

Cahier des charges ProSE - ST

1) Introduction	1
2) Linux	1
3) Android	3
4) Employés, rôle et calendrier	3

1) Introduction

STMicroelectronics (désigné par “**le client**”) souhaite, à terme, développer son propre système de **sonnette industrielle connectée**.

L'entreprise souhaite, par soucis de sécurité et de confidentialité, avoir un contrôle sur le software embarqué ainsi que la manipulation des données au sein de cette dernière.

Pour se faire, ST a mandaté le groupe ProSE A1 (désigné par “**l'équipe projet**”) pour concevoir un premier prototype de sa sonnette. Les besoins, contraintes et fonctionnalités souhaitées pour ce prototype sont répertoriés dans le présent document.

Le prototype de sonnette industrielle connectée se divise en deux sous-ensembles respectivement nommés “**Linux**” et “**Android**”.

Cette dernière doit permettre, grâce à un programme de reconnaissance facial, de reconnaître une personne qui se présente devant la caméra embarquée, et, le cas échéant, simuler l'ouverture de la porte par la commande d'un moteur.

Pour les personnes non reconnues, la sonnette doit notifier une application Android et permettre l'affichage du flux vidéo. L'application Android permet également de configurer la sonnette.

Ce document n'est pas contractuel. Il n'engage pas l'équipe projet dans la réalisation de toutes les attentes formulées par le client. Il est le résumé, à titre indicatif, des demandes formulées lors de la rencontre client du jeudi 9 février. Le détail de la réalisation concrète de l'équipe projet sera incluse dans le futur contrat signé entre les deux parties.

2) Linux

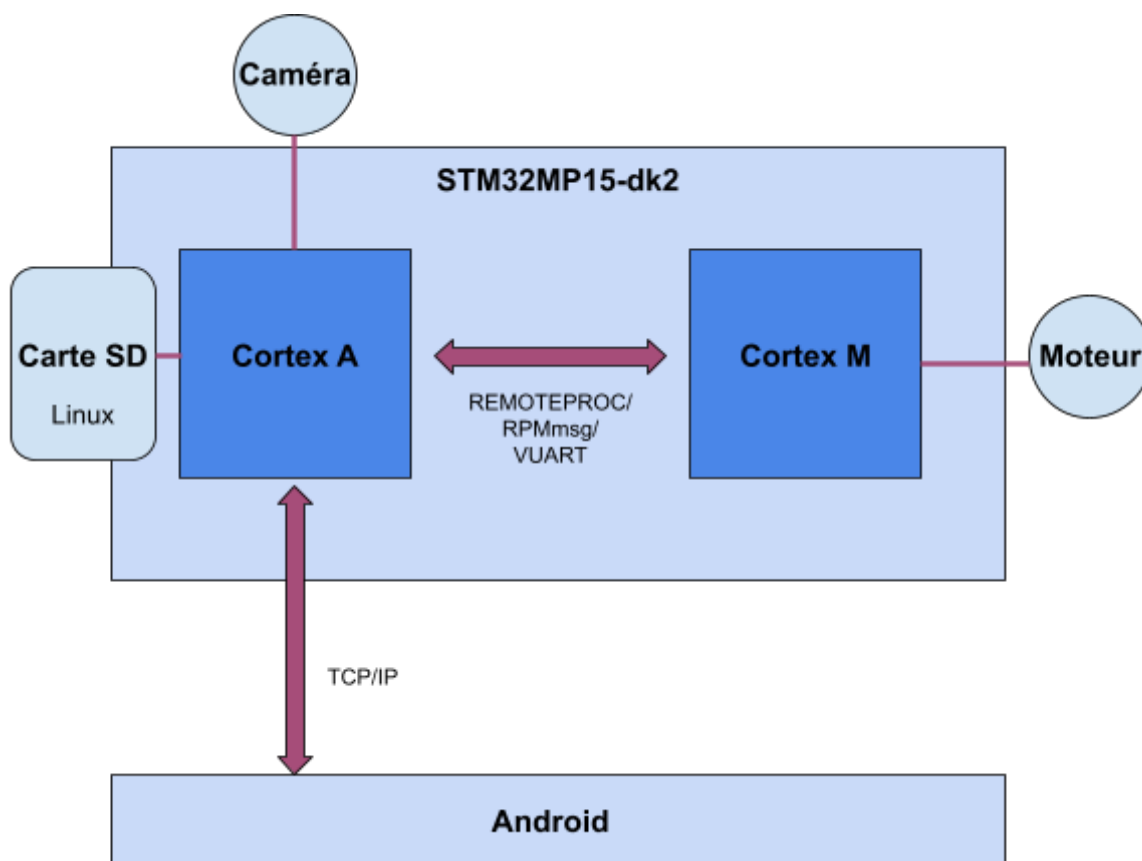
La partie Linux de la sonnette connectée est en charge des aspects suivant :

