



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

المدرسة الوطنية العليا للتكنولوجيات المتقدمة
Ecole Nationale Supérieure des Technologies Avancées



Département Génie Électrique et Informatique Industrielle

Mémoire de fin d'étude en vue de l'obtention du diplôme MASTER

Filière :
Automatique

Spécialité :
Automatique et Informatique Industrielle

Thème :

**Développement d'une application mobile
pour la supervision d'un système de
récupération des boîtes défectueuses**

Réalisé par :
HABHOUB Safouane & MESSALI Idir

Soutenu le 22 / 06 / 2025

Membres du jury :

Nom et Prénom	Établissement	Grade	Qualité
M. ZAOUI Abdelhalim	ENSTA	Professeur	Président
M. CHERABIT Nouredine	ENSTA	MAA	Promoteur
M. DERMOUCHE Reda	ENSTA	MCA	Co-Promoteur
M. ABNOUN Hichem	TCHIN LAIT Candia	Expert métier	Co-Promoteur
Mme. BOULAAM Karima	ENSTA	MCB	Examinatrice
Mme. OUSAADI Zahira	ENSTA	MAA	Examinatrice

Année universitaire : 2024/2025

بسم الله الرحمن الرحيم

الحمد لله حمداً طيباً مباركاً فيه، كما ينبغي لجلال وجهه وعظيم سلطانه، والصلاة والسلام على سيدنا محمد، خير من دعا إلى العلم والعمل، وعلى آله وصحبه أجمعين.

نحمد الله تعالى الذي بنعمته تتم الصالحات، ونشكره شكراً يليق بجلاله وعظيم فضله، إذ يَسِّرَ لنا إتمام هذا البحث، الذي نسأل الله أن يجعله خالصاً لوجهه الكريم، وأن ينفع به، إنه ولي ذلك والقادر عليه.

Dans cette même gratitude qui emplit nos cœurs envers Allah, nous tenons également à exprimer notre sincère reconnaissance à ceux qui nous ont accompagnés et soutenus tout au long de ce chemin.

À nos chers parents, qui nous ont donné la vie, qui nous aiment et nous soutiennent toujours avec patience et amour.

À nos frères et sœurs, qui nous apportent de la joie et qui sont toujours là pour nous.

À notre famille, qui nous encourage et nous donne la force d'avancer.

À nos amis, qui ont partagé avec nous les bons moments et les moments plus difficiles, et qui sont restés à nos côtés avec fidélité.

Safouane et Idir

Remerciment

Nous tenons à exprimer notre sincère gratitude à toutes les personnes qui ont contribué à la réalisation de ce travail.

Nous exprimons notre reconnaissance à **M. CHERABIT Nouredine**, notre encadrant académique, pour son soutien méthodologique, sa disponibilité et sa bienveillance.

Nos sincères remerciements vont également à **M. DERMOUCHE Reda**, notre co-encadrant, pour ses conseils avisés, son suivi rigoureux et son implication précieuse dans l'avancement de notre travail.

Nous remercions particulièrement notre co-encadrant industriel **M. ABNOUN Hichem** pour sa disponibilité, son accompagnement et ses précieux conseils tout au long de notre projet.

Un grand merci à toute l'équipe de **SPA Tchin Lait Candia** pour leur accueil chaleureux, leur collaboration et leur accompagnement durant notre présence en entreprise.

Enfin, nous remercions nos familles et amis pour leur soutien moral et leur encouragement tout au long de notre parcours académique.

ملخص

يهدف هذا المشروع إلى تطوير منصة للإشراف والتحكم عن بُعد في نظام استرجاع العلب المعيبة. بعد دراسة نظرية للتقنيات المتعلقة بالأتمتة والواجهات، تم تطوير واجهة مستخدم محمولة بواسطة Node-Red على جهاز Raspberry Pi. تتيح هذه الواجهة إدارة المستخدمين، المراقبة الفورية والوصول إلى سجل الأحداث. تم تفعيل الوصول عن بُعد عبر Cloudflared. لاختبار النظام في ظروف واقعية.

الكلمات المفتاحية: الإشراف - العلب المعيبة - واجهة مستخدم محمولة - التحكم عن بُعد.

Résumé

Ce projet consiste à développer une application de supervision et de contrôle à distance d'un système de récupération de boîtes défectueuses. Après une étude théorique sur les technologies industrielles et les interfaces, une IHM mobile a été développée avec **Node-RED** sur **Raspberry Pi 400**. Elle permet la gestion des utilisateurs, la visualisation en temps réel et l'historique du système. Enfin, l'accès distant a été mis en place via **Cloudflared** pour valider la solution en conditions réelles.

Mots clés : Supervision mobile, Node-RED, Raspberry Pi, Cloudflared, boîtes défectueuses

Abstract

This project aims to create a remote supervision and control platform for a defective box recovery system. After theoretical research on industrial technologies and interfaces, a mobile HMI was built using Node-RED on a Raspberry Pi. It enables user management, real-time monitoring, and event history. Remote access was finally established through Cloudflared to test the system in real conditions.

Keywords : Mobile supervision, Node-RED, Raspberry Pi, Cloudflared, defective boxes

Table des matières

Liste des tableaux	2
Table des figures	3
1 Introduction	1
2 La supervision mobile	1
3 Environnement de développement de l'interface : Node-RED	2
3.1 Comparaison avec d'autres solutions	2
3.2 Justification du choix	2
4 Protocole de communication : S7comm	3
4.1 Comparaison avec d'autres protocoles industriels	3
4.2 Justification du choix de S7comm	3
5 Plateforme matérielle : Raspberry Pi	4
5.1 Comparaison avec d'autres plateformes	4
5.2 Justification du choix	5
6 Solution d'accès distant sécurisé : Cloudflared	5
6.1 Comparaison avec d'autres solutions	5
6.2 Justification du choix	5
7 Architecture globale	6
8 Installation et configuration des outils	6
8.1 Installation de Node-RED sur Raspberry Pi	6
8.2 Configuration de Node-RED	6
8.3 Mise en place de Cloudflared	7
9 Développement de l'application de supervision	8
9.1 Organisation des vues fonctionnelles	8
9.2 Communication avec l'automate Siemens (S7comm)	13
9.3 Intégration de la base de données (MySQL)	14
9.3.1 Structure de la base de données	14
9.3.2 Configuration du nœud MySQL dans Node-RED	15
9.3.3 Branchement du nœud MySQL avec les autres blocs	15
9.3.4 Historique des alarmes – exemples réels	16
9.4 Déploiement distant sécurisé et contrôle des accès utilisateurs	16
9.5 Sécurisation multi-niveaux et gestion avancée des rôles utilisateurs	16
9.5.1 Premier niveau : Authentification de Node-RED (via <code>settings.js</code>)	16
9.5.2 Deuxième niveau : Authentification interne	17
10 Conclusion	18
Bibliographie	19
Annexes	20

Liste des tableaux

1	Comparaison entre différents environnements de supervision	2
2	Comparaison de protocoles de communication industriels	3
3	Comparaison entre différentes plateformes matérielles	4
4	Comparaison entre différentes solutions d'accès distant sécurisé	5

Table des figures

1	Exemple d'interface graphique de Node-RED pour la création du flux de l'Identification et Attribution des Rôles Utilisateurs	2
2	Exemple de configuration des nœuds S7comm dans Node-RED	3
3	Vue extérieure du Raspberry Pi 400	4
4	Vue interne de la carte du Raspberry Pi 400	4
5	Schéma de l'architecture globale retenue	6
6	Liste des nœuds installés Node-RED	7
7	Configuration du tunnel Cloudflared sur Raspberry Pi	7
8	Saisie des identifiants de connexion	9
9	Connexion acceptée – accès autorisé	9
10	Connexion refusée – identifiants incorrects	9
11	Vue de sélection du mode de fonctionnement	9
12	Interface de supervision de la Station1 (Guidage)	10
13	Interface de supervision de la Station2 (Découpage)	10
14	Interface de supervision de la Station3 (Contrôle qualité)	10
15	Affichage du pH en temps réel	10
16	Liste des alarmes actives avec horodatage	11
17	Graphique de performance de récupération	11
18	Ajout d'un utilisateur	12
19	Liste des utilisateurs	12
20	Modification d'un utilisateur	12
21	Interface de déconnexion sécurisée	12
22	Configuration d'un nœud S7comm:read dans Node-RED	13
23	Structure des tables de la base de données MySQL	14
24	Configuration du nœud MySQL dans Node-RED	15
25	Connexion du nœud MySQL avec les autres nœuds du flow	15
26	Extrait des alarmes enregistrées le 5 mai	16
27	Extrait des alarmes enregistrées le 20 mai	16
28	Page d'authentification intégrée à Node-RED	17
29	Programmation de l'identification et de la gestion des rôles dans Node-RED . .	1
30	Code Node-RED pour la vérification des identifiants et l'attribution des rôles . .	2
31	Affichage conditionnel des groupes de l'interface selon le rôle utilisateur	3
32	Schéma Node-RED pour l'ajout, la modification, l'affichage et la suppression d'utilisateurs	3
33	Activation de SSH et VNC via l'outil de configuration de Raspberry Pi	4
34	Commandes d'installation de Node-RED	4
35	Commandes pour Cloudflared	5

1 Introduction

La transformation des pratiques industrielles induite par les technologies numériques a été considérable. De nombreuses entreprises sont désormais en cours de transition vers des systèmes plus intelligents, plus automatisés et plus connectés. Cette montée en gamme, souvent désignée par le terme d'Industrie 4.0 [1], repose sur l'intégration de l'informatique et de la communication dans les processus automatisés, tout en transformant en profondeur la façon dont on supervise les installations industrielles. Ce qui était limité à des postes éloignés dans des salles de contrôle n'est plus aujourd'hui que mobile, flexible et accessible en tous lieux.

