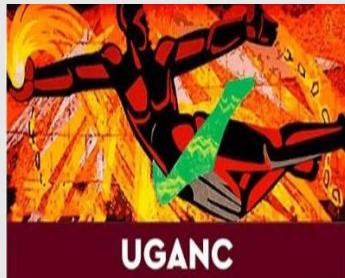


# REPUBLIQUE DE GUINEE

*Travail-Justice-Solidarité*



**UNIVERSITE GAMAL ABDEL NASSER DE CONAKRY**

-----  
**CENTRE INFORMATIQUE  
(NTIC)**

Projet de Fin d'Etude

**Système de Détection et d'alerte Précoce en utilisant les Réseaux de Capteurs distribués et IA**

Spécialité : NTIC (Nouvelle Technologie de l'Information et de la Communication)

Entreprise : ENTACIG (Espace Numérique de Technologie et d'Alerte du Centre Informatique de Gamal

Présenté Par :

1/ BERETE Moussa

Licence III (NTIC)

2/ SAGNO Beko

Licence III (NTIC)

## Résumé

---

Ces dernières décennies le monde a connu un développement indéniable dans le secteur de technologie en générale et dans le bien être du personnel en particulier, où l'apparition de la domotique a rendu la vie quotidienne beaucoup plus simple.

Notre travail s'est articulé autour de la réalisation d'un prototype d'une salle de cours ou de TP domotique, à base de capteurs de température et d'humidité, de flamme, d'antivol et aussi la sécurité de salle de cours et de TP en intégrant un contrôle d'accès aux étudiants ainsi qu'aux enseignants. Aussi notre prototype se charge de contrôler le chevauchement des occupations de salles de cours et de TP et la protection des équipements pédagogiques dans les salles de cours et de TPs. Une coordination entre laborantins et enseignants de TP via une interface graphique conçue sous environnement de programmation Processing est établie.

Les informations collectées par les capteurs sont analysées à l'aide d'une carte ARDUINO MEGA. Ensuite, ces données sont présentées sur un afficheur LCD, tandis que nous pouvons également les visualiser sur une interface personnalisée grâce à une connexion WIFI via ESP32.

En conséquence, la demande pour les salles de cours ou de TPs domotiques devrait augmenter en raison de la disponibilité des équipements de confort et de sécurité, ainsi que de la réduction des coûts. Dans le cadre de notre projet de recherche, nous cherchons à intégrer davantage la technologie moderne dans ces espaces en utilisant des solutions open source. **Mots clés :** Domotique, salle de cours, salle de TP, Cartes Arduino, capteurs, Processing. Communication wifi.

## **Annexes**

---

Table des matières .....	I
Liste des figures... .....	II
Liste des tableaux.....	III
Glossaire.....	IV
Liste des annexes.....	V
Introduction générale.....	1
Introduction .....	3
Définition de la Domotique.....	3
I.3 Historique et Evolution de la Domotique .....	3
Principe de Fonctionnement de la Domotique .....	4
Domaine d'application de la domotique .....	4
La gestion de l'énergie .....	5
La sécurité .....	5
Le confort .....	5
L'accessibilité .....	5
La communication .....	5
Le divertissement .....	6
La santé .....	6
Limites de la domotique .....	6
Coût élevé .....	6
Complexité .....	6
Dépendance à l'électricité et à internet .....	6
Sécurité .....	6
mises à jour .....	7
Définition de la Maison Intelligente .....	7
Définition de la Salle Intelligente .....	7
Caractéristiques de la Salle Intelligente .....	8
Systèmes de contrôle automatisés .....	8
Capteurs .....	8
Équipements connectés .....	8
Éclairage intelligent .....	9

Systèmes de sécurité .....	9
Gestion de l'énergie .....	9
Télésurveillance .....	9
Conclusion .....	9
Introduction .....	10
Arduino .....	10
Avantages de l'Arduino .....	11
Carte Arduino UNO .....	12
Caractéristiques de la carte UNO .....	12
Brochage de la carte UNO .....	13
Carte Arduino Méga 2560 .....	14
Brochage de la carte Méga 2560 .....	14
Caractéristiques de La carte Arduino Méga 2560 .....	15
Afficheur LCD .....	15
Afficheur LCD 16×02 .....	16
Afficheur LCD 20×04 .....	16
Brochage LCD 16×2 et 20×4.....	17
Capteur Detection .....	17
Introduction .....	17
Esp32 .....	18
Caractéristique ESP32 .....	19
 Buzzer... .....	20
Caractéristiques du buzzer.....	20
Caractéristiques du buzzer .....	20
Brochage du buzzer .....	20
Relais .....	21
Caractéristiques du relais .....	21
Brochage du relais .....	21
Empreinte Digitale .....	22
Fonctionnement de l'EMPREINTE Digitale .....	59

Camera de Surveillance .....	60
Caractéristiques Caméra de Surveillance .....	90
Capteur de mouvement .....	91
DHT22.....	92
Caractéristiques du DHT22 .....	98
 Capteur de gaz MQ2 ET MQ7 .....	109
II.15. 1. Caractéristique du MQ2 ET MQ7.....	105
2. FONCTIONNEMENT MQ2 .....	107
Capteur de flamme KY-026 .....	108
Caractéristique du KY-026 .....	109
Brochage du KY-026 .....	110
Alimentation .....	111
Langage de programmation .....	125
Logiciel Arduino .....	128
Principe général d'utilisation .....	129
Description des menus .....	120
Description de la barre des boutons .....	122
Processing .....	123
Interface de Processing .....	124
Conclusion .....	125
III-1. INTRODUCTION .....	126
III-2. Schéma synoptique générale .....	127
Etude pratique et fonctionnement de la salle intelligente .....	130
Système de détection de gaz .....	131
Système de détection de flamme .....	132
Système de température et d'humidité .....	133
Simulation sous proteus et réalisation pratique de température et d'humidité .....	134
Système d'INTRUSION.....	135
Simulation sous proteus et réalisation pratique de système d'antivol .....	136
Système de messagerie .....	137

Système de contrôle des volets .....	138
Système d'accès .....	139
Simulation sous proteus et réalisation pratique de système d'accès .....	140
Système de communication .....	142
Interface graphique de contrôle de la salle de cours ou de TP .....	145
Présentation d'interface .....	146
Schéma global du prototype de la salle de cours ou de TP domotique .....	146
Conclusion .....	147
Conclusion générale.....	150

## Liste des figures

### **CHAPITRE II**

Figure II.1 - Brochage de la carte UNO .....	13
Figure II.2 –Arduino Méga 2560 .....	14
Figure II.3 -Afficheur LCD 16×02 .....	15
Figure II.4 -Afficheur LCD 20×04 .....	15
Figure II.5-broches de l'afficheur LCD. .....	
Figure II.6 - .....	17
Figure II.7 -Brochage du Clavier 4×4 .....	17
Figure II.8 – Esp32 .....	18
Figure II.9 – Buzzer .....	19
Figure II.10 – Brochage du buzzer .....	19
Figure II.11 – Relais .....	20
Figure II.12- Brochage de relais .....	21
Figure II.15 -Détections. ....	23

Figure II.16 -Empreinte Digitale .....	25
Figure II.17 – Cam Surv capteur IR .....	26
Figure II.18 – Capteur de température et d'humidité DHT22 .....	26
Figure II.19 -Brochage du DHT22 .....	27
Figure II.20– MQ2 .....	28
Figure II.21 – KY-026 .....	29
Figure II.23- Structure d'un programme .....	31
Figure II.24-Espace de développement Intégré (EDI). ....	32
Figure II.25-Menu file. ....	33
Figure II.26 –Menu de Sketch. ....	34
Figure II. 27-Menu de Tools. ....	35
Figure II.28-boutons du logiciel Arduino. ....	35
Figure II. 29-Logo du logiciel processing .....	36

