



LETTER

0.0.1 *Système de Sécurité Anti-Intrusion pour les Zones Rurales*

Ben yaagoub Wided and BenAli Mayssa,

Department of Industrial computing, ISIMG, Gabes University, Tunisia

*Corresponding author, Email:

wided.benyaagoub@isimg.tn

; maysa.benali@isimg.tn

Abstract

Dans un contexte où la sécurité devient un enjeu crucial, notamment dans les zones isolées, la surveillance intelligente s'impose comme une solution innovante et accessible. Ce projet vise à concevoir un système autonome capable de détecter automatiquement toute présence humaine à l'aide d'un capteur de mouvement (PIR), de capturer une image grâce à une caméra connectée à un Raspberry Pi, et d'envoyer une notification instantanée via le protocole MQTT.

Ce dispositif repose sur des technologies embarquées et de communication légères, telles que Python pour la programmation, MQTT pour la transmission des alertes, et la bibliothèque Picamera2 pour la gestion de la caméra. Le système est pensé pour être simple, efficace et facilement déployable dans des environnements à faible infrastructure.

Le projet met en œuvre une chaîne complète allant de la détection du mouvement jusqu'à l'envoi de l'image capturée, avec des tests concluants sur la rapidité et la fiabilité du dispositif. Cet article détaille l'ensemble du processus de conception, de développement et d'évaluation, tout en ouvrant la voie à des améliorations futures telles que l'intégration de la reconnaissance d'objets ou l'ajout d'un module d'alimentation solaire.

Keywords: MQTT; Picamera2; PIR; Application Mobile; Python

Abbreviations: PIR:Capteur de mouvement, IOT: Internet des Objets (Internet of Things), MQTT:MQ Telemetry Transport ,LoRaWAN:long-range wide-area network

1. Introduction générale

Dans un monde en constante évolution, la sécurité des personnes et des biens est devenue une préoccupation majeure, tant dans les zones urbaines que rurales. L'augmentation des actes de cambriolage, d'intrusion ou encore les risques liés à l'isolement rendent indispensable la mise en place de dispositifs de surveillance et d'alerte fiables. Que ce soit dans les maisons individuelles, les bureaux,

les exploitations agricoles ou les établissements publics, le besoin d'un système de sécurité efficace et accessible se fait de plus en plus sentir.

Cependant, les systèmes de sécurité traditionnels présentent plusieurs limites. Leur coût d'acquisition et d'installation peut être prohibitif pour de nombreux foyers ou petites structures. De plus, la dépendance à une connexion Internet stable, la complexité de configuration ou encore la difficulté de maintenance les rendent peu adaptés à certains environnements, notamment les zones reculées ou faiblement équipées.

Face à ces constats, il devient essentiel d'imaginer des solutions alternatives, plus simples, plus économiques et surtout plus accessibles. C'est dans ce cadre que s'inscrit notre projet : la conception d'un **système de sécurité intelligent basé sur un réseau de capteurs d'alerte**, destiné à renforcer la protection des zones isolées et à offrir une réponse technologique concrète à une problématique bien réelle.

1.1 Importance de la sécurité dans différents environnements

La sécurité représente un besoin fondamental dans toute société, quel que soit le type d'environnement. Elle garantit la protection des personnes, des biens matériels et de l'information. L'évolution des technologies a permis l'apparition de solutions de plus en plus intelligentes pour répondre aux besoins croissants en matière de sûreté, surtout dans un monde confronté à des risques variés : cambriolages, intrusions, agressions, accidents domestiques ou encore situations d'urgence médicale.

Maisons individuelles

Dans les habitations, la sécurité vise avant tout à protéger les occupants contre les intrusions ou les incidents domestiques. De nombreux foyers cherchent à sécuriser leur domicile à l'aide de caméras, d'alarmes ou de capteurs de mouvement. Cependant, ces dispositifs ne sont pas toujours accessibles, notamment dans les régions où la technologie est moins développée ou les revenus sont modestes. La prévention des cambriolages, la détection de mouvements suspects et la surveillance à distance sont donc des enjeux majeurs.

Bureaux et entreprises

Dans les locaux professionnels, la sécurité joue un rôle stratégique. Elle concerne à la fois la protection des employés, la prévention des vols, la sécurité des données, et la surveillance des accès. Les entreprises investissent dans des systèmes sophistiqués pour contrôler les allées et venues, assurer la surveillance vidéo continue, ou déclencher des alertes automatiques. L'enjeu ici est également la continuité de l'activité, qui peut être mise en danger en cas de défaillance de sécurité.

Zones rurales ou isolées

Les zones reculées ou faiblement peuplées sont souvent mal desservies en matière de sécurité. Le manque d'infrastructures, l'absence de couverture réseau, ou l'éloignement des services de secours rendent ces régions vulnérables. Dans ces contextes, la surveillance et la détection de mouvements deviennent cruciales, particulièrement pour les travailleurs isolés, les fermes ou les habitations situées loin des centres urbains.