Les exigences actuelles appellent à une réactivité accrue, une meilleure visibilité sur les processus et une capacité de réaction rapide aux dysfonctionnements. Reposant sur la mobilisation des technologies numériques disponibles (smartphone, tablette, ordinateurs portables), la supervision mobile permet de suivre un système tout en respectant un processus de contrôle de la qualité et du fonctionnement. Elle peut ainsi être perçue comme un outil d'amélioration de la performance industrielle, notamment dans les domaines de la maintenance et de la qualité [2].

C'est dans ce cadre que le présent projet de master s'inscrit. Il s'agit de concevoir et mettre en œuvre une solution de supervision mobile pour un système automatisé de récupération de boîtes défectueuses au sein de l'entreprise Tchén Lait Candia, comportant plusieurs étapes : guidage des boîtes, découpe, récupération du contenu, contrôle qualité et tri. La supervision en temps réel des étapes d'un système automatisé de production est importante pour garantir le bon fonctionnement de celui-ci tant en efficacité.

Pour cela, une application mobile a été développée, permettant à un opérateur, ou un technicien, de superviser à distance l'état du système pour intervenir au besoin, d'en consulter les valeurs critiques (pH, etc.), de recevoir des alertes sur les anomalies éventuelles, ainsi que d'analyser les performances globales. Les outils et technologies retenus ont été choisis selon une combinaison de critères de fiabilité et de simplicité, de compatibilité avec les systèmes existants ainsi que de sécurité d'accès à distance.

2 La supervision mobile

L'évolution des technologies de communication et d'information a eu un impact sur les systèmes de travail dans l'industrie. Parmi ces évolutions, la supervision mobile s'affirme comme un pas en avant. Elle permet aux opérateurs, techniciens et ingénieurs de consulter à distance et en temps réel les informations critiques d'un système de production, directement depuis un smartphone, une tablette ou encore un ordinateur portable connecté [2, 3].

Grâce à cette mobilité, il est désormais possible de garder un œil sur le fonctionnement d'un équipement, d'être alerté en cas de problème, de consulter des indicateurs de performance et d'intervenir rapidement sans se trouver devant une interface fixe. Une telle capacité d'intervention améliore sans nul doute la réactivité des équipes et limite les arrêts suite à un dysfonctionnement.

Pour mettre en place ce type de supervision, il est important de bien choisir les outils, les technologies et les protocoles adaptés. Ce choix conditionne la performance, la sécurité et la compatibilité de l'application. Pour cela, nous avons opté pour **Node-RED**, une solution légère et flexible, bien adaptée aux besoins de la supervision mobile industrielle.

3 Environnement de développement de l'interface : Node-RED

Node-RED est un logiciel open source qui propose des solutions pour l'*Internet of Things* (IoT) et l'automatisation des flux de données. Basé sur une interface graphique de type *low-code*, il permet de créer la logique fonctionnelle à l'aide de blocs appelés *nœuds*. Il supporte nativement les protocoles MQTT, Modbus, HTTP et S7comm pour communiquer avec les automates Siemens [4].

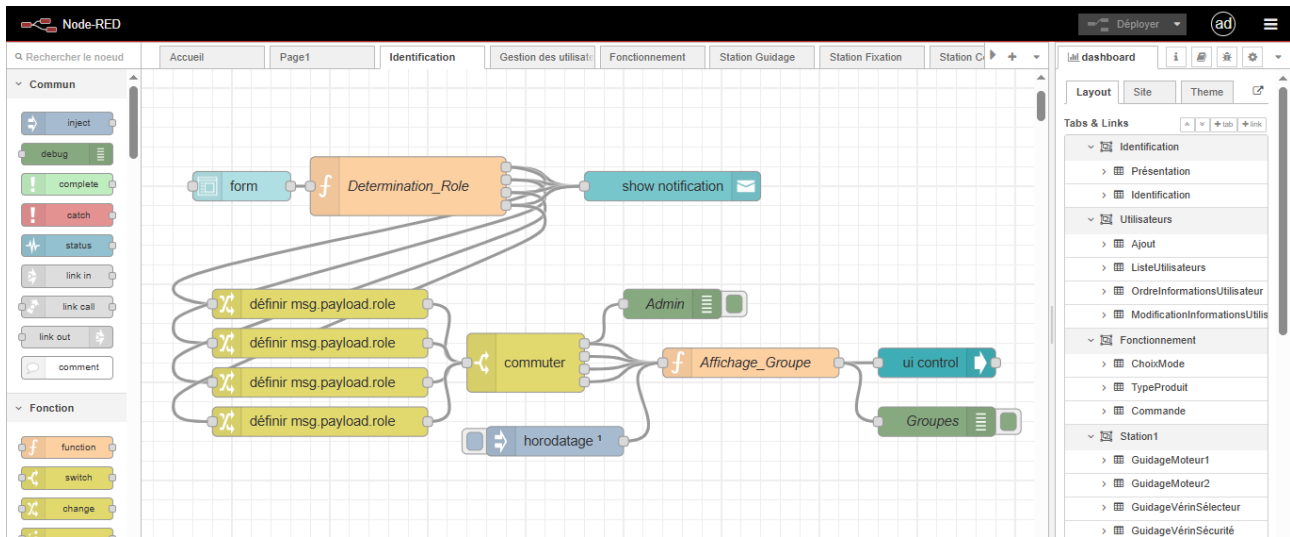


FIG. 1 : Exemple d'interface graphique de Node-RED pour la création du flux de l'Identification et Attribution des Rôles Utilisateurs

3.1 Comparaison avec d'autres solutions

Le tableau 1 compare brièvement Node-RED, Ignition et WinCC Unified [5, 6, 7].

TAB. 1 : Comparaison entre différents environnements de supervision

Critère	Node-RED	Ignition	WinCC Unified
Licence	Open source, gratuit	Payante	Payante (via Siemens)
Déploiement sur Raspberry Pi	Oui	Non recommandé	Non compatible
Intégration S7-1200	Oui (S7comm)	Oui (OPC UA)	Native
Interface graphique web	Oui (Dashboard)	Oui	Oui
Orientation IIoT	Oui	Oui	Partielle

3.2 Justification du choix

Node-RED a été retenu pour plusieurs raisons : sa prise en main aisée, sa compatibilité avec le S7-1200 via le protocole S7comm, la personnalisation de ses interfaces web, ainsi que son déploiement facile sur Raspberry Pi.

4 Protocole de communication : S7comm

Le protocole S7comm est un protocole de communication propriétaire développé par Siemens pour permettre l'échange de données avec les automates de la gamme **S7-300**, **S7-1200**, et **S7-1500**. A l'aide de ce protocole, il est possible d'accéder directement aux zones mémoire de l'automate pour faciliter l'échange d'informations entre le API et une application tierce, telle qu'un outil de supervision [8].

4.1 Comparaison avec d'autres protocoles industriels

Le tableau 2 compare les fonctionnalités de S7comm et Modbus TCP [9, 10, 11].

TAB. 2 : Comparaison de protocoles de communication industriels

Protocole	S7comm	Modbus TCP
Compatibilité Siemens	Native (S7-1200, 1500)	Possible avec configuration
Sécurité native	Moyenne	Moyenne
Complexité	Faible	Moyenne

4.2 Justification du choix de S7comm

le choix du protocole **S7comm** repose sur plusieurs arguments :

- Compatibilité native des automates Siemens **S7-1200** avec ce protocole.
- La prise en charge native de **S7comm** par **Node-RED** via le module **node-red-contrib-s7**.
- La simplicité des développements logiciels et la faible consommation des ressources informatiques du système, ce qui est un avantage décisif pour une éventuelle exécution sur **Raspberry Pi**.

La figure 2 ci-après illustre la configuration des nœuds S7comm dans un flux Node-RED.