### **Liste des figures**

---

Figure II. 30. L'interface de processing .....	37
--	----

### **CHAPITRE III**

Figure III.1- Schéma synoptique générale .....	39
Figure III.3- Simulation sous proteus de système de gaz .....	40
Figure III.2-Schémas de fonctionnement Système de détection de gaz .....	40
Figure III.4- information de présence de gaz .....	41
Figure III.5 –Etat OFF de système de gaz .....	41
Figure III.6 –Etat ON de système de gaz .....	41
Figure III.8- Simulation sous proteus de système de flamme .....	42
Figure III.7-Schémas de fonctionnement Système de détection de flamme .....	42
Figure III.9- information de présence de flamme .....	43
Figure III.10 –Etat ON de système de gaz et de flamme .....	43
Figure III.11-Schémas de fonctionnement système de température/humidité .....	44
Figure III.13– système d'affichage de température et d'humidité dans l'interface .....	45
Figure III.14-Schémas de fonctionnement système antivol .....	45
Figure III.15–Simulation sous proteus de système d'antivol .....	46

Figure III.16- information de tentation de vol .....	46	
Figure III.17- système d'affichage de tentation de vol dans l'interface graphique... .....	46	
Figure III.18– système de messagerie implémenté sous environnement Processing .....	47	
Figure III.19 –Schéma de fonctionnement de système de contrôle des volets .....	47	
Figure III.20 –Simulation sous proteus de système de contrôle des volets .....	48	
Figure III.21–système de contrôle des volets .....	48	
Figure III.22– système d'affichage de contrôle des volets dans l'interface graphique .....	48	
Figure III.23– système de contrôle de porte dans l'interface graphique .....	49	
Figure III.24-Schémas de fonctionnement système d'accès .....	49	
Figure III.25–Simulation sous proteus de système d'accès .....	50	
Figure III.26- Accès à la salle avec un tag contenant un faux code .....	51	
Figure III.27 Accès à la salle avec un tag contenant le bon code, (b) accès à la salle avec un tag .....	51	
Figure III.28-Affichage d'accès à la salle .....	51	
Figure III.29-Affichage d'accès à la salle sur l'interface graphique Processing. ....	52	
Figure III.30-Affichage d'accès à la salle avec un code erroné .....	52	
Figure III.31. Exemple de communication non filaire .....	53	
Figure III.32-Module wifi ESP32 .....	53	
Figure III.34. Schéma global du prototype .....	54	
Figure	III.33-L'interface	graphique
.....	.....	54

## Liste des Tableaux

---

Tableau II.1 - caractéristiques de la carte UNO .....	12
Tableau II.2 - caractéristiques de la carte Arduino Méga2560.....	14
Tableau II.3 - Brochage LCD 16×02 et 20×4[8] .....	16
Tableau II.4 - caractéristiques Esp32 .....	18

Tableau II.5- caractéristiques du Buzzer .....	19
Tableau II.6 - caractéristiques du relais .....	20
TableauII.7caractéristiquesd'empreinteDigit .....	22
Tableau II.8 - caractéristiques de camera surveillance .....	23
Tableau II.11 – Caractéristiques du DHT22 .....	26
Tableau II.12 – Caractéristiques du MQ2 .....	27
Tableau II.13 – Caractéristiques du KY-026 .....	28

## Glossaire

---

<b>ICT</b>	Information and communication technologies
<b>IoT</b>	Internet of Things
<b>AI</b>	artificial intelligence
<b>LCD</b>	Liquid Crystal Display
<b>ICSP</b>	In-Circuit Serial Programming
<b>UARTs</b>	Universal Asynchronous Receiver Transmitter
<b>SRAM</b>	Static Random Access Memory
<b>EEPROM</b>	Electrically-Erasable Programmable Read-Only Memory
<b>Esp32</b>	Espressif Systems Puce
<b>CPU</b>	Central Processing Unit
<b>Wifi</b>	Wireless Fidelity
<b>GND</b>	Ground
<b>VCC</b>	Voltage continu current
<b>RFID</b>	Radio-Frequency Identification
<b>EDI</b>	Electronic Data Interchange

---

## **Introduction général**

---

## Introduction générale

### Introduction générale

La domotique représente le fruit de l'évolution technologique qui ne cesse de se développé depuis son apparition en 1980 jusqu'à nos jours et cela grâce à la fusion de l'électronique et l'informatique, donnant lieu au système embarqué. La domotique regroupe différentes techniques avancées qui permettent de contrôler, programmer et automatiser les équipements et appareils intégrés dans une habitation à distance ou sur place, en servant l'électronique, l'informatique, les télécommunications et l'automatisme.

L'émergence de systèmes de communication performants, tels que la numérisation des réseaux, a contribué à l'apparition de systèmes innovants axés sur la communication et les échanges à l'intérieur et à l'extérieur de l'habitat. L'objectif de la domotique était d'apporter plus de confort, de sécurité et de convivialité dans la gestion des habitations. C'est dans ce contexte que nous avons eu l'idée de concevoir un système performant pour la sécurité des salles, des ateliers et des laboratoires.

Une salle intelligente utilise des dispositifs de commande et de contrôle pour gérer les différents équipements qui s'y trouvent. Ces dispositifs peuvent être pilotés à distance depuis une interface, et certaines fonctions sont visualisables tandis que d'autres nécessitent des capteurs installés sur place. Ces capteurs contribuent à l'accès à la salle, au contrôle de l'ouverture et de la fermeture des rideaux, ainsi qu'à l'exécution de tâches de sécurité.

Dans notre projet de fin d'étude, nous nous sommes intéressés à la conception d'un système de salle domotique afin d'améliorer la sécurité en évitant les intrusions, le vol et la perte des équipements de la salle. Pour se faire, nous avons mis en œuvre différents systèmes tels que :

## **Introduction générale**

---

Système de sécurité : ce dernier est chargé de surveiller les équipements de la salle et de détecter toute activité suspecte. Il peut comprendre des capteurs de mouvement, des capteurs de gaz et de fumée, des alarmes, etc.

Système d'occupation de la salle : il permet de gérer l'utilisation de la salle en fonction des besoins en matière d'affichage de l'emploi du temps, la liste des étudiants présents. Il peut inclure des modules de présence tels que le module RFID pour détecter l'occupation de la salle et ajuster les paramètres en conséquence.

Système de communication avec les laborantins : ce système facilite la communication entre des enseignants occupant la salle et le personnel de laboratoire, ce dernier est muni des interfaces de communication, voir même des systèmes d'interphonie, etc.

Pour mieux cerner notre travail, on s'est basé sur trois chapitres :

Le premier chapitre est consacré aux notions générales, en expliquant les principes de base, les avantages et les applications de cette technologie.

Le deuxième chapitre aborde les outils matériels et logiciels utilisés pour la réalisation de notre prototype de salle domotique. Où Nous présenterons les composants, les dispositifs de contrôle, ainsi les logiciels que nous avons utilisés, etc.

Le dernier chapitre est dédié aux étapes suivies pour la réalisation pratique de notre système de salle domotique. Où Nous expliquons les processus de conception, d'installation, de configuration et de test du système.

En fin, nous terminons notre projet de fin d'étude par une conclusion générale

# **CHAPITRE I**

---

## **Système Domotique**

---

## **Introduction**

La domotique est un domaine en pleine croissance qui consiste à automatiser, à surveiller et à contrôler les équipements et les systèmes électroniques d'une maison, d'un immeuble ou d'un lieu de travail tel qu'une salle de cours intelligente et qui fait l'objet de notre projet de fin d'étude. La domotique est de plus en plus considérée comme un élément clé d'une salle de cours intelligente tout en offrant un environnement pratique, surveillé et sécurisé. Pour achever à la réalisation de notre salle de cours intelligente il faut en premier mieux cerner l'aspect de la domotique dont le premier chapitre est dédié à détailler cette dernière.

## **Définition de la Domotique**

La domotique est l'ensemble des technologies et des systèmes qui permettent d'automatiser, de surveiller et de contrôler les différents équipements d'une habitation, d'un bâtiment ou d'une salle, tels que l'éclairage, le chauffage, la ventilation, la climatisation, les appareils électroménagers, les serrures, les volets roulants, les systèmes de sécurité, les systèmes audio et vidéo etc ...