- **Commande du moteur** (pour simuler l'ouverture et la fermeture d'une porte)
- **Serveur** pour la communication avec l'application Android
- **Streaming vidéo** via une Webcam sur l'application Android et sur l'écran embarqué de la sonnette
- **Reconnaissance faciale** via un programme d'Intelligence Artificielle des visages présents dans la base de données
- Dans un second temps, **encryptage et sauvegarde des photos** des visages sauvegardés
- Notification de l'application Android

Ce sous-ensemble se compose des éléments hardware suivant :

- Une **STM32MP15-dk2** (embarquant le kernel linux), imposé par le client
- Une webcam pour simuler une future caméra embarquée, à la discrétion de l'équipe projet
- Un moteur pour simuler l'ouverture et fermeture de la porte, à la discrétion de l'équipe projet

La STM32MP-15-dk2 se divise elle même en deux sous-ensembles :



Cortex A :

- Embarque le kernel Linux (stocké sur la carte SD)
- En charge de :
 - L'IA
 - Le traitement du flux vidéo
 - La communication avec l'application Android
- Version de linux à utiliser :
 - OpenSTLinux (OSTL) version 4.1 (Kernel linux : 5.15) (recommandé)
 - OpenSTLinux (OSTL) version 3.1 (Kernel linux : 5.10)

Cortex M :

- = MCU, architecture ARM
- En charge de :
 - Contrôle du moteur
- Version de l'environnement à utiliser :
 - Cube FW Package version 1.6
 - CubeIDE (dernière version)

La communication entre le Cortex A et le Cortex M peut se faire avec l'un des protocoles suivant :

- REMOTEPROC
- RPMmsg
- UART

Le client souhaite que la priorité soit donnée au développement de l'IA embarquée. Cette dernière devra :

- Être pré-entraînée
- Reconnaître un visage sur la base d'une image seulement enregistrée dans la base de donnée

Le client laisse à la discrétion de l'équipe projet le choix des bibliothèques utilisées.

3) Android

La sonnette pourra communiquer avec une application Android. L'application permet de :

- Recevoir le **flux vidéo** de la webcam
- "**Ouvrir la porte**" depuis l'application (activation/désactivation du moteur)
- Visualiser les horaires d'entrée des employés sur un **calendrier** d'une semaine typique de 7 jours
- Visualiser la **liste des employés** avec leur nom, rôle et photo
- Pouvoir **ajouter, modifier et supprimer** des employés
- La modification de la liste des employés implique une **connexion administrateur**.
- Pouvoir créer un **rôle "privilège"** avec des accès définis manuellement

L'application sera rendue sous le format .apk. Le client n'impose pas si l'application doit être développée pour une tablette ou un téléphone. Le **format portrait** est cependant recommandé. Le client laisse à la discrétion de l'équipe projet le type de connexion entre le serveur Linux et l'application Android.

Employés, rôle et calendrier

La sonnette pourra gérer différents types d'accès en fonction du rôle des employés.

- Un employé jour : 8-20h / lundi au vendredi
- Employé matin : 3-13h / lundi au vendredi
- Employé soir : 13h-23h / lundi au vendredi
- Employé sécu : accès illimité
- Accès privilège : définition manuelle dans l'application