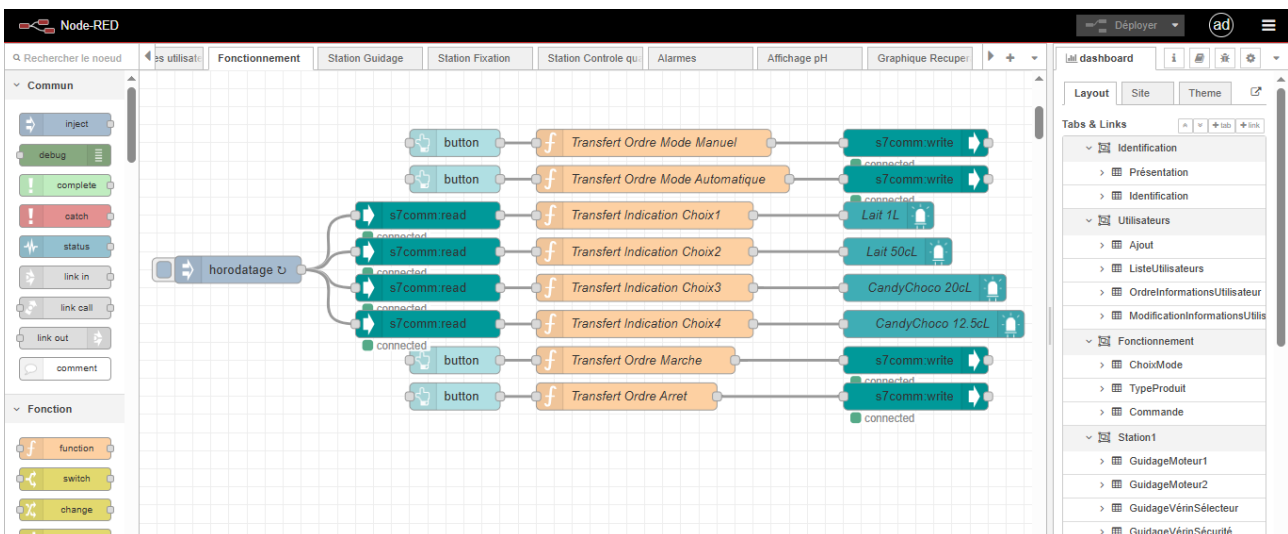


FIG. 2 : Exemple de configuration des nœuds S7comm dans Node-RED

5 Plateforme matérielle : Raspberry Pi

Le Raspberry Pi 400 est un nano-ordinateur peu coûteux, compact et intégré à un clavier, capable de faire fonctionner un système Linux complet. Il permet d'héberger Node-RED, d'héberger une base de données, et d'assurer la communication avec l'automate. Il dispose des interfaces réseau (Ethernet, Wi-Fi), des ports USB et GPIO [12], tout en offrant un format plus ergonomique, adapté aux environnements industriels ou éducatifs.

En outre, à l'intérieur de ce petit ensemble compact, un excellent processeur quadri-cœur et 4 Go de mémoire vive lui confèrent la possibilité de gérer de multiples tâches en même temps. Sur mon bureau, le Raspberry Pi 400 peut faire office de station de supervision locale, de terminal de programmation, ou encore de serveur d'applications légères. Son design intégré contribue à réduire l'encombrement des postes de travail [13].

Les deux images 3 et 4 suivantes présentent une vue extérieure du Raspberry Pi 400 ainsi qu'une vue interne de sa carte électronique.



FIG. 3 : Vue extérieure du Raspberry Pi 400

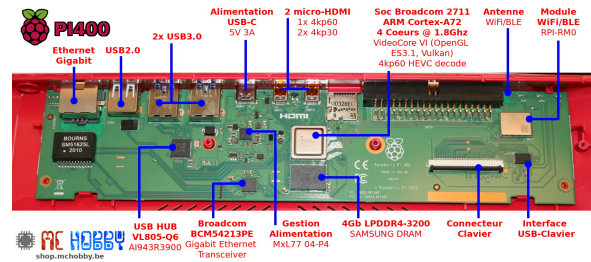


FIG. 4 : Vue interne de la carte du Raspberry Pi 400

5.1 Comparaison avec d'autres plateformes

Le tableau 3 montre une comparaison entre le **Raspberry Pi 400** et un **PC industriel (IPC)** selon plusieurs critères techniques et fonctionnels [13, 14, 15, 16].

TAB. 3 : Comparaison entre différentes plateformes matérielles

Critère	Raspberry Pi 400	PC Industriel (IPC)
Coût	Faible	Très élevé
Déploiement Node-RED	Oui	Oui
Système d'exploitation	Complet (Linux)	Complet
Connectivité réseau	Ethernet, Wi-Fi	Ethernet, Wi-Fi
Portabilité	Très bonne	Faible
Consommation électrique	Faible	Moyenne/Élevée

5.2 Justification du choix

Le **Raspberry Pi** apporte des caractéristiques suffisantes, à un prix raisonnable, une compatibilité avec les logiciels choisis, et une portabilité souhaitable. Il entre donc pleinement dans le besoin d'un prototype industriel connecté, favorisant une grande souplesse d'intégration.

6 Solution d'accès distant sécurisé : Cloudflared

Cloudflared est un tunnel sécurisé proposé par **Cloudflare**, permettant d'exposer un service local via un lien public **HTTPS** sécurisé sans ouvrir de ports sur le routeur. Cette solution apporte une sécurité renforcée, une gestion fine des accès et une grande simplicité de mise en œuvre [17].

6.1 Comparaison avec d'autres solutions

Pour mieux comprendre les avantages de **Cloudflared**, voici une comparaison avec une autre solution populaire d'accès distant sécurisé, **ngrok**, selon plusieurs critères clés [17, 18] (voir tableau 4).

TAB. 4 : Comparaison entre différentes solutions d'accès distant sécurisé

Critère	Cloudflared	ngrok
Sécurité HTTPS	Très élevée	Bonne
Coût	Gratuit	Gratuit/Payant
Facilité d'intégration	Très simple	Très simple
Ouverture de ports	Non requise	Non requise

6.2 Justification du choix

Cloudflared a été choisi pour sa simplicité d'intégration, sa sécurité renforcée (**HTTPS**), son intégration native avec **Node-RED** et son coût nul. C'est pour nous la solution appropriée pour sécuriser l'accès distant à l'interface de supervision mobile.

7 Architecture globale

La combinaison des trois éléments – **Node-RED**, **Cloudflared** et **Raspberry Pi** – constitue une architecture légère, flexible et entièrement compatible avec les exigences de l'IIoT. Elle permet de :

- Superviser en temps réel des données issues de l'automate Siemens S7-1200 ;
- Offrir un accès distant sécurisé via HTTPS ;
- Réduire les coûts de déploiement et de maintenance.

Architecture retenue

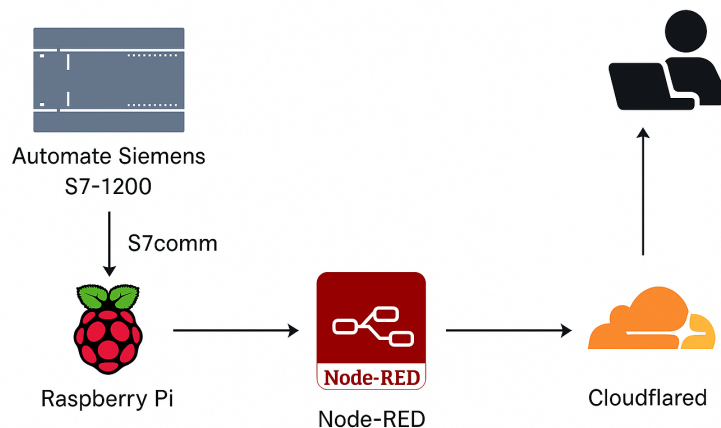


FIG. 5 : Schéma de l'architecture globale retenue

8 Installation et configuration des outils

8.1 Installation de Node-RED sur Raspberry Pi

La configuration de notre application de supervision mobile s'est réalisée sur un **Raspberry Pi 400**. Le système d'exploitation choisi est **Raspberry Pi OS**, léger et adapté aux performances de l'appareil. L'installation de **Node-RED** a été effectuée à l'aide du script officiel [19] fourni par la communauté **Node-RED**.

Après avoir finalisé l'installation des logiciels, **Node-RED** a été paramétré pour se lancer à l'aide de la commande suivante : `node-red`

8.2 Configuration de Node-RED

À la fin de l'installation, l'environnement Node-RED dispose de plusieurs modules pour assurer les fonctions de communication, de visualisation et de traitement des données :

- `node-red-dashboard` pour la création d'interfaces graphiques ;
- `node-red-contrib-s7` pour l'échange de données avec l'automate Siemens S7-1200 ;
- `node-red-node-mysql` pour la connexion avec la base de données locale.

Le gestionnaire de palette a permis, comme montrée dans la Figure 6, d'installer les modules nécessaires au développement.

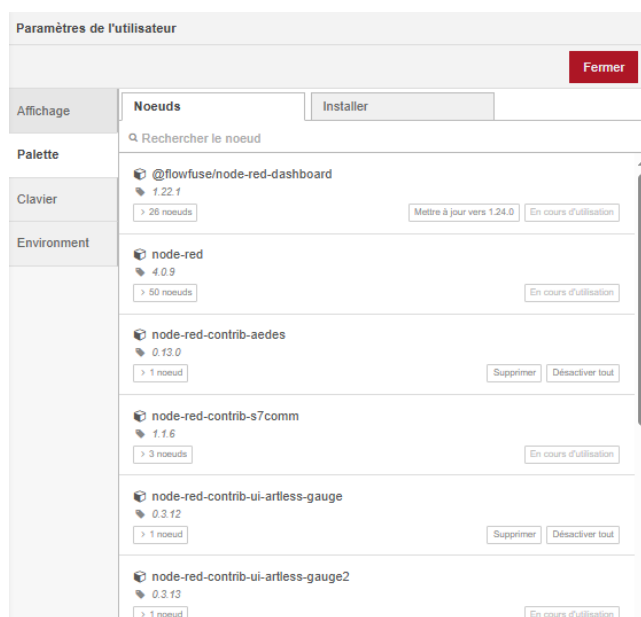


FIG. 6 : Liste des nœuds installés Node-RED

8.3 Mise en place de Cloudflared

Pour assurer un accès distant sécurisé à l'application, nous avons opté pour **Cloudflared**, le tunnel **HTTPS** proposé par **Cloudflare**. Cette solution permet d'exposer l'application via une **URL** publique, sans avoir à ouvrir de port sur le réseau local.

La figure 7 illustre la configuration de Cloudflared sur notre Raspberry Pi.