Établissements publics (écoles, hôpitaux...)

Dans les institutions publiques, la sécurité est une priorité constante. Elle doit assurer la protection de populations parfois vulnérables, comme les enfants ou les patients. Des systèmes de surveillance sont installés pour prévenir les actes malveillants, gérer les flux de personnes et garantir une réponse rapide en cas d'incident. La sécurité dans ces lieux contribue directement à la sérénité des usagers et du personnel.

\

1.2 Solution Proposée

Face aux limites des systèmes de sécurité traditionnels, il est essentiel de concevoir des solutions alternatives, à la fois efficaces, économiques et adaptées aux différents contextes, notamment les zones

isolées ou les foyers à faible revenu. Dans cette optique, notre projet propose le développement d'un **système de sécurité intelligent basé sur un réseau de capteurs connectés**, capable de détecter les mouvements suspects et de notifier instantanément les utilisateurs ou les autorités, même en l'absence de connexion Internet stable.

Ce système repose sur l'intégration de **capteurs PIR (infrarouges passifs)** pour la détection de mouvements, d'une **caméra** pour capturer les preuves visuelles en temps réel, et d'une **carte Raspberry Pi** comme unité centrale de traitement. La transmission des alertes se fait à l'aide du **protocole MQTT**, qui permet une communication rapide et légère entre les différents modules du système. Pour les régions sans réseau Internet, une solution via **LoRaWAN** peut être envisagée, permettant une transmission longue distance à faible consommation.

L'ensemble du dispositif est conçu pour être **économique, autonome et facilement déployable** dans divers environnements. Il peut être installé dans des maisons individuelles, des exploitations agricoles, des bâtiments publics ou encore auprès de travailleurs isolés. L'objectif est de démocratiser l'accès à la sécurité en proposant une alternative open-source, personnalisable et modulable selon les besoins spécifiques de chaque utilisateur.

1.3 Organisation du manuscrit

Le reste de ce manuscrit est organisé comme suit:

- **La deuxième partie** aborde la conception du système de sécurité proposé. Elle détaille les choix technologiques, l'architecture matérielle et logicielle du prototype, ainsi que les méthodes de communication utilisées (capteur, caméra, protocole MQTT...).
- **La troisième partie** est dédiée à la mise en œuvre et à l'évaluation du système. Elle présente les différentes étapes de réalisation, les tests effectués en situation réelle, les résultats obtenus.

2. Description du système de surveillance IoT

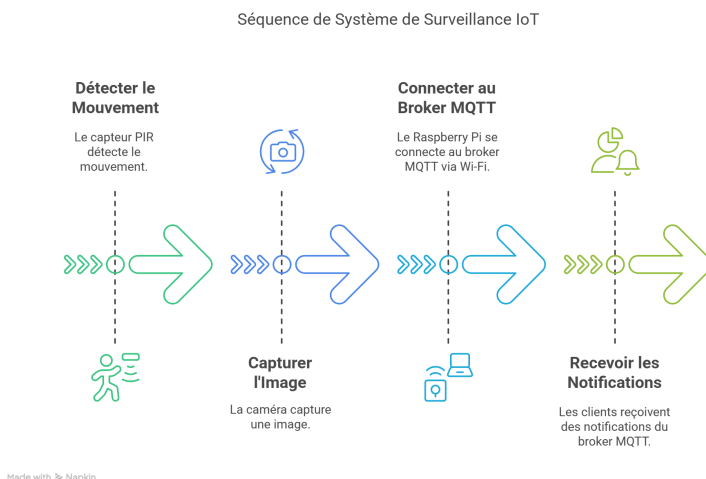


Figure 1. Architecture générale du système de surveillance IoT

Le schéma ci-dessus illustre l'enchaînement des différentes étapes fonctionnelles d'un système de **surveillance intelligent basé sur l'IoT**. L'objectif est de détecter des mouvements suspects, capturer des images, et transmettre des alertes à distance en temps réel, en s'appuyant sur une communication entre plusieurs composants matériels et logiciels. L'ensemble du système repose sur la combinaison d'un capteur de mouvement, d'une caméra, d'un microcontrôleur (Raspberry Pi), et d'un serveur MQTT assurant la diffusion des notifications. 1.

Le schéma présenté illustre les différentes étapes du **fonctionnement d'un système de surveillance intelligent basé sur l'IoT**. Ce système repose sur une **interaction fluide entre plusieurs composants matériels et logiciels** : capteur PIR, caméra, Raspberry Pi, broker MQTT, et client de réception.

2.1 Détection du Mouvement

La première phase du système repose sur la **détection d'un mouvement physique** dans la zone surveillée. Pour cela, on utilise un **capteur PIR (Passive Infrared Sensor)**. Ce capteur est capable de repérer les changements dans le rayonnement infrarouge, émis naturellement par les corps chauds comme les humains ou les animaux. Il agit donc comme un **déclencheur d'événements**, envoyant un signal électrique vers le Raspberry Pi dès qu'un mouvement est détecté. Cette approche permet un **monitoring permanent**, même dans des conditions de luminosité faible, sans nécessiter de vidéo continue (ce qui économise de l'énergie et du traitement). Grâce à la simplicité de ce capteur, le système reste à la fois économique, fiable et facile à installer dans différents environnements.