La domotique vise à améliorer le confort et la sécurité des occupants de la pièce intelligente comme elle permet notamment de programmer et de contrôler les différents équipements à distance à l'aide d'un ordinateur, d'un Smartphone ou d'une tablette.

Le principe de fonctionnement de la domotique repose essentiellement sur l'utilisation de capteurs, de détecteurs, de réseaux de communication et de logiciels d'automatisation pour surveiller et contrôler les différents équipements d'une maison, d'un immeuble ou d'un lieu de travail tel qu'une salle de cours intelligente. Elle peut être mise en place dans des bâtiments neufs ou existants, et peut être adaptée aux besoins spécifiques de chaque utilisateur.[1]

### I.3 Historique et Evolution de la Domotique

La domotique remonte aux années 1970-1980 cependant, à cette époque la domotique était encore un domaine inconnu qui n'était pas encore concrétisé.

Il est important de noter que la domotique a existé sous une forme différente depuis des siècles tels que l'arrosage automatique inventé par les Égyptiens au 15ème siècle avant Jésus- Christ, ou encore le système de portes automatisées de Léonard de Vinci en 1486.

D'autres inventeurs ont également contribué à l'évolution de la domotique, notamment Eugène Robert-Houdin en 1850 avec son système électrique de portail qui s'ouvre tout seul dès que l'on y sonne. Alexander Graham Bell en 1876 avec l'invention du téléphone qui était l'invention du 21 siècles et Christian Hulsmeyer en 1904 avec l'invention du radar. Enfin, Eugène Polley en 1955 a créé la première télécommande sans fil.

Dans les années 1990, l'avènement d'Internet a permis de connecter les différents équipements de la maison, ce qui a ouvert la voie à de nouveaux services de la domotique.

Au cours des dernières années, la domotique a connu une évolution rapide avec l'émergence de nouvelles technologies telles que les réseaux sans fil, l'intelligence artificielle et l'Internet des objets (IoT). Les maisons intelligentes, les immeubles intelligents voire même les lieux de travail sont devenus plus abordables et plus accessibles.

Aujourd'hui, la domotique est utilisée dans une grande variété d'applications, allant des maisons intelligentes aux bâtiments commerciaux et industriels, en passant par les villes intelligentes jusqu'aux endroits de travail.

Les technologies de domotique sont utilisées pour améliorer la sécurité, le confort et l'efficacité énergétique, tout en offrant de nouvelles opportunités de connectivité et de contrôle aux utilisateurs.[2]

## **Principe de Fonctionnement de la Domotique**

Le principe de fonctionnement de la domotique consiste à intégrer des technologies de l'information et de la communication (TIC) dans les différentes fonctions de la maison, du bâtiment ou d'un lieu de travail tout en programmant et en contrôlant à distance ou sur place une panoplie d'équipements. Ces différentes fonctions sont interconnectées et communiquent entre elles grâce à un système centralisé qui peut être contrôlé à distance via un ordinateur, une tablette, un Smartphone ou une télécommande. L'objectif est d'optimiser la gestion des ressources énergétiques et de faciliter la vie quotidienne des occupants en offrant des fonctionnalités personnalisées et préprogrammées selon les besoins des utilisateurs.

## **Domaine d'application de la domotique**

Le domaine d'application de la domotique comprend l'ensemble des secteurs où la technologie est utilisée pour automatiser et gérer les différents équipements d'un bâtiment ou d'un lieu de vie. Cela inclut notamment la maison intelligente, les lieux de travail, l'hôtellerie, les bâtiments publics, la santé et l'agriculture. La domotique peut être utilisée pour différents besoins, tels que la gestion de

l'éclairage, la sécurité, le chauffage, la climatisation, la commande des appareils électroménagers, la surveillance vidéo dans un but de confort et de sécurité.

### **La gestion de l'énergie :**

La domotique permet de contrôler et de réguler la consommation d'énergie dans une habitation en fonction des besoins. Elle peut notamment gérer l'éclairage, le chauffage, la climatisation en fonction de la présence ou de l'absence des occupants. [3]

### **La sécurité :**

La domotique offre des solutions de sécurité pour protéger les biens et les personnes. Elle peut détecter les intrusions, les incendies, les fuites de gaz, les inondations, etc. et déclencher des actions d'alerte ou de prévention via capteurs, des détecteurs de mouvement, des caméras de surveillance où des alarmes peuvent être programmés pour signaler tout comportement suspect en alertant le propriétaire ou les services de sécurité.

### **Le confort :**

La domotique permet d'optimiser le confort dans une habitation en offrant des solutions pour faciliter la vie quotidienne par le biais de contrôler plusieurs éléments de la maison à distance. Elle peut gérer l'ouverture et la fermeture des volets, l'arrosage automatique, l'ouverture des portes et des portails, réglage de la température de la maison etc...

### **L'accessibilité :**

La domotique offre des solutions pour faciliter l'accès aux équipements et aux espaces pour les personnes à mobilité réduite (handicap ou âgées). En leur permettant de contrôler les équipements de la maison à distance, sans avoir à se déplacer.

### **La communication :**

La domotique offre des solutions de communication pour faciliter les échanges entre les habitants d'une habitation et avec l'extérieur. Elle peut gérer les appels téléphoniques, les interphones, les vidéo-phones, les systèmes de diffusion sonore, etc.

### **Le divertissement :**

La domotique permet de gérer les équipements de divertissement dans une habitation, tels que la télévision, la musique, les jeux vidéo, etc. Elle peut également offrir des solutions de home cinéma avec une qualité d'image et de son optimisée.

### **La santé :**

La domotique offre des solutions pour surveiller la santé des personnes à domicile, notamment les personnes âgées ou atteintes de maladies chroniques. Elle peut mesurer les paramètres de santé, tels que la tension artérielle, la glycémie, etc. et envoyer des alertes en cas d'anomalies.

### **Limites de la domotique**

Limites de la domotique

Voici quelques limites de la domotique :

#### **Coût élevé :**

Le coût d'achat et d'installation d'un système domotique peut être très élevé, en particulier pour les systèmes personnalisés, ce qui peut rendre la technologie inaccessible pour certaines personnes.

#### **Complexité :**

La domotique implique l'intégration de plusieurs technologies et systèmes différents, ce qui peut rendre la configuration et la maintenance du système complexes et potentiellement problématiques, en particulier pour les personnes qui ne sont pas familières avec la technologie.

#### **Dépendance à l'électricité et à internet :**

La plupart des systèmes domotiques nécessitent une connexion Internet fiable et une source d'électricité constante pour fonctionner correctement. Ce qui peut poser des problèmes en cas de panne de courant ou de perte de connexion.

### **Définition de la Maison Intelligente**

La maison intelligente est considérée comme une évolution de la domotique, avec une portée plus large qui inclut les technologies de l'Internet des objets (IoT) et de l'intelligence artificielle (IA). La maison intelligente, également connue sous le nom de maison connectée, qui est un concept de logement équipé de technologies avancées permettant de créer un environnement connecté et automatisé. La maison intelligente utilise des dispositifs électroniques, des capteurs, des systèmes de communication et des logiciels de gestion pour surveiller et contrôler différents équipements tels que

l'éclairage, le chauffage, la climatisation, les appareils électroménagers, les serrures et les systèmes de sécurité.

L'objectif principal de la maison intelligente est d'améliorer la qualité de vie de ses occupants en leur offrant des avantages tels qu'une meilleure sécurité, une gestion énergétique plus efficace, un confort accru et une plus grande commodité. Les occupants peuvent contrôler et surveiller les différents équipements à distance via leur smartphone, leur tablette ou leur ordinateur.

## **Définition de la Salle Intelligente**

Une salle intelligente est une salle équipée de technologies avancées telles que des capteurs, des écrans interactifs, des systèmes de communication et des logiciels de gestion qui permettent de créer un environnement connecté et automatisé. Cette salle peut être utilisée pour des activités telles que des réunions, des présentations, des cours, des travaux pratiques ou des conférences.