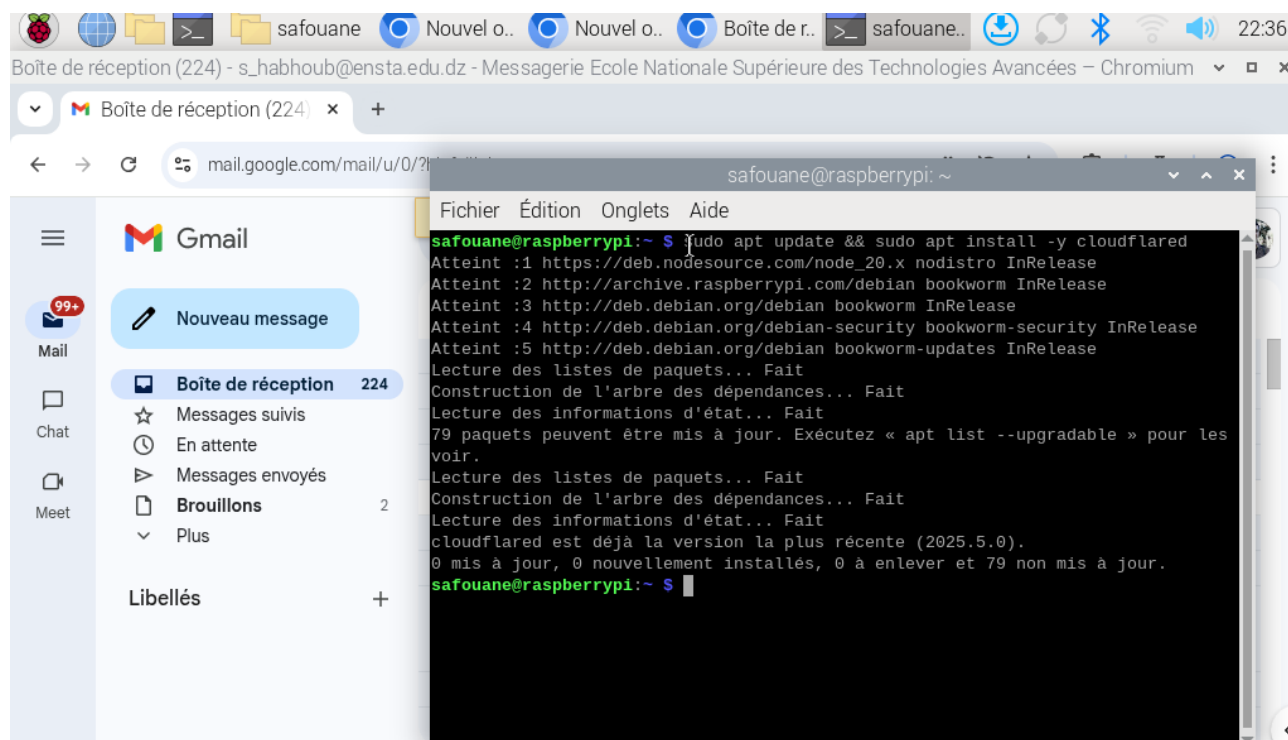


FIG. 7 : Configuration du tunnel Cloudflared sur Raspberry Pi

9 Développement de l'application de supervision

Une fois les outils nécessaires à la mise en place de l'application de supervision installés (**Node-RED**, ses modules complémentaires, ainsi que **Cloudflared**), une interface mobile intuitive, sécurisée et connectée à l'automate **Siemens S7-1200** via le protocole **S7comm** a été mise en œuvre. L'application intègre plusieurs interfaces fonctionnelles, conçues pour surveiller et piloter le système en temps réel, rendant ainsi son exploitation effective.

9.1 Organisation des vues fonctionnelles

L'application se compose de plusieurs onglets (vues) accessibles via un menu principal. Chaque vue est dédiée à une fonction précise.

Menu de navigation entre les vues L'application mobile dispose d'un menu latéral de navigation visible sur toutes les pages, permettant à l'utilisateur de passer rapidement d'une vue à une autre. Ce menu est dynamique : les droits d'accès sont limités selon les rôles (admin, user_control, user).

Le menu donne accès aux éléments suivants :

- Interface principal
- Mode de fonctionnement et Choix
- Supervision Station A (Guidage)
- Supervision Station B (Découpage)
- Supervision Station C (Contrôle qualité)
- Affichage des mesures de pH
- Alarmes et historique
- Performances du système
- Gestion des utilisateurs (admin uniquement)
- Déconnexion

a. Vue de connexion (Interface principal) Représente la première étape d'accès à l'application, où l'utilisateur saisit son identifiant et son mot de passe (figure 8). Si les informations sont correctes, l'utilisateur accède à l'interface de supervision (figure 9), sinon un message d'erreur apparaît (figure 10). Les droits d'utilisation sont attribués en fonction du rôle de l'utilisateur dans la base de données.

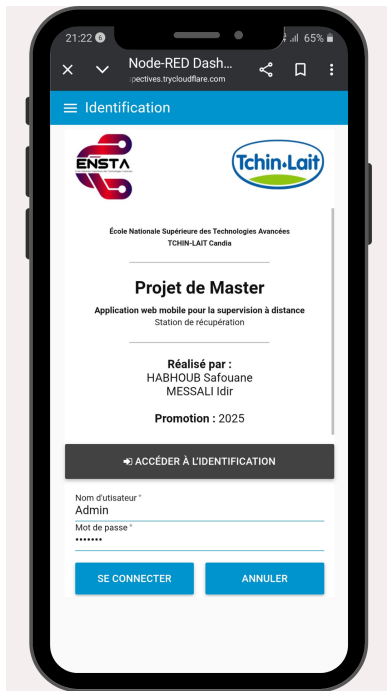


FIG. 8 : Saisie des identifiants de connexion



FIG. 9 : Connexion acceptée – accès autorisé



FIG. 10 : Connexion refusée – identifiants incorrects

b. Vue du mode de fonctionnement Permet de choisir entre le mode automatique et manuel, de contrôler la mise en marche et l'arrêt, et de visualiser le produit en temps réel.

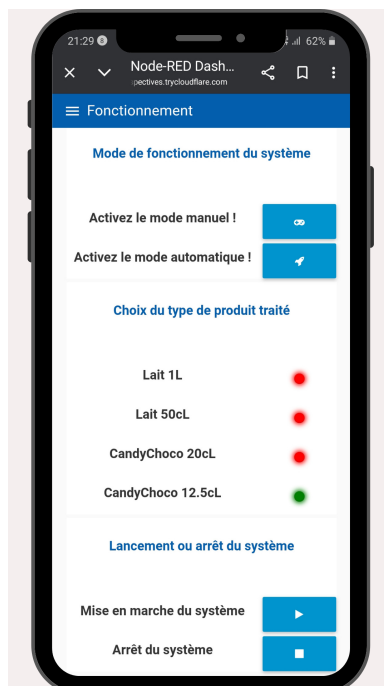


FIG. 11 : Vue de sélection du mode de fonctionnement

c. Supervision des stations (1, 2, 3) Chaque station dispose d'une interface dédiée affichant en temps réel les états des capteurs, moteurs, électrovannes, et niveaux analogiques.



FIG. 12 : Interface de supervision de la Station1 (Guidage)



FIG. 13 : Interface de supervision de la Station2 (Découpage)



FIG. 14 : Interface de supervision de la Station3 (Contrôle qualité)

d. Mesure de pH Cette interface permet un suivi précis de l'acidité du produit récupéré, garantissant sa conformité avec les normes de qualité. Les mesures sont affichées en temps réel sous forme numérique et graphique pour une meilleure lisibilité. Une alerte visuelle est également générée en cas de dépassement des seuils critiques définis.

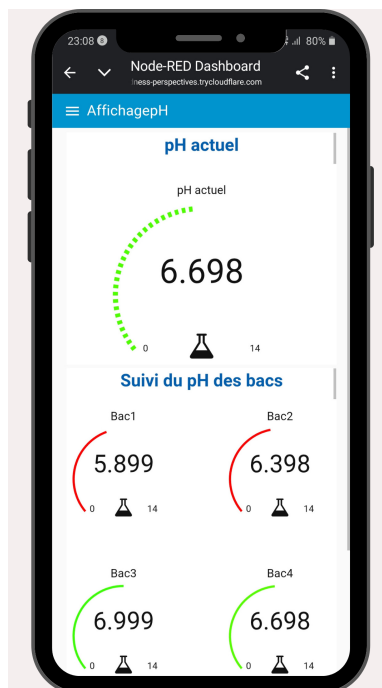


FIG. 15 : Affichage du pH en temps réel

e. Vue des alarmes Cette vue affiche la liste des alarmes actives du système, avec pour chacune un horodatage précis indiquant le moment exact de son déclenchement. Le statut de l'alarme (active ou résolue) est clairement présenté, ce qui facilite la compréhension de la situation en temps réel. Grâce à ces informations, les opérateurs peuvent identifier rapidement les anomalies et intervenir efficacement.

