concevoir et de développer un système intelligent de contrôle d'accès à une salle, reposant sur les technologies embarquées et les standards de l'Industrie 4.0. Ce système permet l'ouverture automatisée d'une porte à l'aide d'une carte RFID, tout en assurant une traçabilité complète des accès et une gestion centralisée des utilisateurs via une plateforme web.

Objectifs techniques spécifiques

- Mettre en place un système de lecture d'identifiants RFID à l'aide de l'ESP32.
- **Validation d'accès en ligne** : Transmettre l'UID au serveur distant via une requête HTTP POST pour vérification dans une base de données MySQL.
- Commander une serrure électrique via un relais lorsque l'accès est autorisé.
- Enregistrer chaque tentative d'accès (UID, date, heure, résultat) dans la base de données pour assurer la traçabilité.
- Développer une interface web en PHP pour :
 - Gérer les cartes RFID (ajout, suppression, consultation).

- Visualiser les historiques d'accès.
- Contrôler l'ouverture de la porte à distance si nécessaire.

Objectifs pédagogiques

- Mettre en œuvre des compétences techniques en systèmes embarqués, gestion de base de données, et développement web.
- Approfondir la compréhension de l'Internet des objets (IoT) appliqué à des cas concrets de sécurité et d'automatisation.
- Favoriser le travail en équipe autour d'un projet complet, de la conception à l'implémentation et à la validation finale.

0.1.3 Contraintes rencontrées

La mise en œuvre de ce projet a été confrontée à plusieurs contraintes, tant sur le plan matériel, technique qu'organisationnel. Ces difficultés ont nécessité des ajustements dans la planification, des efforts de coordination et une adaptation continue des choix techniques.

Contraintes financières et matérielles

- **Budget limité** : Le projet ne disposait pas de tous les composants nécessaires au départ. Certains éléments ont dû être achetés par les membres de l'équipe à leurs propres frais, notamment les câbles, les plaques d'essai (breadboards), et la serrure électrique.
- **Disponibilité du matériel** : Le microcontrôleur ESP32 et les lecteurs RFID disponibles présentaient des défauts ou des pannes. Plusieurs modules ont été testés avant d'en trouver un fonctionnel.

Contraintes techniques

- **Problèmes de compatibilité** : La combinaison entre certains modèles de lecteurs RFID et les modules ESP32 n'était pas toujours stable, ce qui a nécessité des essais multiples et une adaptation du code.
- **Connexion Wi-Fi instable** : Le système dépend d'une connexion réseau fiable. Des interruptions de connexion ont été observées, perturbant la communication entre l'ESP32 et le serveur.

Contraintes humaines et organisationnelles

- **Temps restreint** : Le projet a été mené en parallèle avec d'autres cours, ce qui a réduit le temps dédié au développement, aux tests et à la rédaction du rapport.

0.2 Architecture technique



0.2.1 Composants Matériels

ESP32 (Microcontrôleur)

L'ESP32 est le cerveau du système matériel. Ses fonctions principales sont :

- La lecture des données d'un badge RFID via un module RFID (ex : RC522).
- La connexion au réseau WIFI pour communiquer avec le serveur.
- La réception de la réponse du serveur (GRANTÉ/REFUSÉ) et l'activation du relais et de la serrure (pour déverrouiller la porte) en cas d'accès autorisé.

Module RFID

Le module RFID utilise des ondes radio pour lire l'UID (identifiant unique) d'un badge. Le module RC522 est couramment utilisé pour les fréquences de 13.56 MHz.

Connectivité WiFi

La connexion WiFi permet à l'ESP32 d'envoyer des requêtes HTTP au serveur, notamment l'envoi de l'UID lu.

0.2.2 Composants Logiciels

Serveur PHP / Interface Web

Le serveur, basé sur PHP, est responsable du traitement des requêtes de l'ESP32 et de la gestion de l'interface administrateur.

- **interface.php** :
 - Reçoit l'UID envoyé par l'ESP32 via une requête HTTP POST.
 - Interroge la base de données pour vérifier si l'UID est autorisé.
 - Renvoie une réponse (GRANTÉ/REFUSÉ) au format JSON ou texte à l'ESP32.
- **admin.php** :

- Fournit une interface web pour la gestion des badges (ajout/suppression d'UID, consultation des logs).
- Accès réservé aux administrateurs avec authentification requise.

Base de données MySQL

La base de données MySQL stocke les UID autorisés et l'historique des accès.