Les salles intelligentes sont conçues pour améliorer l'efficacité, la productivité et la surveillance des équipements présents dans la salle. Elles sont souvent équipées de systèmes de contrôle centralisés qui permettent aux utilisateurs de gérer facilement les différents équipements et de personnaliser l'environnement en fonction de leurs besoins.

Les salles intelligentes peuvent être utilisées dans divers domaines tels que l'enseignement, la formation, les affaires et l'industrie, et sont de plus en plus populaires en raison de leur capacité à améliorer l'efficacité et la collaboration.

## **Caractéristiques de la Salle Intelligente**

Une salle intelligente est une pièce équipée d'un système de gestion automatisé qui permet de contrôler et d'optimiser les différents éléments de la salle tels que l'éclairage, la température, la ventilation, les stores, les projecteurs, etc. Les caractéristiques de la salle intelligente peuvent inclure :

### **Systèmes de contrôle automatisés :**

Les systèmes de contrôle automatisés permettent de contrôler les différents équipements de la salle en temps réel à partir d'une interface graphique telles que l'ouverture et la fermeture des portes, l'allumage et l'extinction des lumières, la gestion de la température et la sécurité.

### **Capteurs :**

Les capteurs sont des dispositifs électroniques qui permettent de mesurer différentes grandeurs physiques telles que la température, l'humidité, la présence de personnes, etc.

### **Équipements connectés :**

Les équipements connectés, tels que les projecteurs, les écrans, les haut-parleurs, etc. peuvent être contrôlés à distance à partir d'un smartphone, d'une tablette ou d'un ordinateur.

### **Éclairage intelligent :**

Les éclairages de la salle peuvent être contrôlés automatiquement ou manuellement pour s'adapter aux différentes situations et besoins des utilisateurs.

### **Systèmes de sécurité :**

La salle est équipée de dispositifs de sécurité tels que des caméras, des capteurs de mouvement, des alarmes et des empreintes digitales pour garantir la sécurité des utilisateurs et des équipements.

### **Gestion de l'énergie :**

La salle est équipée de dispositifs pour surveiller et gérer la consommation d'énergie, afin de réduire les coûts et de minimiser l'impact environnemental.

## **Télésurveillance**

La télésurveillance est une technologie permettant de surveiller à distance un lieu, tel qu'un domicile, un commerce ou un bureau. Elle peut être utilisée pour surveiller les accès, détecter des intrusions, des incendies, des fuites de gaz ou d'eau, ou encore pour suivre des personnes âgées ou des enfants. Le système de télésurveillance se base généralement sur des caméras de surveillance reliées à un système d'alarme qui envoie une alerte en cas de détection d'un événement suspect, le système peut également être équipé de capteurs de mouvements, de détecteurs d'intrusion et d'autres dispositifs pour surveiller les activités suspectes. Les images sont alors consultables à distance par les utilisateurs autorisés, qui peuvent également prendre des mesures en conséquence, telles que contacter les autorités compétentes. La télésurveillance peut ainsi contribuer à renforcer la sécurité d'un lieu et à prévenir les risques de cambriolage ou d'incidents.

## **Conclusion**

La salle intelligente représente le futur de l'enseignement. Grâce à ses nombreuses fonctionnalités, elle permet d'améliorer le quotidien des étudiants tout en facilitant le travail des enseignants. La salle intelligente doit être envisagée comme un investissement à long terme qui nécessite une réflexion approfondie et une planification minutieuse pour en tirer tous les avantages. Les informations

présentées dans ce chapitre sont d'une utilité primordiale pour la compréhension de la suite de notre travail.

La conception de la salle intelligente est assurée par des capteurs ainsi que des circuits numériques comme la carte Arduino qui seront développés dans le deuxième chapitre.

# CHAPITRE II

---

## Outils de Développement

---

### Introduction

La réalisation de notre prototype de salle de cours intelligente ne peut voir le jour sans l'utilisation d'un nombre important des outils de développement ainsi que des capteurs. Le présent chapitre met en évidence les différents outils de développement que nous avons usité dans la conception de notre projet de fin d'études, tout en présentant les outils les plus importants pour créer notre prototype à savoir les outils software ou hardware.

En commençant avec la carte Arduino ainsi que son environnement de programmation, l'afficheur LCD, le clavier etc... en finissant avec la présentation des capteurs utilisé.

.

## **Arduino**

C'est une carte électronique programmable destiné à réaliser des tâches selon les besoins de l'utilisateur via un programme préalablement écrit par la suite télé versé sur la carte Arduino. Tout en offrant une multitude de possibilités de combinaisons de circuits électroniques simples avec un faible coût.

Le système Arduino se charge d'assurer la l'union entre les performances de la programmation à celles de l'électronique.

Arduino offre diverses opérations dans de différents domaines tels que :

- Le contrôle des appareils domestiques.
- La fabrication des robots.
- La réalisation d'un jeu de lumières.
- L'interfaçage homme-machine.
- Télécommander un appareil mobile [4]

## **Avantages de l'Arduino**

En vue de différent avantage qu'elle offre la carte arduino, cette dernière a pris une place indispensable dans l'environnement des chercheurs et parmi ces avantages on cite :

- Ça capacité à de piloter des capteurs grâce à son interface programmable
- La carte reçoit des informations analogiques ou numériques sur ces entrées. Le Microcontrôleur traitera ces informations et les transmettra vers les sorties numériques.
- Les cartes Arduino sont peu coûteuses comparativement aux autres plateformes.

- Elle simplifie considérablement les schémas électroniques et par conséquent, le coût de la réalisation.
- Le logiciel Arduino, écrit en Java il peut être exploité par différents systèmes tels que Windows, Macintosh et Linux dont la plupart des systèmes à microcontrôleurs sont limités à Windows.
- Le logiciel Arduino et le langage Arduino sont publiés sous licence open source,
- L'Arduino est conçu pour être modulaire, ce qui signifie que les utilisateurs peuvent ajouter des modules complémentaires (appelés "shields") pour ajouter des fonctionnalités supplémentaires à leur projet.
- Arduino est une plate-forme qui a été conçue pour permettre aux personnes de tous âges et de tous niveaux de compétence de créer facilement des objets interactifs.
- L'Arduino bénéficie d'une large communauté de développeurs qui partagent des tutoriels, des projets et des astuces en ligne, ce qui facilite l'apprentissage et la résolution des problèmes. □ Nombreuses librairies disponibles avec diverses fonctions implémentées. [5]

## **Carte Arduino UNO**

Arduino UNO est une carte électronique programmable basée sur le microcontrôleur l'ATmega328 où elle se constitue de 14 broches entrées/sorties numériques , 06 entrées analogiques (qui peuvent également être utilisées en broches entrées/sorties numériques), une connexion USB, un oscillateur à quartz 16Mhz, un support ICSP et d'un bouton reset et un connecteur d'alimentation jack. Pour commencer à utiliser la carte Arduino UNO, il suffit de la connecter à un ordinateur à l'aide d'un câble USB. [6]