2.2 Capture de l'Image

Dès qu'un mouvement est détecté, le Raspberry Pi active une **caméra connectée** (dans notre cas, une Picamera2) pour **capturer une image instantanée** de la scène. Cette image joue un rôle central dans le système, car elle permet d'avoir une **preuve visuelle immédiate** de l'événement détecté. La capture est réalisée à travers un script Python qui configure la caméra en mode image fixe (still image), assure la prise de vue, puis enregistre l'image dans un répertoire spécifique du Raspberry Pi.

2.3 Connexion au Broker MQTT

Une fois l'image capturée, le système entre dans la phase de **communication réseau**. Le Raspberry Pi se connecte au **broker MQTT**, un serveur léger qui centralise les messages entre les objets connectés. MQTT (Message Queuing Telemetry Transport) est un protocole de messagerie conçu pour les **environnements à faible bande passante**, ce qui le rend parfait pour l'IoT. À travers cette étape, le Raspberry Pi **publie un message d'alerte** sur un canal spécifique (topic), comme par exemple **camera/topic**. Ce message peut inclure une indication de l'événement (« Mouvement détecté ! ») et le nom de l'image enregistrée. Le broker MQTT agit alors comme **relais**, diffusant le message à tous les clients abonnés. Cette architecture **découplée** permet une communication rapide, fiable et scalable entre les composants.

2.4 Réception des Notifications

Enfin, les **clients MQTT** qui se sont abonnés au topic reçoivent **instantanément les notifications d'alerte**. Ces clients peuvent être variés : une application mobile (Android), une interface web de surveillance, ou même un système de contrôle central. Lorsqu'un message est reçu, il peut être affiché sous forme de **notification visuelle ou sonore**, ou déclencher une autre action (ex. allumage d'une alarme, enregistrement vidéo, etc.). Cette étape permet à l'utilisateur final d'être **immédiatement informé d'une activité suspecte**, sans devoir surveiller en permanence. Ce fonctionnement asynchrone assure une **réactivité optimale du système**, tout en restant léger côté ressources.

2.5 Synthèse

Ces quatre étapes travaillent ensemble pour former un système **cohérent, autonome et réactif**, capable de transformer un simple événement physique (comme un mouvement) en **alerte distante intelligente**. L'avantage principal réside dans la simplicité des composants et dans la fluidité de la communication, qui peut s'adapter aussi bien à un usage personnel (maison, ferme) qu'à un usage collectif (écoles, hôpitaux, entrepôts). De plus, grâce à la modularité de l'architecture, il est possible de **faire évoluer le système** (ajouter la détection de son, l'analyse vidéo par IA, l'intégration à une application mobile, etc.) sans devoir le reconstruire entièrement.

2.6 Analyse fonctionnelle

Afin de mieux représenter le concept et les caractéristiques de l'application, nous mettrons en œuvre un diagramme de cas d'utilisation présenté dans la figure 2.