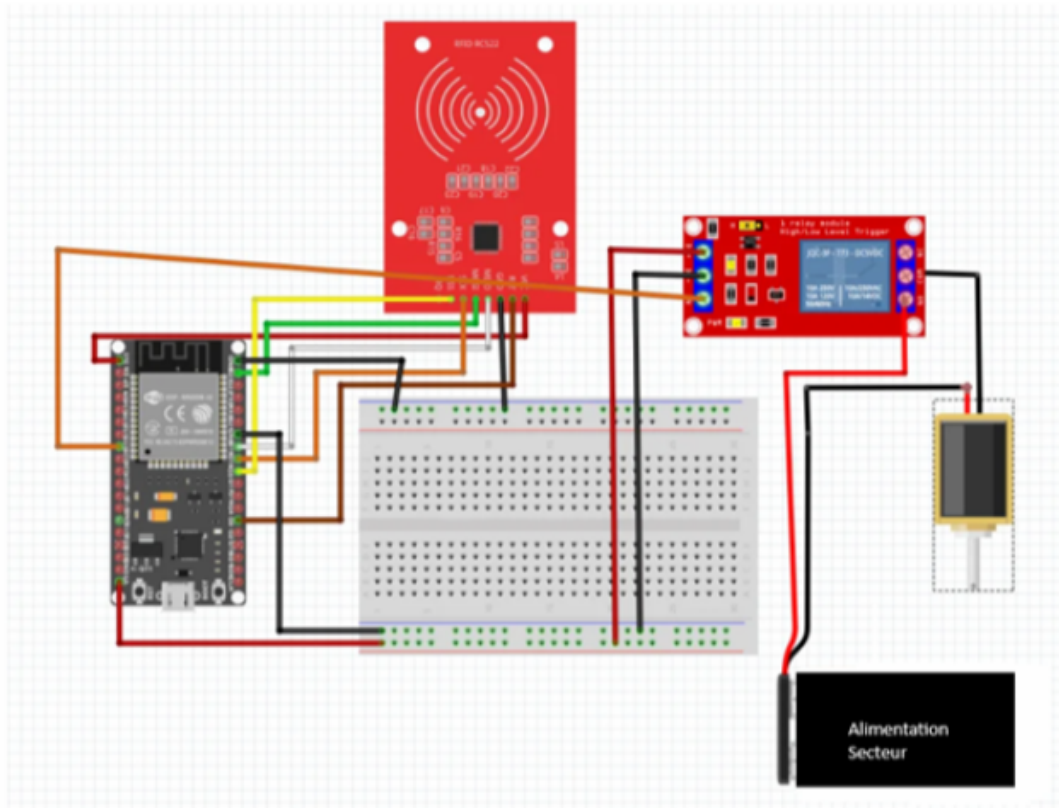
- **badges_authorized** : Contient les UID valides (ex : uid, nom_utilisateur, date_ajout).
- **logs_access** : Enregistre l'historique des tentatives d'accès (ex : uid, date_heure, statut).

0.2.3 Fonctionnement Global (Workflow)

Le processus de contrôle d'accès se déroule comme suit :

1. Lecture du badge : L'ESP32 lit l'UID du badge via le module RFID.
2. Envoi au serveur : L'ESP32 envoie l'UID au serveur via une requête HTTP POST (ex : `http://serveur.com/interface.php?uid=ABC123`).
3. Vérification PHP : Le script PHP (interface.php) interroge la base de données MySQL pour vérifier l'existence et la validité de l'UID.
4. Réponse à l'ESP32 :
 - Lorsque l'UID est validé, le serveur répond "GRANTÉ" et l'ESP32 commande le relais qui alimente la serrure, permettant l'ouverture.
 - Si l'UID n'est pas trouvé ou non autorisé, le serveur renvoie "REFUSÉ". L'accès est alors refusé.

0.3 Schéma du montage utilisé



0.3.1 Inventaire des composants

- **ESP32** : Microcontrôleur Wi-Fi pour lire la carte RFID et contrôler le relais.
- **Module RFID RC522** : Lit les cartes RFID (badges) et envoie les données à l'ESP32.
- **Module Relais 1 canal** : Sert d'interrupteur contrôlé par l'ESP32 pour allumer/éteindre la serrure.
- **Serrure électromagnétique / Solénoïde** : Actionné par le relais pour déverrouiller une porte.
- **Adaptateur secteur 220V → 12V** : Transforme la tension du secteur domestique (220V) en 12V pour alimenter la serrure. Fournit plus de puissance et de stabilité qu'une pile.
- **Breadboard + fils de connexion** : Permet les connexions sans soudure entre tous les composants.

0.3.2 Explication du fonctionnement

- **Lecture RFID** : L'ESP32 lit un badge RFID via le module RC522.
- **Vérification** : Si l'UID de la carte est reconnu, l'ESP32 active le relais.

- **Activation serrure** : Le relais laisse passer le courant de l'adaptateur 12V vers la serrure → elle se déverrouille.
- **Sécurité** : Une fois le délai écoulé, le relais se désactive → la serrure se verrouille à nouveau.
- **Alimentation séparée** : Le solénoïde est alimenté par l'adaptateur 12V, pas par l'ESP32, pour éviter de surcharger le microcontrôleur.
- **GND commun** : Le GND (négatif) de l'adaptateur est relié au GND du montage (ESP32, relais, etc.) pour assurer un retour de courant correct.