## **Caractéristiques de la carte UNO**

Le tableau ci-dessous présente les caractéristiques de la carteUNO

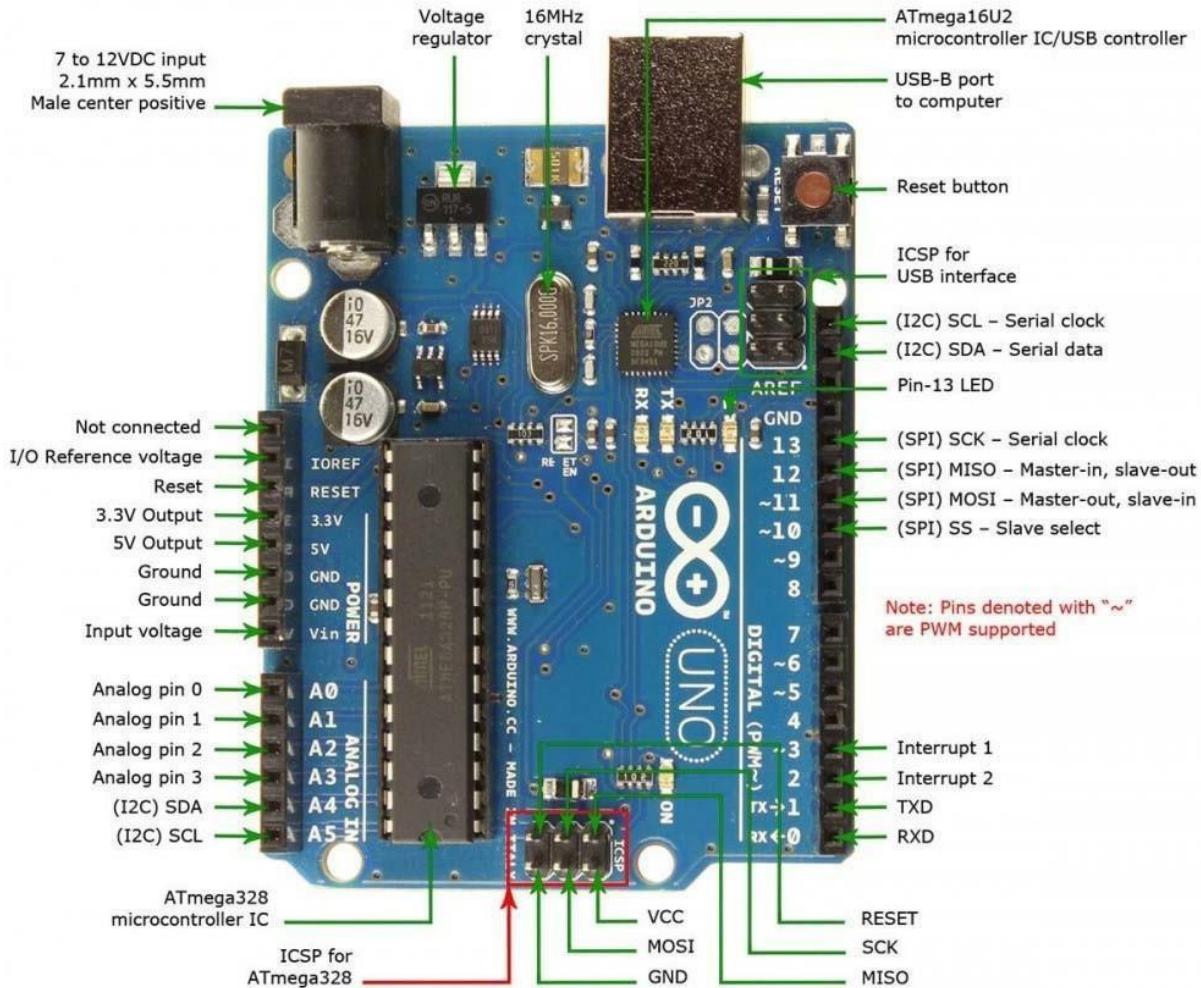
Microcontrôleur	ATMEGA328
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7-12V
Tension d'alimentation (limites)	6-20V

Broches E/S numérique	14 (dent 6 disposent d'une sortie PWM)
Broches d'entrées analogiques	6(utilisables en broches E/S numérique)
Intensité maxi disponible par broches E/S (5V)	40mA (ATTENTION :200 mA annulé pour l'ensemble des broches E/S)
Intensité maxi disponible pour la sortie 3.3V	50mA
Intensité maxi disponible pour la sortie 5V	Fonction de l'alimentation utilisé-500mA max si port USB utilisable seul
Mémoire programme Flash	32 KB (AT méga <u>328</u> ) <u>dont</u> 0.5 KB sont utilisés par le <u>bootloader</u>
Mémoire SRAM (mémoire volatile)	2KB (AT méga 328)
Mémoire EEPROM (mémoire non volatile)	1 KB (ATmega328)
Vitesse d'horloge	16 MHz

**Tableau II.1 - caractéristiques de la carte UNO**

## **Brochage de la carte UNO**

Le brochage de la carte UNO est illustré dans la figure ci-dessous [7]



**Figure II.1 - Brochage de la carte UNO[7]**

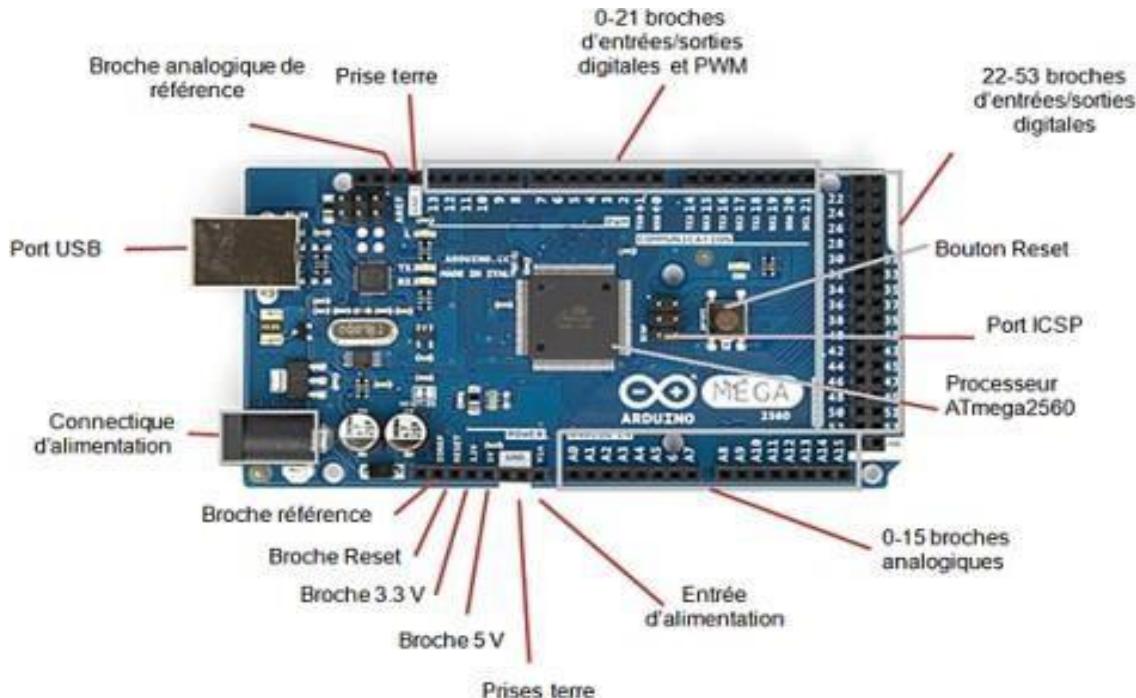
## Carte Arduino Méga 2560

Arduino Méga 2560(Figure II.2) une carte électronique programmable basée sur le microcontrôleur ATMega2560 cadencé à 16 MHz. Elle dispose de 54 broches entrées/sorties numériques, 16 analogiques, 4 UARTs, une connexion USB, une prise d'alimentation, et un bouton de réinitialisation.

La carte Méga 2560 est idéale pour des projets exigeant un grand nombre de pines entrées/sorties

## Brochage de la carte Méga 2560

Le brochage de la carte **Méga 2560** est illustré dans la figure ci-dessous



**Figure II.2 –Arduino Méga 2560**

### Caractéristiques de La carte Arduino Méga 2560

Les caractéristiques de la carte Arduino Méga sont citées dans le tableau ci-dessous :

Micro contrôleur	ATmega2560
Tension d'alimentation interne	5 v
Tension d'alimentation (recommandée)	7 à 12V
Tension d'alimentation (limitée)	6 à 20 v
Entrées/sorties numériques	54 dont 14 sorties PWM (largeur d'impulsion modulée)
Entrées analogiques	16 ( résolution de 10 bits )
Courant max par broches E/S	40 mA
Mémoire Flash	256 KB
Mémoire SRAM	8 KB
Mémoire EEPROM	4 KB
Fréquence horloge	16 MHz
Dimensions	107 x 53 x 15 mm

**Tableau II.2 - caractéristiques de la carte Arduino Méga2560**

### Afficheur LCD

Les afficheurs à cristaux liquides, également appelés afficheurs LCD (Liquid Crystal Display), sont des modules intelligents compacts qui nécessitent peu de composants externes pour fonctionner tout en consommant peu d'énergie. Les afficheurs LCD sont essentiels dans les systèmes d'électroniques

qui ont besoin d'afficher les paramètres de fonctionnement en permettant un affichage facile de messages.