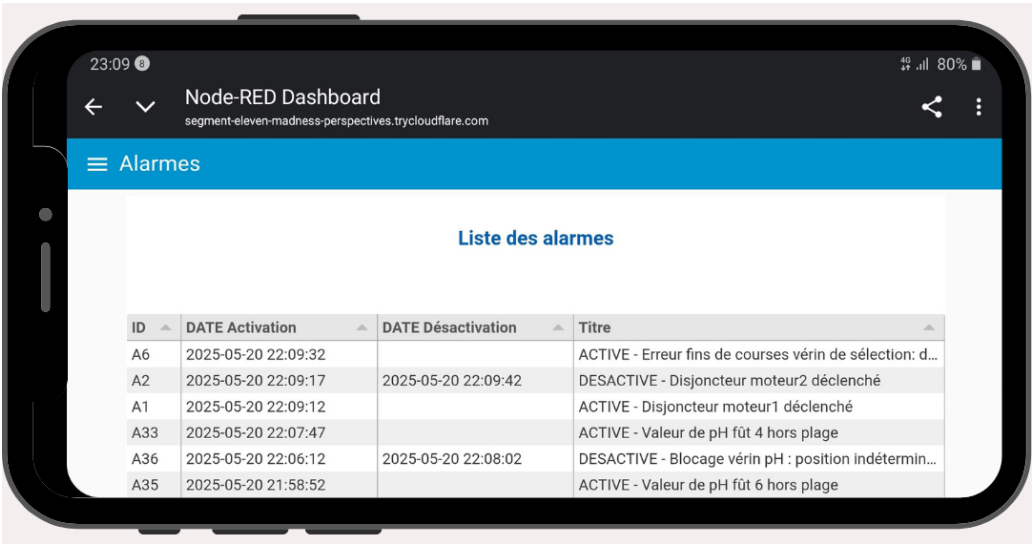


FIG. 16 : Liste des alarmes actives avec horodatage

f. Vue des performances Affiche un graphique représentant les quantités récupérées et la comparaison entre produit conforme et non conforme.



FIG. 17 : Graphique de performance de récupération

g. Vue de gestion des utilisateurs Accessible uniquement au rôle **admin**, permet de gérer les comptes utilisateurs via une interface dédiée. Les fonctionnalités proposées incluent :

- Création de nouveaux comptes.
- Suppression de comptes existants.
- Modification du rôle ou des identifiants d'un utilisateur.



FIG. 18 : Ajout d'un utilisateur

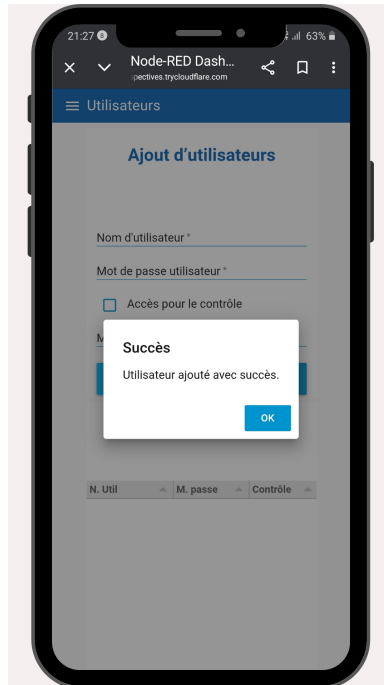


FIG. 19 : Liste des utilisateurs

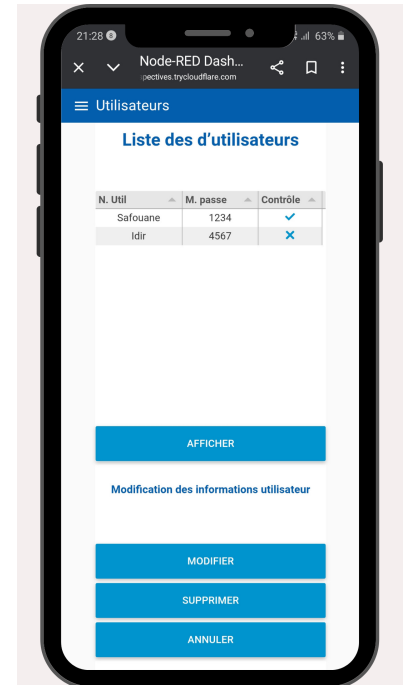


FIG. 20 : Modification d'un utilisateur

h. Vue de déconnexion Permet à l'utilisateur de se déconnecter de manière sécurisée en fermant sa session active. Une fois la déconnexion effectuée, l'utilisateur est redirigé vers l'écran de connexion.



FIG. 21 : Interface de déconnexion sécurisée

9.2 Communication avec l'automate Siemens (S7comm)

La communication entre Node-RED et l'automate Siemens S7-1200 est assurée par le module `node-red-contrib-s7`. Il permet :

- la lecture et l'écriture directe dans les blocs de données (DB),
- l'échange en temps réel des états des capteurs et commandes,

Pour établir cette communication, il est nécessaire de configurer les paramètres de connexion, notamment l'adresse IP de l'automate, le port de communication, le numéro de rack et l'emplacement (slot). Ces paramètres sont définis dans le nœud de configuration S7comm de Node-RED (voir la figure 22).

Signals:					
Input	Bit	Offset: 1	2	FinCourse_Vsel_Lait	:5
Input	Bit	Offset: 1	3	FinCourse_Vsel_CC	:6
Output	Bit	Offset: 0	7	Vsec+	:7
Output	Bit	Offset: 1	0	Vsec-	:8
Input	Bit	Offset: 1	4	FinCourse_Vsel_PHaut	:9
Input	Bit	Offset: 1	5	FinCourse_Vsel_Pbas	:10

FIG. 22 : Configuration d'un nœud S7comm :read dans Node-RED

9.3 Intégration de la base de données (MySQL)

Afin d'assurer un enregistrement fiable des événements critiques, une base de données **MySQL locale** a été intégrée à notre application de supervision. Celle-ci permet de centraliser les données liées aux utilisateurs, aux alarmes et à la production, tout en offrant une traçabilité complète, consultable à tout moment via l'interface Node-RED.

9.3.1 Structure de la base de données

La structure de la base est représentée dans la figure 23, qui montre l'organisation des trois tables principales :

- **utilisateurs** : login, mot de passe (chiffré), rôle (admin, usercontrol, user).
- **alarmes** : idAlarme, Code_Alarme, titre, date_Activation, date_Desactivation.
- **production** : quantités récupérées, conformes, non conformes.

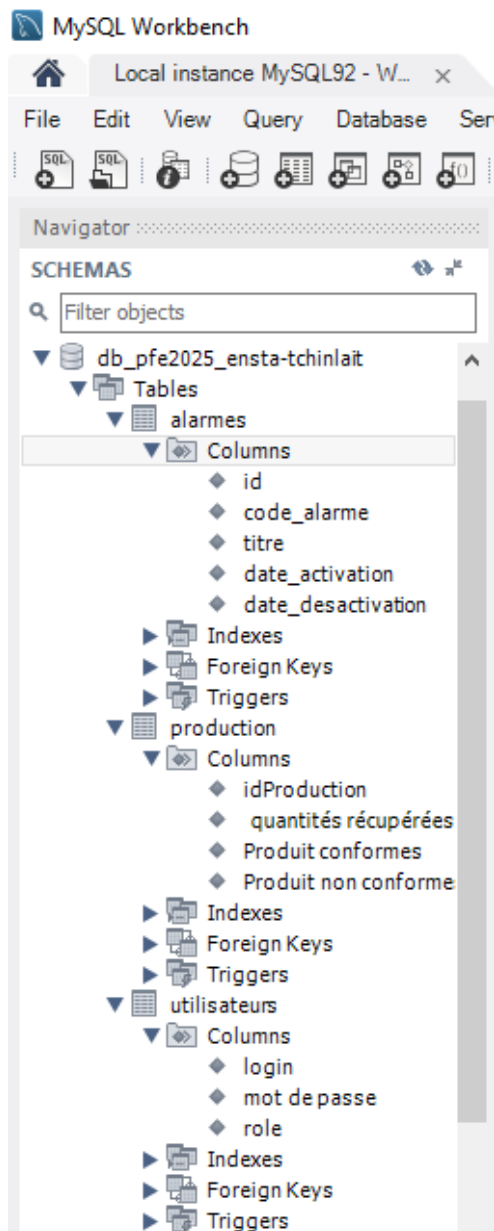


FIG. 23 : Structure des tables de la base de données MySQL

9.3.2 Configuration du nœud MySQL dans Node-RED

L'intégration MySQL s'appuie sur le module `node-red-node-mysql`, configuré comme montré dans la figure 24. Le nœud est connecté à la base de données avec les paramètres suivants :

- Port : 3306
- Utilisateur : root
- Mot de passe : ••••••
- Base : db_pfe2025_ensta-tchinlait

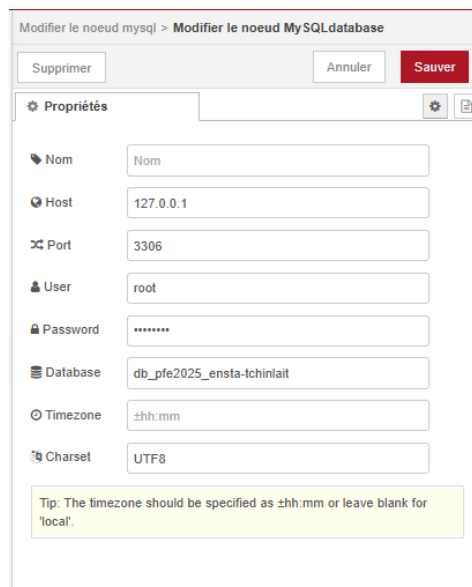


FIG. 24 : Configuration du nœud MySQL dans Node-RED

9.3.3 Branchement du nœud MySQL avec les autres blocs

Une fois configuré, le nœud MySQL est relié à d'autres nœuds fonctionnels dans Node-RED, comme illustré dans la figure 25.