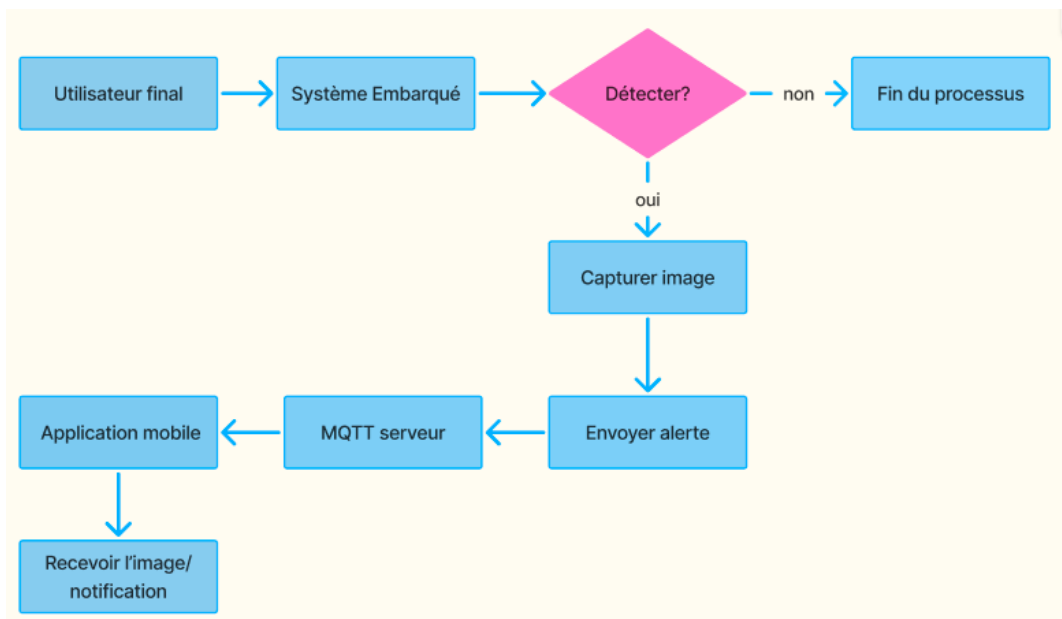


Figure 2. Diagramme de cas d'utilisation

Cette figure illustre le diagramme de cas d'utilisation du fonctionnement global d'un système de surveillance basé sur un système embarqué, tel qu'un Raspberry Pi, équipé d'un capteur de mouvement (PIR) et d'une caméra. Le processus commence avec l'utilisateur final qui interagit ou déclenche le système embarqué. Ce dernier surveille continuellement l'environnement pour détecter une présence ou un mouvement. Lorsqu'aucune détection n'est signalée, le processus se termine immédiatement. En revanche, si un mouvement est détecté, le système passe à l'étape suivante qui consiste à capturer une image à l'aide de la caméra connectée.

Une fois l'image prise, une alerte est générée et transmise via un serveur MQTT. Ce serveur agit comme un intermédiaire de communication, assurant le transfert rapide et fiable de l'image vers les clients abonnés. Parmi ces clients figure une application mobile, qui reçoit les notifications envoyées par le système. L'utilisateur final peut alors consulter l'image capturée et être informé en temps réel de tout mouvement suspect détecté. Ce flux d'information permet une surveillance efficace à distance, combinant détection automatique, capture d'image, transmission et notification via des technologies embarquées et mobiles.

3. Implémentation du système de surveillance

Dans cette section, nous allons présenter les étapes et les détails d'implémentation des différentes sous parties de notre système surveillance.

3.1 Système de surveillance

La mise en place du prototype du système de surveillance repose sur deux volets:

- Un volet matériel: qui regroupe l'ensemble des composants électroniques nécessaires au fonctionnement du système. Cela inclut notamment le capteur de mouvement PIR, la caméra de surveillance, le microcontrôleur Raspberry Pi, ainsi que les modules de communication permettant d'assurer la connectivité du dispositif.
- un volet logiciel: qui concerne le développement de l'architecture logicielle permettant d'assurer la détection des mouvements, la capture d'images et l'envoi des alertes via le protocole MQTT. Ce volet intègre également une interface simple destinée à faciliter l'interaction entre le système et l'utilisateur, en lui permettant de recevoir les notifications de manière fluide et sécurisée.

3.2 Les composants matériels

Afin de réaliser notre prototype matériel, nous avons utilisé les composants:

3.2.1 Le Raspberry Pi 4 Model B+

Le **Raspberry Pi 4 Model B+** est un mini-ordinateur puissant et polyvalent, parfaitement adapté aux projets embarqués comme notre système de sécurité. Il est doté d'un processeur **Broadcom BCM2711**, un **Quad-core Cortex-A72 (ARM v8)** cadencé à **1,5 GHz**, offrant une puissance de calcul suffisante pour exécuter des scripts Python, traiter des images ou même faire tourner des algorithmes d'intelligence artificielle légers. Concernant la mémoire, notre prototype utilise la version équipée de **4 Go de RAM LPDDR4**, un bon compromis entre performance et gestion efficace du multitâche. Côté connectivité, le Raspberry Pi 4 intègre une **connexion Wi-Fi 802.11ac double bande** et le **Bluetooth 5.0**, permettant une liaison sans fil avec divers périphériques, ainsi qu'un **port Ethernet Gigabit** pour une transmission de données rapide et stable, particulièrement utile dans un contexte de surveillance en temps réel. Pour les besoins de communication avec les périphériques externes, il propose **deux ports USB 3.0** et **deux ports USB 2.0**, permettant de connecter caméras, claviers, clés USB ou autres modules. L'affichage est assuré via **deux ports micro-HDMI** compatibles 4K, et une sortie **jack 3,5 mm audio/vidéo** est également disponible. Le stockage du système se fait via une **carte microSD**, avec la possibilité d'extension par **SSD USB 3.0**. Un autre avantage essentiel réside dans ses **40 broches GPIO programmables**, qui facilitent l'intégration de capteurs (comme le PIR), de LED, de relais ou de tout autre module électronique. Enfin, l'alimentation s'effectue via un port **USB-C 5V/3A**, ce qui est suffisant pour alimenter la carte et les composants annexes du projet.