### Afficheur LCD 16x02

Un afficheur LCD 16x02 est un type d'afficheur à cristaux liquides qui est capable d'afficher 16 caractères sur 2 lignes.

### Afficheur LCD 20x04

Un afficheur LCD 20x04 est un type d'afficheur à cristaux liquides qui est capable d'afficher 20 caractères sur 4 lignes.

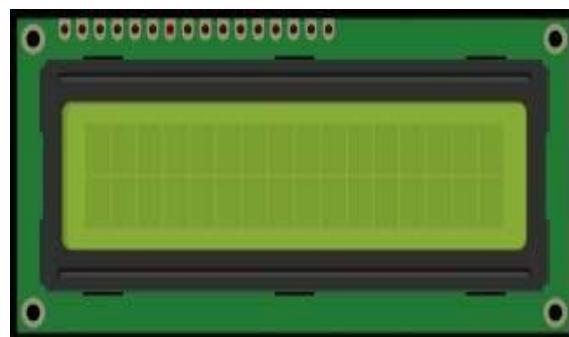


Figure II.3 -Afficheur LCD 16x02



Figure II.4 -Afficheur LCD 20x04

### Brochage LCD 16x2 et 20x4

L'écran LCD 16x02 présente 16 broches pour permettre la gestion de l'affichage et du contraste.

Nom	Rôle
1 V <sub>SS</sub>	Masse
2 V <sub>DD</sub>	+5V
3 V <sub>EE</sub>	Réglage du contraste
4 RS	Sélection du registre (commande ou donnée)
5 R/W	Lecture ou écriture
6 E	Entrée de validation
7 à 14 D0 à D7	Bus de données
15 A	Anode (+5V)
16 K	Cathode (masse)

Tableau II.3 - Brochage LCD 16×02 et 20×4[8]

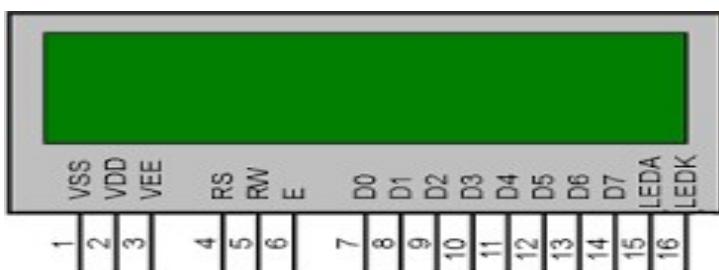


Figure II.5-broches de l'afficheur LCD.

## Esp32

L'ESP32 est un microcontrôleur à faible consommation d'énergie et à haute performance qui intègre le Wi-Fi et le Bluetooth, développé par la société Espressif. L'ESP32 est largement utilisé dans les projets IoT (Internet of Things) et les applications embarquées pour sa polyvalence, ses performances et sa connectivité sans fil avancée, permettant la communication avec d'autres appareils et l'accès à Internet

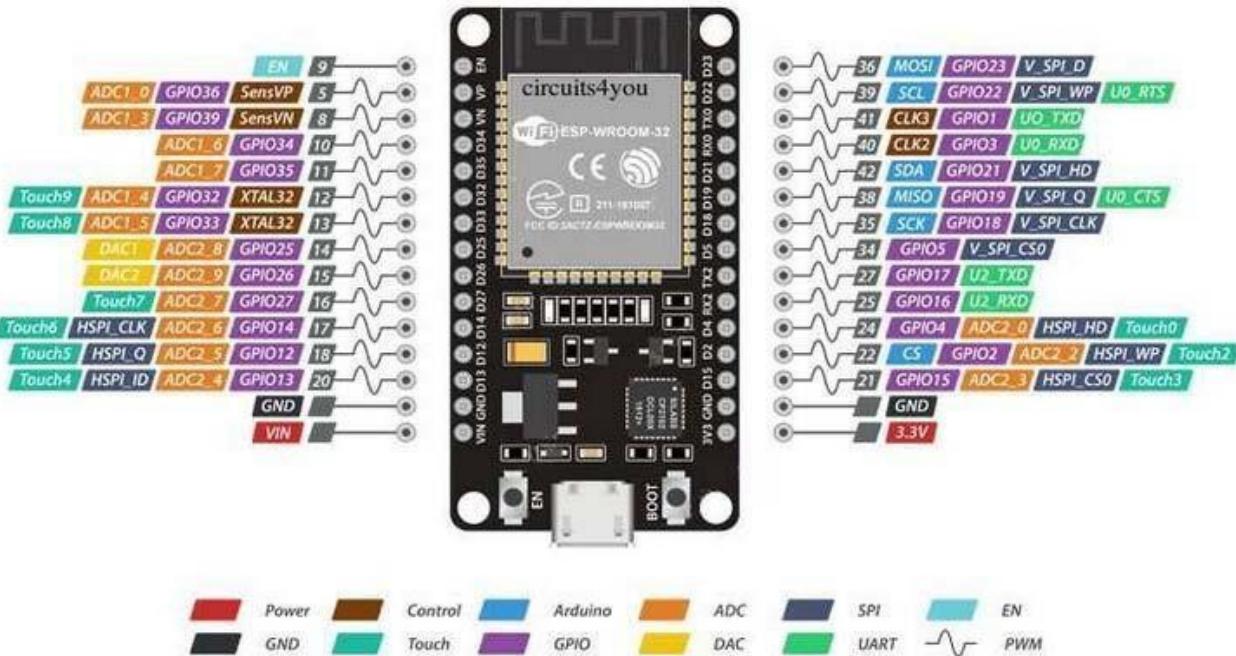


Figure II.8 – Esp32

### Caractéristique ESP32 :

Les caractéristiques de la carte ESP32 sont citées dans le tableau ci-dessous : [10]

CPU	ESP-WROOM-32 (Tensilica Xtensa LX6)
Tension d'alimentation	7-12V
E/S digitales	14
Entrées analogiques	6
Flash	4000kB
SRAM	520kB
EEPROM	448kB
Fréquence d'horloge	240 MHz
Wifi	oui
Bluetooth	oui

Tableau II.4 - caractéristiques Esp32

## Buzzer

Un buzzer est un dispositif électronique utilisé dans l'univers Arduino destiné à produire un son. Le buzzer est composé d'un élément piézoélectrique qui se déforme mécaniquement sous l'effet d'un courant électrique, créant ainsi des vibrations qui génèrent le son. Il est principalement utilisé pour produire des signaux sonores d'alerte, des indications sonores, des alarmes ou des effets sonores dans les jeux électroniques. La figure II.7 montre un buzzer.