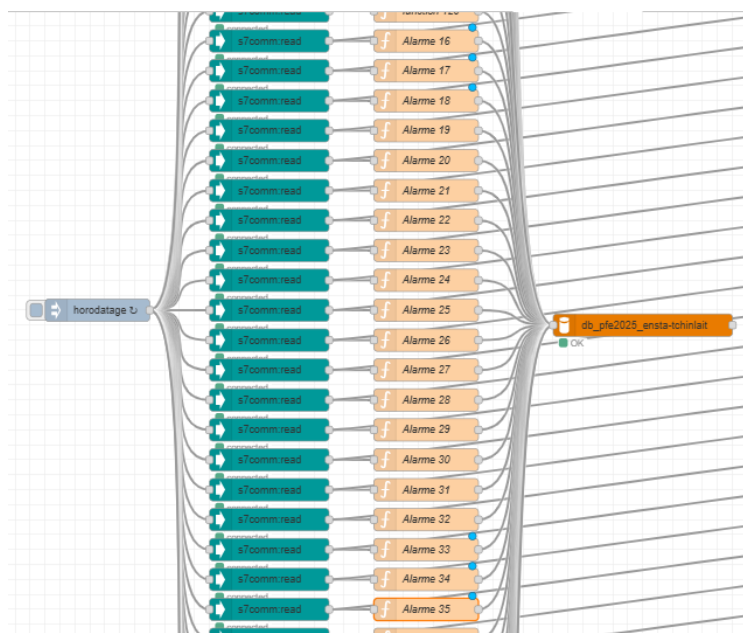


FIG. 25 : Connexion du nœud MySQL avec les autres nœuds du flow

9.3.4 Historique des alarmes – exemples réels

Deux extraits réels du contenu de la table `alarmes` sont présentés ci-dessous. Ils correspondent à des alarmes enregistrées automatiquement le **5 mai** (figure 26) et le **20 mai** (figure 27).

id	code_alarme	titre	date_activation	date_desactivation
1	A1	Disjoncteur moteur 1 déclenché	2025-05-05 19:04:12	
2	A1	Disjoncteur moteur 1 déclenché	2025-05-05 19:06:59	
3	A1	Disjoncteur moteur 1 déclenché	2025-05-05 19:07:53	2025-05-05 19:25:05
4	A1	Disjoncteur moteur 1 déclenché	2025-05-05 19:25:10	2025-05-05 19:32:46
5	A1	Disjoncteur moteur 1 déclenché	2025-05-05 19:34:28	2025-05-05 19:45:58
6	A2	Disjoncteur moteur 2 déclenché	2025-05-05 19:35:35	2025-05-05 19:47:13
7	A1	Disjoncteur moteur 1 déclenché	2025-05-05 19:46:18	
8	A1	Disjoncteur moteur 1 déclenché	2025-05-05 19:50:57	2025-05-05 19:51:36
9	A1	Disjoncteur moteur 1 déclenché	2025-05-05 19:52:02	2025-05-05 21:29:56
10	A1	Disjoncteur moteur 1 déclenché	2025-05-05 19:57:11	2025-05-05 21:30:01

FIG. 26 : Extrait des alarmes enregistrées le 5 mai

id	code_alarme	titre	date_activation	date_desactivation
397	A35	Valeur de pH fût 6 hors plage	2025-05-20 21:58:52	
398	A36	Blocage vérin pH : position ind...	2025-05-20 21:58:52	2025-05-20 22:00:21
399	A49	Défaut démarrage vérin pH	2025-05-20 22:00:42	2025-05-20 22:00:52
400	A36	Blocage vérin pH : position ind...	2025-05-20 22:00:51	2025-05-20 22:01:42
401	A49	Défaut démarrage vérin pH	2025-05-20 22:02:02	2025-05-20 22:02:22
402	A36	Blocage vérin pH : position ind...	2025-05-20 22:02:27	2025-05-20 22:03:12
403	A49	Défaut démarrage vérin pH	2025-05-20 22:03:47	2025-05-20 22:03:57
404	A36	Blocage vérin pH : position ind...	2025-05-20 22:03:57	2025-05-20 22:05:17
405	A49	Défaut démarrage vérin pH	2025-05-20 22:05:47	2025-05-20 22:06:07
406	A36	Blocage vérin pH : position ind...	2025-05-20 22:06:17	2025-05-20 22:08:03

FIG. 27 : Extrait des alarmes enregistrées le 20 mai

9.4 Déploiement distant sécurisé et contrôle des accès utilisateurs

Accès distant sécurisé avec Cloudflared

L'accès distant à l'interface mobile est rendu possible grâce à **Cloudflared**, qui crée un tunnel HTTPS.

- Lancement du tunnel via une ligne de commande.
- Génération d'un lien sécurisé de type `https://xxxx.trycloudflare.com`

9.5 Sécurisation multi-niveaux et gestion avancée des rôles utilisateurs

L'application de supervision exposée à distance via un tunnel sécurisé Cloudflared nécessite la mise en œuvre d'une double sécurité pour garantir un contrôle strict des accès. Deux niveaux de protection ont été définis :

9.5.1 Premier niveau : Authentification de Node-RED (via `settings.js`)

Lorsqu'un utilisateur tente d'accéder à l'URL publique générée par Cloudflared, il est automatiquement redirigé vers une page d'authentification intégrée à Node-RED. Ce système repose sur une configuration réalisée dans le fichier `settings.js`, où chaque utilisateur est défini avec un identifiant, un mot de passe (chiffré via `bcrypt`), et un rôle attribué.

Deux types d'accès sont ainsi définis à ce niveau :

Administrateur (admin) : Peut accéder à l'éditeur de flux Node-RED, visualiser et modifier les flux, déployer des changements.

Utilisateur lecture seule (user) : Peut accéder à l'éditeur de flux uniquement en lecture, sans possibilité de déploiement ni de modification.



FIG. 28 : Page d'authentification intégrée à Node-RED

9.5.2 Deuxième niveau : Authentification interne

Après l'authentification initiale sécurisée via le fichier `settings.js` de Node-RED, une seconde authentification est mise en place dans l'interface de connexion. Celle-ci repose sur une base de données MySQL contenant les identifiants et rôles des utilisateurs. Chaque rôle conditionne dynamiquement les vues accessibles et les droits d'action, permettant ainsi une supervision adaptée à chaque profil :

- **Rôle : admin**
 - o **Vues accessibles :**
 - * Vue de supervision complète (états des capteurs et actionneurs)
 - * Vue des commandes manuelle/automatique
 - * Vue de choix du produit
 - * Historique des alarmes
 - * **Vue de gestion des utilisateurs** (ajout, modification, suppression)
 - o **Droits :** Lecture/écriture sur tous les composants, accès complet à tous les paramètres et fonctions.
- **Rôle : usercontrol**
 - o **Vues accessibles :**
 - * Vue de supervision
 - * Vue des commandes manuelle/automatique
 - * Vue de choix du produit
 - o **Droits :** Lecture et écriture partielle (pas de gestion des utilisateurs).

- **Rôle : user**
 - o **Vues accessibles :**
 - * Vue de supervision (visualisation des états uniquement)
 - * Vue du produit sélectionné (lecture seule)
 - o **Droits :** Aucun contrôle, accès strictement en lecture seule.
- **Rôle : error**
 - o **Vues accessibles :** Aucune
 - o **Droits :** Accès refusé (identifiants invalides ou non reconnus).

10 Conclusion

Dans une démarche d'intégration des technologies mobiles dans la supervision des systèmes industriels, la réalisation de cette application demandée par l'entreprise Tchén Lait Candia représente une étape importante. Elle démontre qu'avec un matériel simple comme un **Raspberry Pi** et des outils accessibles comme **Node-RED**, il est possible de développer une solution sûre, robuste et intuitive pour le suivi d'un processus en temps réel, que ce soit en phase de test ou dans une utilisation future sur le terrain. Au-delà de cette dimension purement technique, ce travail met en lumière l'intérêt d'une interface bien conçue, adaptable aux rôles des différents utilisateurs et aussi aux besoins du milieu industriel. C'est cette combinaison entre technologie et ergonomie qui rend possible une supervision moderne, évolutive et ouverte à d'autres perspectives comme l'automatisation avancée, l'extension des connexions à distance ou encore l'amélioration de l'expérience utilisateur.