Dans le cadre de notre système de sécurité intelligent, le **Raspberry Pi 4 Model B+** joue un rôle central en tant qu'unité de traitement principale. Il est chargé de recevoir et d'interpréter les signaux provenant du capteur de mouvement PIR afin de détecter toute activité suspecte dans la zone surveillée. Lorsqu'un mouvement est détecté, le Raspberry Pi active automatiquement la caméra connectée pour capturer une image de la scène. Cette image est ensuite enregistrée localement avec un horodatage, puis une notification est générée et transmise à un serveur distant via le protocole MQTT. Grâce à sa puissance de calcul et à ses interfaces de communication, le Raspberry Pi permet également l'exécution fluide de scripts Python ainsi que l'intégration d'une interface graphique simple pour une interaction locale avec le système. Sa polyvalence, son faible coût et sa compatibilité avec de nombreux capteurs en font une plateforme idéale pour un système embarqué de sécurité accessible et efficace.

3.2.2 Capteur PIR

Le capteur PIR (Passive Infrared Sensor) est un composant fondamental du système de sécurité, utilisé pour détecter les mouvements dans une zone définie. Fonctionnant sur une tension d'alimentation de 3.3V à 5V, ce capteur se distingue par sa faible consommation d'énergie et sa réactivité. Il est capable de repérer des mouvements humains dans un champ de vision d'environ 120° et sur une portée allant de 3 à 7 mètres. Sa sortie est numérique, ce qui signifie qu'il transmet simplement un signal haut ou bas selon qu'un mouvement est détecté ou non. Dans le cadre de notre projet, le capteur PIR est utilisé pour surveiller l'environnement. Lorsqu'un mouvement est détecté, il déclenche automatiquement la caméra connectée au Raspberry Pi afin de capturer une image de la scène. Cette image est ensuite transmise via le protocole MQTT pour alerter le serveur ou un utilisateur distant. Grâce à sa simplicité, sa fiabilité et sa capacité à fonctionner dans des zones sans surveillance constante, le capteur PIR constitue un élément clé pour assurer une sécurité intelligente et autonome.

3.2.3 Caméra Pi V2

La caméra Raspberry Pi V2 (officiellement appelée **Camera Module v2**) est un module de capture d'image haute qualité spécialement conçu pour être utilisé avec les cartes Raspberry Pi. Elle est équipée d'un capteur Sony IMX219 de 8 mégapixels, capable de capturer des images en résolution 3280 x 2464 et de filmer des vidéos en 1080p à 30 images par seconde. Compacte et légère, elle se connecte facilement au port CSI (Camera Serial Interface) du Raspberry Pi, offrant une communication rapide et efficace. Dans notre projet de sécurité basé sur la détection de mouvement, la caméra joue un rôle essentiel : elle est déclenchée automatiquement par le capteur PIR lors de la détection d'une présence. Elle capture alors une image de la scène ou de l'intrus, qui est ensuite transmise via le protocole MQTT au serveur ou à une interface de surveillance distante. Grâce à sa qualité d'image élevée et à sa compatibilité parfaite avec le Raspberry Pi, la caméra Pi V2 permet une surveillance visuelle fiable et instantanée, renforçant l'efficacité du système de sécurité mis en place.

3.3 Architecture matérielle

Dans la figure 3, nous pouvons observer la représentation graphique de la connexion entre les différents composants.

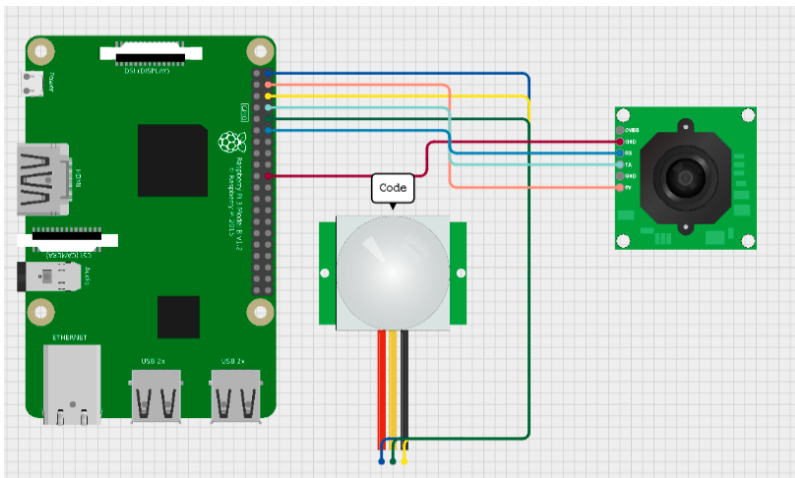


Figure 3. Connexion entre les composants

3.4 Les composants logiciels

Le système utilise le protocole **MQTT (Message Queuing Telemetry Transport)**, connu pour sa légèreté et son efficacité dans les environnements IoT. Il suit le modèle **publish/subscribe** dans lequel le Raspberry Pi joue le rôle de *publisher* (émetteur) tandis qu'un terminal (application mobile) agit comme *subscriber* (récepteur). Le **broker Mosquitto** est utilisé comme serveur MQTT pour gérer les échanges entre les différents clients présentée dans la figure 4 :