**Figure II.9 – Buzzer**

### Caractéristiques du buzzer [11]

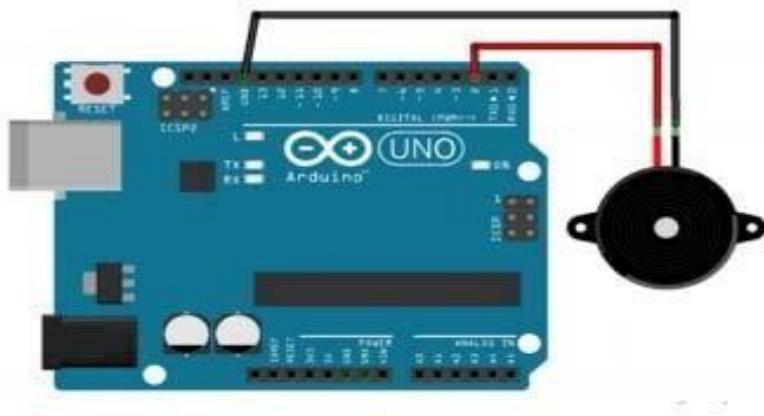
tension de fonctionnement	3,5 à 5.5V
Courant de fonctionnement maximum	30mA à 3V
Sortie sonore minimum	85 dB / 10 cm
température de fonctionnement	-20°C à 70°C
Poids	2g

**Tableau II.5- caractéristiques du Buzzer**

### Brochage du buzzer

Le buzzer possède généralement deux bornes. Il est important de noter que certains buzzers peuvent également avoir une troisième borne dédiée au contrôle de volume.

Une borne du buzzer est connectée au GND et l'autre borne est connectée à une sortie digitale de l'Arduino. La figure II.8 montre comment brancher un buzzer sur l'Arduino



**Figure II.10 – Brochage du buzzer**

## Relais

Un relais est un interrupteur électromécanique qui permet de contrôler un circuit électrique à partir d'un autre circuit. Il est utilisé pour isoler électriquement des circuits, afin de commuter des charges électriques élevées à partir de signaux de faible puissance, ou pour contrôler des circuits haute tension à partir de circuits basse tension. Le relais est identique à d'autres circuits électriques du fait qu'il reçoive un signal électrique et envoie le signal à d'autres équipements en allumant et éteignant l'interrupteur.

Le relais est un contact qui est initialement dans un état fixe, soit normalement fermé, soit normalement ouvert, et il reste dans cet état tant qu'il n'est pas alimenté. L'état du contact ne sera modifié que lorsqu'un courant électrique sera appliqué aux contacts. [12]



**Figure II.11 – Relais**

## Caractéristiques du relais

Les caractéristiques du relais sont citées dans le tableau ci-dessous : [13]

Courant de consommation	58,3 mA (12 V), 29,2 mA (24 V) Courant consommé par la bobine donc courant de commande
Tension min. de commutation	12 V ou 24 V (suivant les modèles)
Courant de commutation max	8A
Tension de commutation/tension max	250 V c.a./440 V <del>c.c.</del>
Durée de vie mécanique	Mécanique : 10 000 000 manœuvres min Nombre de manœuvres garanties par le constructeur
Temps d'enclenchement	15 ms max
Temps de relâchement	5 ms max
Rigidité diélectrique bobine-contacts	4000 V Tension que l'on peut appliquer entre les contacts et la bobine sans risque d'arc électrique
Poids	9g

Tableau II.6 - caractéristiques du relais

## Brochage du relais

- VCC : 3.3V-5 V
- GND : la Masse
- S (Signal) : broche de réception du signal envoyé par arduino
- NC : Normalement fermé
- COM : Commun
- NO: Normalement ouvert (Figure II.11)

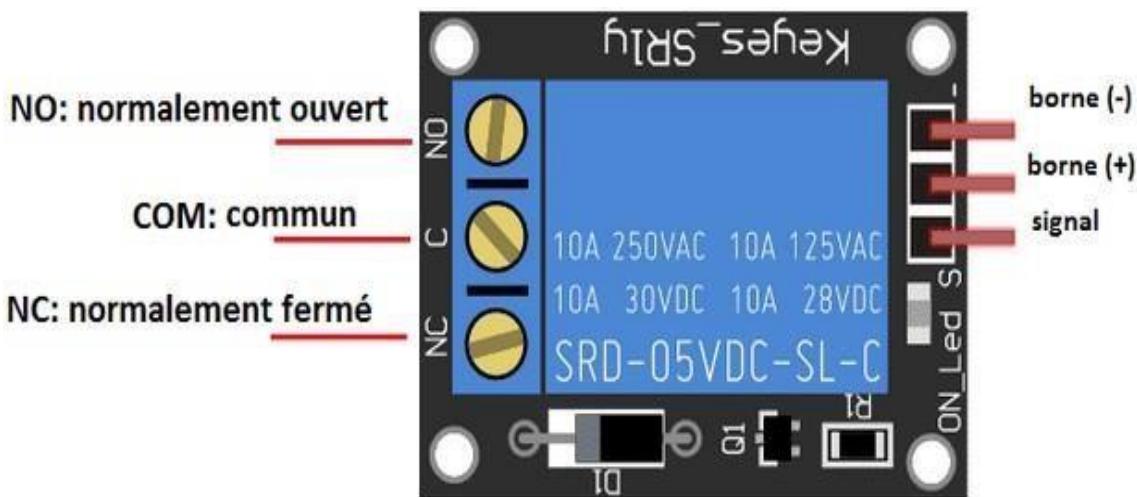


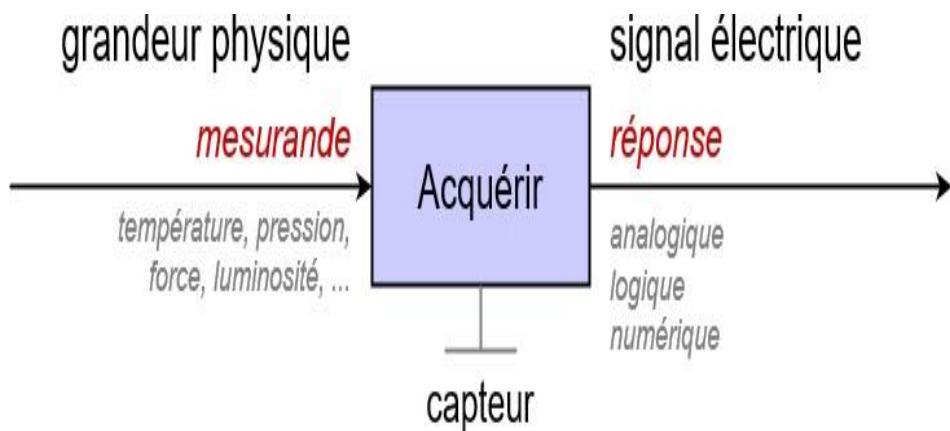
Figure II.12- Brochage de relais

# DETECTION (CAPTEURS)

## Introduction sur les Appareils de Détections

### Capteurs

Les capteurs sont des dispositifs utilisés pour détecter et mesurer des phénomènes physiques ou des conditions environnementales et cela en convertissant une grandeur physique en une grandeur électrique. Ils sont largement utilisés dans de nombreux domaines tels que l'électronique, l'automatisation, les systèmes de contrôle, les sciences, la robotique, etc.



Dans les systèmes de **sécurité et de surveillance modernes**, la détection des anomalies et des dangers est devenue une nécessité incontournable. Divers dispositifs et technologies sont utilisés pour assurer cette détection efficace et rapide, chacun avec des fonctions spécifiques adaptées aux besoins des environnements critiques. Les **appareils détections**, tels que les **lecteurs d'empreintes digitales**, les **caméras de surveillance**, les **capteurs de température**, les **capteurs de fumée et de gaz**, jouent un rôle crucial dans la protection des infrastructures, des individus et des équipements.

Ces dispositifs permettent une **détection en temps réel des incidents**, garantissant ainsi une réponse automatique et immédiate en cas de problème. Ils offrent une meilleure précision et une meilleure fiabilité que les systèmes traditionnels, tout en réduisant considérablement les risques humains et matériels.

- **Les lecteurs d'empreintes digitales**, par exemple, assurent un **contrôle d'accès sécurisé**, empêchant toute tentative d'intrusion non autorisée.
- Les **caméras de surveillance**, équipées d'algorithmes avancés de **vision par ordinateur**, permettent de détecter les **mouvements suspects et les comportements anormaux**, offrant ainsi une surveillance constante des espaces.
- Les **capteurs de température** jouent un rôle essentiel dans la **détection des changements thermiques**, ce qui est crucial pour prévenir les **incendies ou les défaillances des équipements**.
- Les **capteurs de fumée et de gaz**, quant à eux, assurent une protection indispensable dans les environnements industriels et domestiques, en détectant la **présence de substances dangereuses**.