Perspectives

Les résultats obtenus constituent une base solide pouvant être perfectionnée ou étendue selon plusieurs perspectives, notamment :

- **Amélioration de l'expérience utilisateur.**
- **Extension vers d'autres lignes de production.**
- **Ajout d'un système de notifications intelligentes.**

Bibliographie

- [1] Henning Kagermann, Wolfgang Wahlster, and Johannes Helbig. Recommendations for implementing the strategic initiative industrie 4.0. [\[Lien\]](#), 2013. Consulté le 21 mai 2025.
- [2] Jay Lee, Behrad Bagheri, and Hung-An Kao. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. [\[Lien\]](#), 2015. Consulté le 21 mai 2025.
- [3] Li Da Xu, Eric L. Xu, and Ling Li. Industry 4.0: State of the art and future trends. [\[Lien\]](#), 2018. Consulté le 21 mai 2025.
- [4] Node-RED. Node-red official website. [\[Lien\]](#), 2024. Consulté en mai 2025.
- [5] Node-RED Community. Node-red : Flow-based programming for the internet of things. [\[Lien\]](#), 2024. Consulté en mai 2025.
- [6] Inductive Automation. Ignition : The industrial application platform for scada and iiot. [\[Lien\]](#), 2024. Consulté en mai 2025.
- [7] Siemens AG. Simatic wincc unified system. [\[Lien\]](#), 2024. Consulté en mai 2025.
- [8] Node-RED Contributors. node-red-contrib-s7: S7 communication node for node-red. [\[Lien\]](#), 2024. Consulté en mai 2025.
- [9] Siemens. S7 communication between simatic s7-1200 and simatic s7-300. [\[Lien\]](#), 2019. Consulté le 29 mai 2025.
- [10] Modbus Organization. Modbus messaging on tcp/ip implementation guide v1.0b. [\[Lien\]](#), 2006. Consulté le 29 mai 2025.
- [11] Modbus Organization. Modbus application protocol specification v1.1b3. [\[Lien\]](#), 2006. Consulté le 29 mai 2025.
- [12] Raspberry Pi Foundation. Raspberry pi 400. [\[Lien\]](#), 2020. Consulté en mai 2025.
- [13] Raspberry Pi Foundation. Documentation officielle raspberry pi. [\[Lien\]](#), 2025. Consulté en mai 2025.
- [14] Siemens. Industrial pcs (ipc). [\[Lien\]](#), 2025. Consulté en mai 2025.
- [15] Advantech. Industrial computing solutions. [\[Lien\]](#), 2025. Consulté en mai 2025.
- [16] Beckhoff. Industrial pcs - product overview. [\[Lien\]](#), 2025. Consulté en mai 2025.
- [17] Cloudflare. Cloudflared — secure tunnels. [\[Lien\]](#), 2024. Consulté le 29 mai 2025.
- [18] ngrok. ngrok documentation. [\[Lien\]](#), 2024. Consulté le 29 mai 2025.
- [19] Node-RED Community. Node-red documentation. [\[Lien\]](#), 2024. Consulté en mai 2025.

Annexes

Annexes

A. Extraits de la programmation dans Node-RED

Les figures suivantes présentent des extraits représentatifs de la programmation réalisée dans l'environnement Node-RED pour la supervision mobile du système automatisé.

A.1 – Gestion de l'identification des utilisateurs

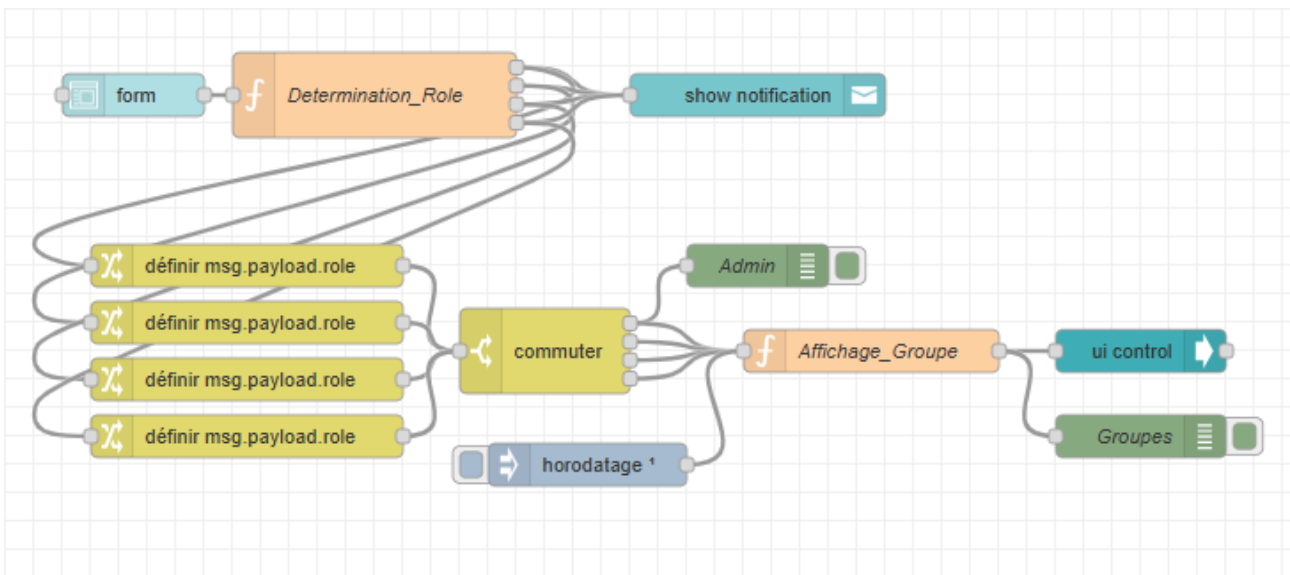


FIG. 29 : Programmation de l'identification et de la gestion des rôles dans Node-RED

Ce flow 29 permet de vérifier les identifiants saisis par l'utilisateur, puis d'assigner dynamiquement un rôle (*admin*, *user_control*, *user* ou *error*). Selon le rôle détecté, l'utilisateur est redirigé vers l'interface correspondante du Dashboard avec des droits d'accès spécifiques.

A.2 – Gestion de l'identification et des rôles utilisateurs

Comme montré dans la Figure 30, Ce bloc de code JavaScript est utilisé dans le nœud fonction (**Determination_Role**). Il permet de vérifier les identifiants saisis lors de la connexion, de distinguer l'administrateur des autres utilisateurs, puis d'assigner dynamiquement un rôle (*admin*, *user_control*, *user*, ou *error*) selon les droits définis dans les données internes.

```
1  var userdata = global.get("userdata") || []; // Récupère les utilisateurs stockés
2  var userRole = null; // Stocke le rôle de l'utilisateur
3  // Vérification si c'est l'Admin
4  if (msg.payload.id === "Admin" && msg.payload.pass === "safouan") {
5      msg.payload = { message: "Hi admin", role: "admin" };
6      msg.topic = "login Success";
7      return [msg, null, null, null];
8  } // Admin va vers la sortie 1
9  // Vérification pour les autres utilisateurs
10 var userFound = false; // Vérifier si l'utilisateur existe
11 var ec = false; // Contrôle activé ou pas
12 for (var i = 0; i < userdata.length; i++) {
13     if (userdata[i].id === msg.payload.id && userdata[i].pass === msg.payload.pass) {
14         ec = userdata[i].ec; // Récupère la valeur de contrôle
15         userFound = true;
16         break;
17     }
18 }
19 // Si l'utilisateur est trouvé
20 if (userFound) {
21     msg.payload = { message: "Hi " + msg.payload.id, role: ec ? "user_control" : "user" };
22     msg.topic = "login Success";
23     if (ec) {
24         return [null, msg, null, null]; // Vers la sortie 2 (user_control)
25     } else {
26         return [null, null, msg, null]; // Vers la sortie 3 (user)
27     }
28 }
29 // Si l'utilisateur n'existe pas
30 msg.payload = { message: "Password or User is wrong", role: "error" };
31 msg.topic = "login Failed";
32 return [null, null, null, msg]; // Vers la sortie 4 (erreur)
```

FIG. 30 : Code Node-RED pour la vérification des identifiants et l'attribution des rôles

Ce code illustré dans la Figure 31 définit les groupes visibles dans le Dashboard en fonction du rôle détecté. Il masque tous les éléments par défaut, sauf le groupe *Identification_Présentation*, puis affiche dynamiquement les vues autorisées pour chaque rôle grâce à une configuration centralisée.

```

1 // Liste complète des groupes
2 let allGroups = [
3   "Identification_Présentation", "Identification_Identification", "Fonctionnement_ChoixMode", "Fonctionnement_TypeProduit", "Fonctionnement_Commande",
4   "Station1_GuidageMoteur1", "Station1_GuidageMoteur2", "Station1_GuidageVérinsélecteur", "Station1_GuidageVérinsécurité", "Station2_FixationVérinsGauche",
5   "Station2_VérinsPositionnementVerticalLamme", "Station2_VérinPorteurLamme", "Station3_Evannes", "Station3_VérinspHNett", "Station3_NiveauProduit ", "Ala
6   "Récupération_QuantitéL", "Récupération_QuantitéCC", "Récupération_Comparaison", "Utilisateurs_Ajout", "Utilisateurs_ListeUtilisateurs",
7   "Utilisateurs_OrdreInformationsUtilisateur", "Utilisateurs_ModificationInformationsUtilisateur", "AffichagepH_pHActuel", "AffichagepH_pHBacs", "Déconnex
8 ];
9 // Si le rôle n'est pas défini, on affiche uniquement Identification_Présentation
10 if (!msg.payload || !msg.payload.role) {
11   let startupCommands = [];
12   // Cacher tous les groupes sauf Identification_Présentation
13   for (let group of allGroups) {
14     if (group !== "Identification_Présentation") {
15       startupCommands.push({ payload: { "group": { "hide": group } } });
16     }
17   }
18   // Afficher explicitement Identification_Présentation
19   startupCommands.push({ payload: { "group": { "show": "Identification_Présentation" } } });
20   return [startupCommands];
21 }
22 // Récupération du rôle (ou "error" par défaut)
23 let role = msg.payload.role || "error";
24 // Définir les groupes visibles pour chaque rôle
25 let groupsConfig = {
26   "admin": [
27     "Identification_Présentation", "Identification_Identification", "Fonctionnement_ChoixMode", "Fonctionnement_TypeProduit", "Fonctionnement_Commande",
28     "Station1_GuidageMoteur1", "Station1_GuidageMoteur2", "Station1_GuidageVérinsélecteur", "Station1_GuidageVérinsécurité", "Station2_FixationVérinsGau
29     "Station2_FixationVérinsPositionnementVerticalLamme", "Station2_FixationVérinPorteurLamme", "Station3_Evannes", "Station3_VérinspHNett", "Station3_Ni
30     "Récupération_QuantitéL", "Récupération_QuantitéCC", "Récupération_Comparaison", "Utilisateurs_Ajout", "Utilisateurs_ListeUtilisateurs",
31     "Utilisateurs_OrdreInformationsUtilisateur", "Utilisateurs_ModificationInformationsUtilisateur", "AffichagepH_pHActuel", "AffichagepH_pHBacs", "Déconn
32   ],