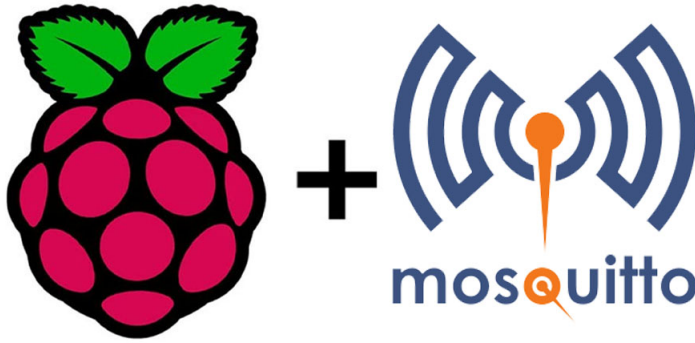


Figure 4. MQTT : Protocole de communication

Dans la figure 5 présente les différentes bibliothèques utilisées dans l'environnement du Raspberry Pi, à savoir :

```

rasbyp@rasberrypi: ~
GNU nano 7.2                               finaaal.py
import RPi.GPIO as GPIO
import time
from picamera2 import Picamera2
from datetime import datetime
import io
import base64
import paho.mqtt.client as mqtt
from PIL import Image # Assurez-vous que cette bibliothèque est installée

```

Figure 5. Les bibliothèques utilisées

- **RPi.GPIO** est spécifiquement conçue pour permettre au Raspberry Pi d'interagir avec ses broches GPIO (General Purpose Input/Output). Elle est utilisée pour configurer et lire l'état de composants électroniques connectés, comme des capteurs de mouvement, des relais, des LEDs ou des boutons poussoirs. Dans un projet de sécurité ou de surveillance, elle permet par exemple de détecter lorsqu'un capteur PIR identifie une présence et de déclencher des actions en conséquence, comme l'activation d'une caméra ou l'envoi d'une alerte.
- **picamera2** est l'outil moderne pour utiliser la caméra du Raspberry Pi avec la pile **libcamera**. Elle permet de capturer des images et des vidéos, de configurer la résolution, le format ou le

mode de prévisualisation, et de gérer plusieurs flux. Dans ton projet, Picamera2 est utilisée pour prendre automatiquement une photo lorsqu'un mouvement est détecté, tout en offrant la possibilité d'afficher la vidéo en direct dans une interface ou sur un navigateur via Flask.

- **base64** est utilisée pour encoder des fichiers binaires (comme des images) en texte. Cette opération est particulièrement utile pour l'envoi d'images via des protocoles comme MQTT, qui sont basés sur des messages textuels. Grâce à cette conversion, une image capturée peut être envoyée sous forme de chaîne de caractères à un serveur ou une application mobile, puis reconstruite à l'arrivée.
- **paho.mqtt.client** est une bibliothèque de communication basée sur le protocole MQTT, un protocole léger et rapide très utilisé dans les applications IoT. Elle permet à ton Raspberry Pi d'agir comme **client MQTT** : il peut se connecter à un broker, publier des messages (comme une image encodée ou une alerte), ou en recevoir. Dans ton projet, c'est grâce à cette bibliothèque que le système embarqué peut envoyer des notifications en temps réel à une application mobile ou un serveur distant.
- **PIL (Pillow)** permet de manipuler des images en Python. Elle est utile pour ouvrir, convertir, redimensionner ou sauvegarder les images. Dans ton cas, elle peut être utilisée pour traiter une image capturée par la caméra, l'optimiser pour l'envoi, ou effectuer des modifications avant son affichage ou sa transmission.

Le développement logiciel a été réalisé en **Python**, en raison de sa simplicité et de la richesse de ses bibliothèques.

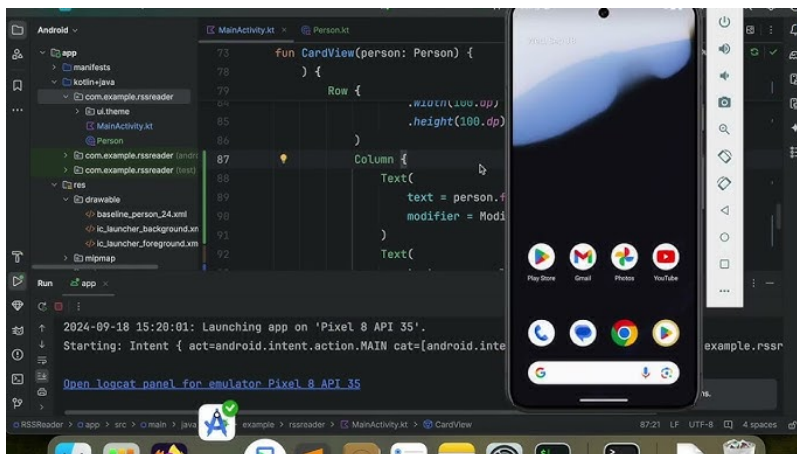


Figure 6. Android studio

La Figure 6 illustre l'interface principale de l'application mobile conçue à l'aide d'Android Studio. Cette application joue un rôle crucial dans la réception des notifications envoyées via MQTT, permettant à l'utilisateur de visualiser instantanément les images capturées par le système. Elle a été développée de manière à garantir une expérience utilisateur fluide et intuitive, tout en assurant une intégration efficace avec les services IoT embarqués sur le Raspberry Pi.