0.4 Logiciels et outils de développement

0.4.1 Environnement local avec XAMPP

Dans ce projet, aucun serveur distant n'est utilisé. Tout fonctionne en local grâce à XAMPP, qui joue le rôle de serveur de développement. XAMPP fournit les éléments suivants :

- **Apache** : serveur web pour exécuter les fichiers PHP.
- **MySQL** : système de gestion de base de données.
- **phpMyAdmin** : interface web pour gérer la base de données facilement.

0.4.2 Les fichiers PHP

interface.php

Ce fichier est contacté par l'ESP32 via une requête HTTP.

- L'ESP32 envoie une requête POST contenant l'UID du badge.
- Le script interface.php effectue les actions suivantes :
 1. Récupère l'UID envoyé par l'ESP32.
 2. Interroge la base de données pour voir si cet UID est autorisé.
 3. Renvoie une réponse en texte simple : "GRANTED" ou "REFUSE".

admin.php

Ce fichier propose une interface web pour l'administrateur.

- Accessible localement via un navigateur (ex : `http://localhost/admin.php`).
- Permet à l'administrateur de :
 - Consulter les badges autorisés.
 - Ajouter / supprimer des UID.
 - Afficher les logs des tentatives d'accès.

- Utilise des formulaires HTML combinés à du code PHP pour interagir avec la base de données.

0.4.3 Base de données MySQL

Deux tables principales sont utilisées dans ce projet :

- **badges_autorises** : Contient tous les UID autorisés à accéder.
- **logs_access** : Archive toutes les tentatives d'accès, qu'elles soient acceptées ou refusées.

uid	date_heure	statut
A1B2C3	2024-05-25 10 :12 :00	GRANTE
XYZ123	2024-05-25 10 :13 :00	REFUSE

0.4.4 Résumé du fonctionnement logiciel local

1. Lecture du badge : L'ESP32 lit l'UID via le module RFID.
2. Envoi au serveur : L'ESP32 envoie l'UID en HTTP POST à interface.php.
3. Vérification dans MySQL :
 - Si l'UID est reconnu → le script répond "GRANTED".
 - Sinon → réponse "REFUSE".
4. Réaction de l'ESP32 : En cas de réponse "GRANTED", la serrure est déverrouillée via relais.
5. Enregistrement : Chaque tentative est enregistrée dans la table logs_access.

0.5 Pipeline de Développement

Le développement du système s'est déroulé en plusieurs étapes clés :

0.5.1 Développement Côté Serveur

- Installation d'un environnement local via XAMPP (Apache + PHP + MySQL)
- Création des tables nécessaires (utilisateurs, logs)
- Implémentation d'un serveur PHP gérant les requêtes HTTP envoyées par l'ESP32.
- Création d'une interface d'administration web pour la gestion des accès

0.5.2 Programmation de l'ESP32

- Programmation via Arduino IDE
- Lecture des badges RFID
- Connexion WiFi sécurisée et envoi des requêtes HTTP

0.5.3 Intégration et Tests

- Connexion de l'ESP32 au réseau WiFi
- Test de communication avec le serveur
- Vérification du fonctionnement global du système
- Tests fonctionnels de l'interface web

0.5.4 Configuration Matérielle et Câblage

- Montage des composants sur breadboard
- Connexion du lecteur RFID à l'ESP32
- Montage de la serrure et du relais

0.6 Résultats et conclusions

0.6.1 Résultats obtenus

Le système de contrôle d'accès a été implémenté avec succès et a permis de valider les points suivants :

- Lecture correcte des cartes RFID à l'aide du module RC522.
- Communication stable entre l'ESP32 et le serveur local (XAMPP) via requêtes HTTP POST.
- Réponses appropriées du serveur ("GRANTED" ou "REFUSE") selon les données de la base MySQL.
- Déclenchement physique du relais et de la serrure en cas d'autorisation.
- Enregistrement automatique des tentatives d'accès dans la base de données.
- Interface web fonctionnelle pour gérer les badges et consulter l'historique.

Ces résultats montrent que le système est fonctionnel, fiable et modulable pour une utilisation de base dans un environnement local.

0.6.2 Conclusions

Ce projet a permis de :

- Mettre en œuvre une solution complète d'authentification par RFID.
- Comprendre l'interaction entre microcontrôleur, réseau WiFi, base de données, et interface web.
- Proposer un modèle simple mais extensible de contrôle d'accès.

Le système remplit ses objectifs initiaux : sécuriser un accès en validant une identité à l'aide d'un badge RFID, avec un retour en temps réel.