```

FIG. 31 : Affichage conditionnel des groupes de l'interface selon le rôle utilisateur

A.3 – Gestion des utilisateurs dans l'application Node-RED

La Figure 32 montre l'organisation des nœuds dans Node-RED pour la gestion des utilisateurs. Cette interface permet d'ajouter, de modifier, d'afficher ou de supprimer un utilisateur, ainsi que d'accéder à la page d'identification ou de se déconnecter. Les données saisies dans les formulaires sont traitées via des fonctions JavaScript avant d'être redirigées vers les vues appropriées.

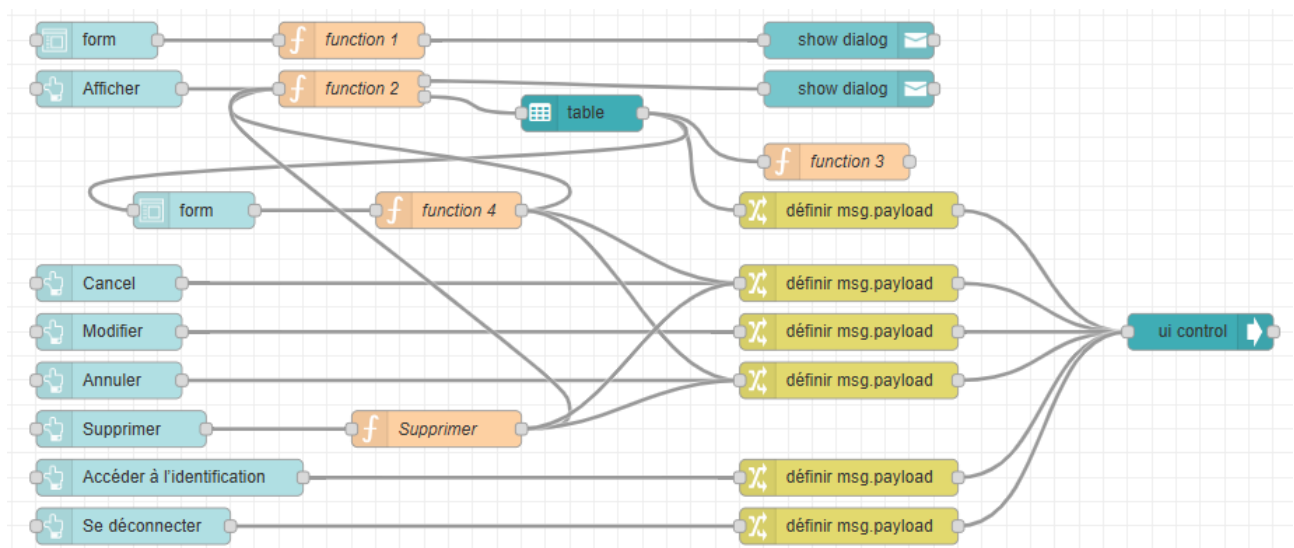


FIG. 32 : Schéma Node-RED pour l'ajout, la modification, l'affichage et la suppression d'utilisateurs

B. Installation et Configuration sur le Raspberry Pi

Cette annexe regroupe l'activation des interfaces nécessaires et l'installation des outils Node-RED et Cloudflared sur le Raspberry Pi.

B.1 - Activation des interfaces SSH et VNC

Dans cette étape, on active les interfaces SSH et VNC à partir de l'outil « Configuration du Raspberry Pi » afin de permettre l'accès distant et le contrôle graphique du Raspberry Pi.

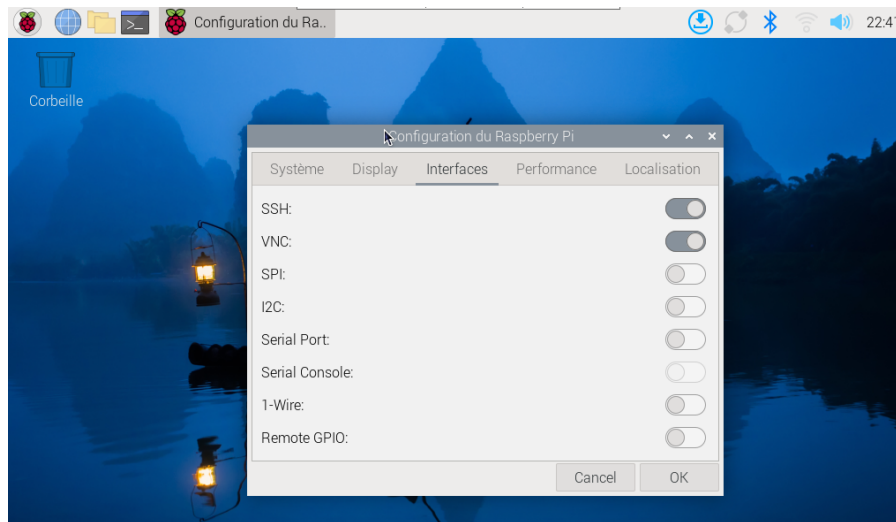


FIG. 33 : Activation de SSH et VNC via l'outil de configuration de Raspberry Pi

B.2 - Installation de Node-RED

Cette figure 34 montre le terminal au moment de l'installation de Node-RED.

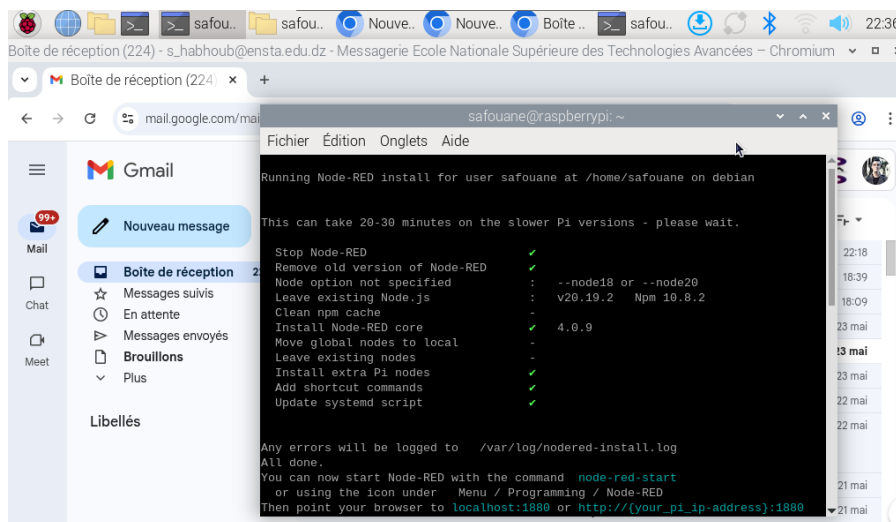


FIG. 34 : Commandes d'installation de Node-RED

B.3 - Installation de Cloudflared

Cette étape consiste à installer l'outil Cloudflared. Les commandes d'installation exécutées dans le terminal sont présentées dans la figure 35 suivante.

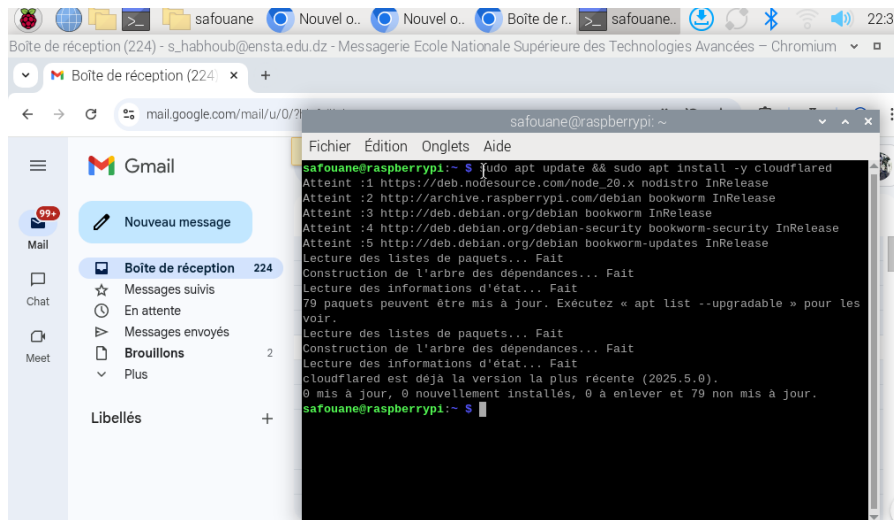


FIG. 35 : Commandes pour Cloudflared