4. Expérimentation et résultats

La figure 7 représente le prototype du système en fonctionnement, illustrant la capture automatique d'une image suite à la détection d'un mouvement.

L'application mobile développée dans le cadre de ce projet permet à l'utilisateur de recevoir des alertes de sécurité en temps réel. Elle affiche automatiquement l'image capturée suite à la détection d'un

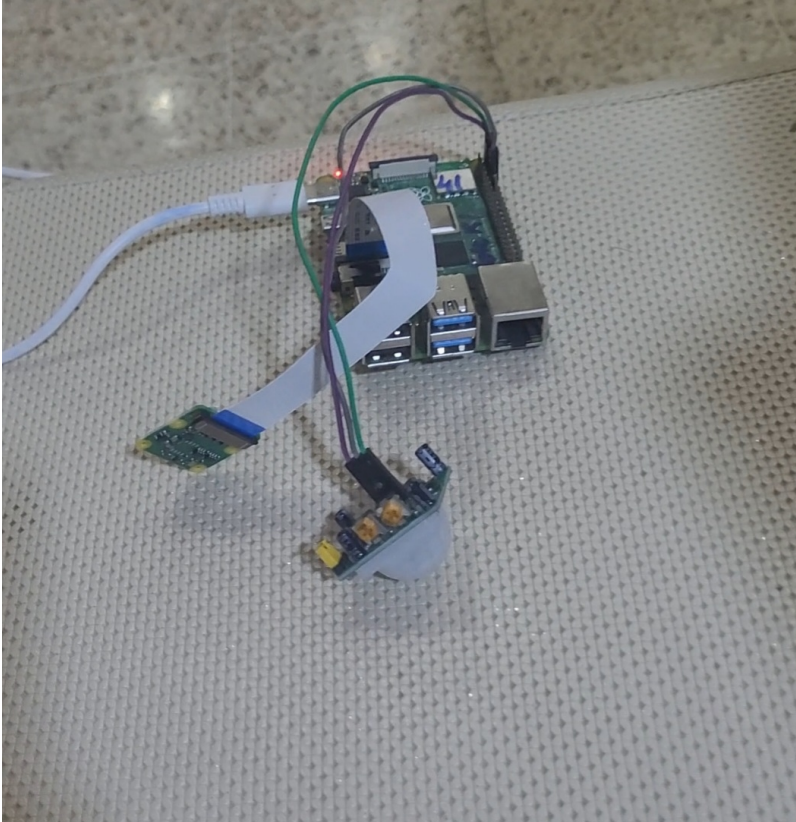


Figure 7. Le prototype du système

mouvement, facilitant ainsi la vérification rapide d'une intrusion potentielle. La figure 8 ci-dessous illustre l'écran principal de l'application nommé *PiSecure*.

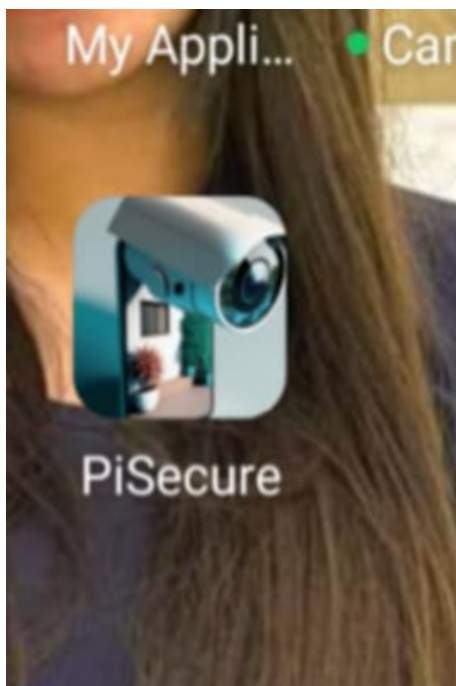


Figure 8. Interface principale de l'application mobile PiSecure

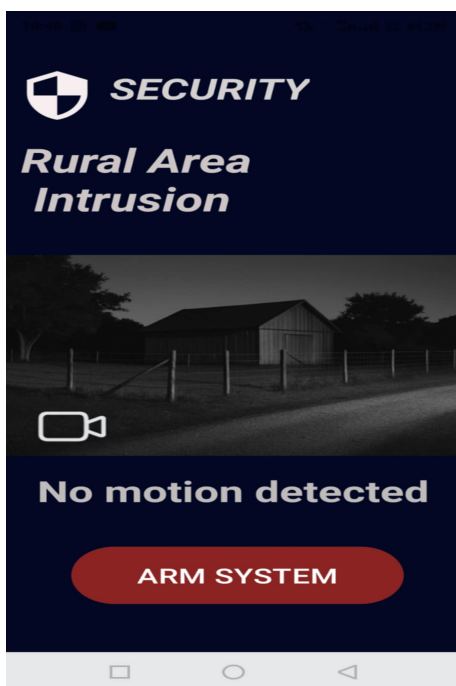


Figure 9. Interface de l'application

L'interface d'accueil qui représente dans la figure ci-dessous permet à l'utilisateur de se connecter

à son compte en saisissant son adresse e-mail et son identifiant. Elle propose également une option pour réinitialiser le mot de passe ainsi qu'un lien d'inscription pour les nouveaux utilisateurs.

Figure 10. Interface de connexion

L'interface dans la figure 7 affiche l'état des différents points d'accès d'une maison : hall d'entrée, porte et cuisine. Elle informe l'utilisateur en temps réel sur la sécurité du domicile, avec des icônes claires pour indiquer l'état (fermé, détecté, etc.).



Figure 11. Interface d'accueil

Comme le montre la figure 12, cette interface présente le journal des détections effectuées par le système. Elle permet de visualiser les événements de sécurité détectés (comme les mouvements ou l'état des portes) et de suivre les notifications envoyées via le protocole MQTT.

```

🔔 Attente de détection par le capteur PIR...
🔔 Mouvement détecté ! Démarrage de la caméra...
📷 Image enregistrée : /home/rasbypi/image_20250409_212155.jpg
📡 Notification envoyée via MQTT !
🔔 Mouvement détecté ! Démarrage de la caméra...
📷 Image enregistrée : /home/rasbypi/image_20250409_212201.jpg
📡 Notification envoyée via MQTT !
🔔 Mouvement détecté ! Démarrage de la caméra...
📷 Image enregistrée : /home/rasbypi/image_20250409_212206.jpg
📡 Notification envoyée via MQTT !
🔔 Mouvement détecté ! Démarrage de la caméra...
📷 Image enregistrée : /home/rasbypi/image_20250409_212212.jpg
📡 Notification envoyée via MQTT !
🔔 Mouvement détecté ! Démarrage de la caméra...
📷 Image enregistrée : /home/rasbypi/image_20250409_212217.jpg
📡 Notification envoyée via MQTT !
🔔 Mouvement détecté ! Démarrage de la caméra...
📷 Image enregistrée : /home/rasbypi/image_20250409_212222.jpg

```

Figure 12. Journal de détection et d'envoi via MQTT

Cette interface montre la réception d'une notification instantanée suite à la détection d'un mouvement. L'alerte comprend une image capturée automatiquement par la caméra au moment de la détection, permettant à l'utilisateur de visualiser immédiatement la situation et de réagir rapidement en cas d'intrusion.

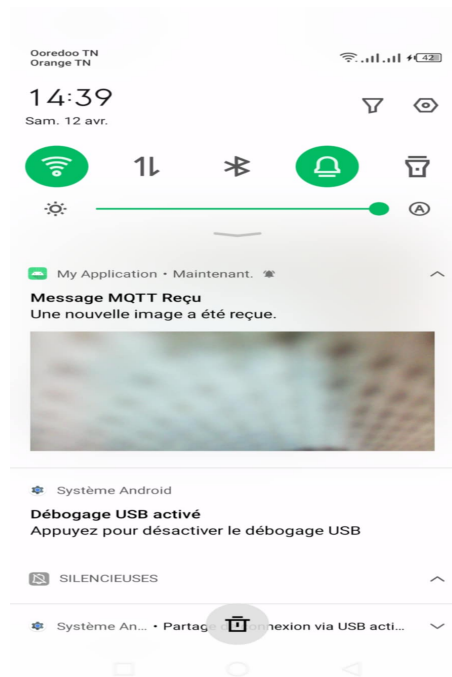


Figure 13. Interface de notification

5. Conclusion

Ce mini-projet a permis de concevoir et de mettre en œuvre un système de sécurité intelligent, adapté aux zones isolées ou faiblement surveillées. Grâce à l'utilisation de capteurs PIR, d'une caméra embarquée et du protocole MQTT, le système est capable de détecter un mouvement, de capturer automatiquement une image et de la transmettre instantanément à un terminal abonné, offrant ainsi une solution simple, réactive et peu coûteuse.

L'intégration d'une interface mobile intuitive renforce l'accessibilité du système en permettant à l'utilisateur de surveiller son domicile à distance. Ce projet a également été une opportunité concrète pour exploiter des technologies actuelles comme l'IoT, la programmation Python, et la communication légère via MQTT.

En somme, cette réalisation démontre qu'il est possible de concevoir un dispositif de sécurité autonome, efficace et évolutif, pouvant être enrichi par des perspectives futures telles que la reconnaissance faciale, l'ajout de stockage cloud ou encore une alimentation par panneau solaire.