L'intégration de ces technologies dans un **système de détection et d'alerte précoce**, en combinaison avec des outils d'intelligence artificielle, permet une **analyse rapide et précise des données**, garantissant ainsi une réponse efficace et automatisée. Ce système vise à exploiter ces différents dispositifs pour créer un **système robuste et fiable**, capable de surveiller et d'alerter en temps réel, tout en assurant une protection optimale des environnements critiques

## EMPREINTES DIGITALES

### INTRODUCTION

La Reconnaissance des empreintes digitales est une branche de la biométrie la plus rependue, aussi bien dans le domaine de la sécurité publique (contrôle, enquête), que privée (accès à un domicile, protection de biens...). Le principe de la reconnaissance des empreintes digitales consiste à comparer 2 empreintes fournies au système à une ou plusieurs autres empreintes aussi appelé « Template » ou signatures, le système biométrique renvoie un résultat positif au cas où l'empreinte fournie à l'entrée correspond à l'un des Template, et un résultat négatif dans le cas contraire. Les systèmes de reconnaissances des empreintes digitales sont utilisés dans plusieurs applications par exemples : sécuriser l'accès à un ordinateur, et dans le domaine de la criminologie, les services de la police scientifique utilisent l'empreinte digitale comme moyen d'identification d'une personne depuis plus de 100 ans.

### Définition de l'empreinte digitale

L'**empreinte digitale** est une **caractéristique biométrique unique** qui consiste en un ensemble de **crêtes et de minuties** présentes sur la **surface des doigts humains**. Chaque individu possède une empreinte digitale distincte, qui ne peut être ni falsifiée ni reproduite.

Les **crêtes et minuties** sont des motifs naturels formés lors du développement fœtal, et ces motifs restent **inchangés tout au long de la vie** d'une personne. Cela signifie que l'empreinte digitale est une **preuve fiable pour l'identification biométrique**.

L'**utilisation des empreintes digitales** est courante dans divers systèmes biométriques : **contrôles d'accès sécurisés, systèmes judiciaires, banques, et dispositifs électroniques**, garantissant ainsi un haut niveau de sécurité et une authentification efficace des individus.

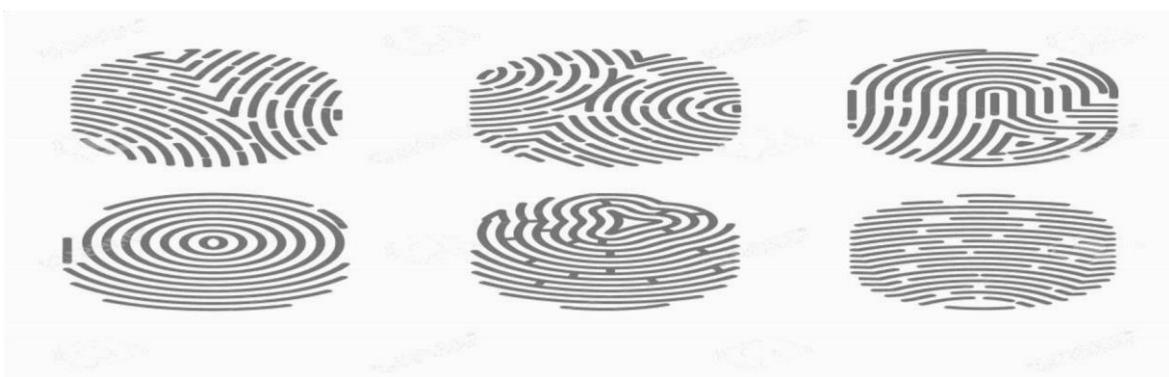


Figure 2.1.1 : images des différentes classes d'empreintes digitales

La Guinée est en pleine transition vers une **modernisation des infrastructures technologiques**, avec une demande croissante pour des systèmes de sécurité robustes. Des technologies comme les **lecteurs biométriques d'empreintes digitales sont adaptées à cette évolution**, soutenant le développement des **smart homes**, des **systèmes bancaires sécurisés et des solutions administratives avancées**

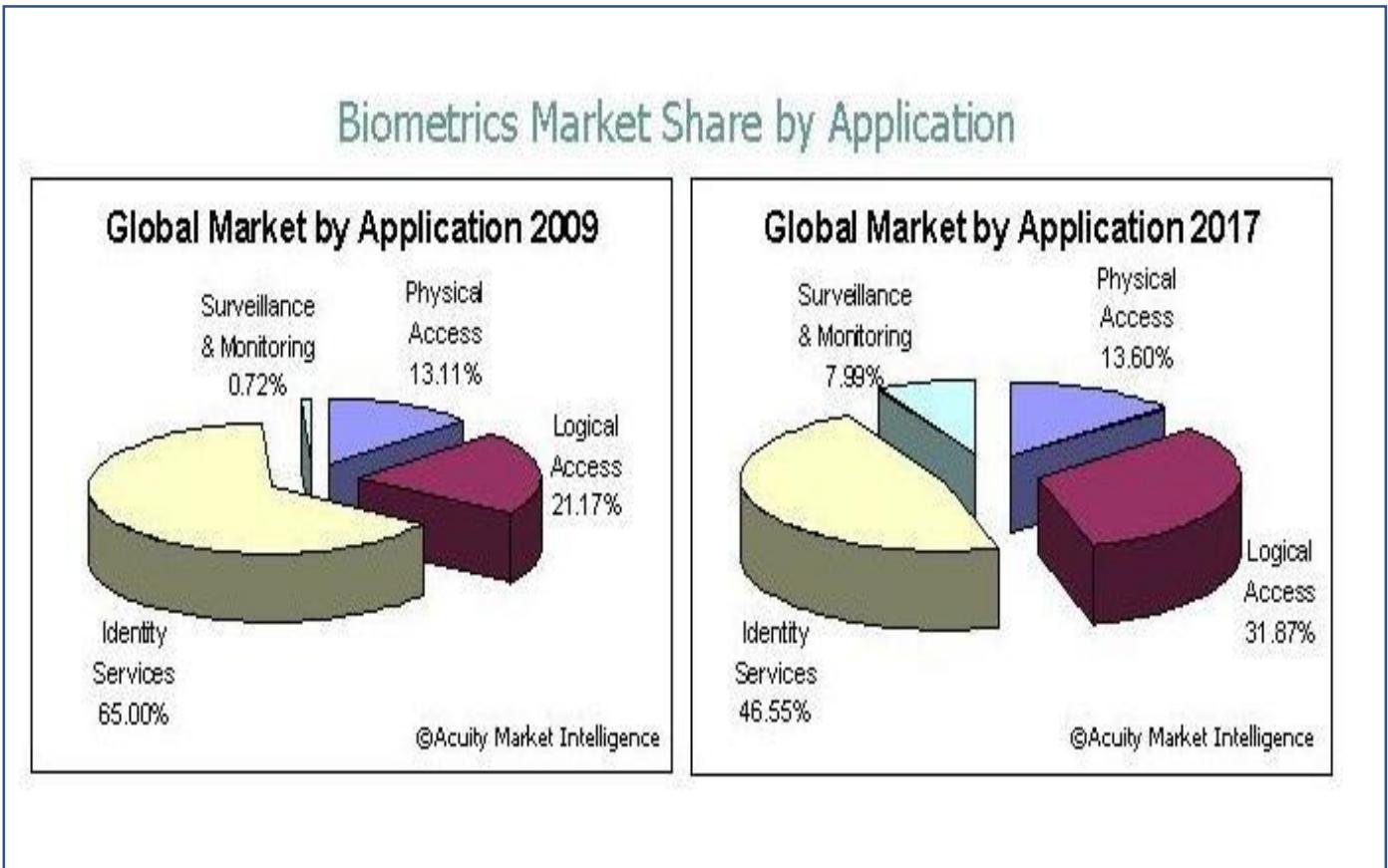
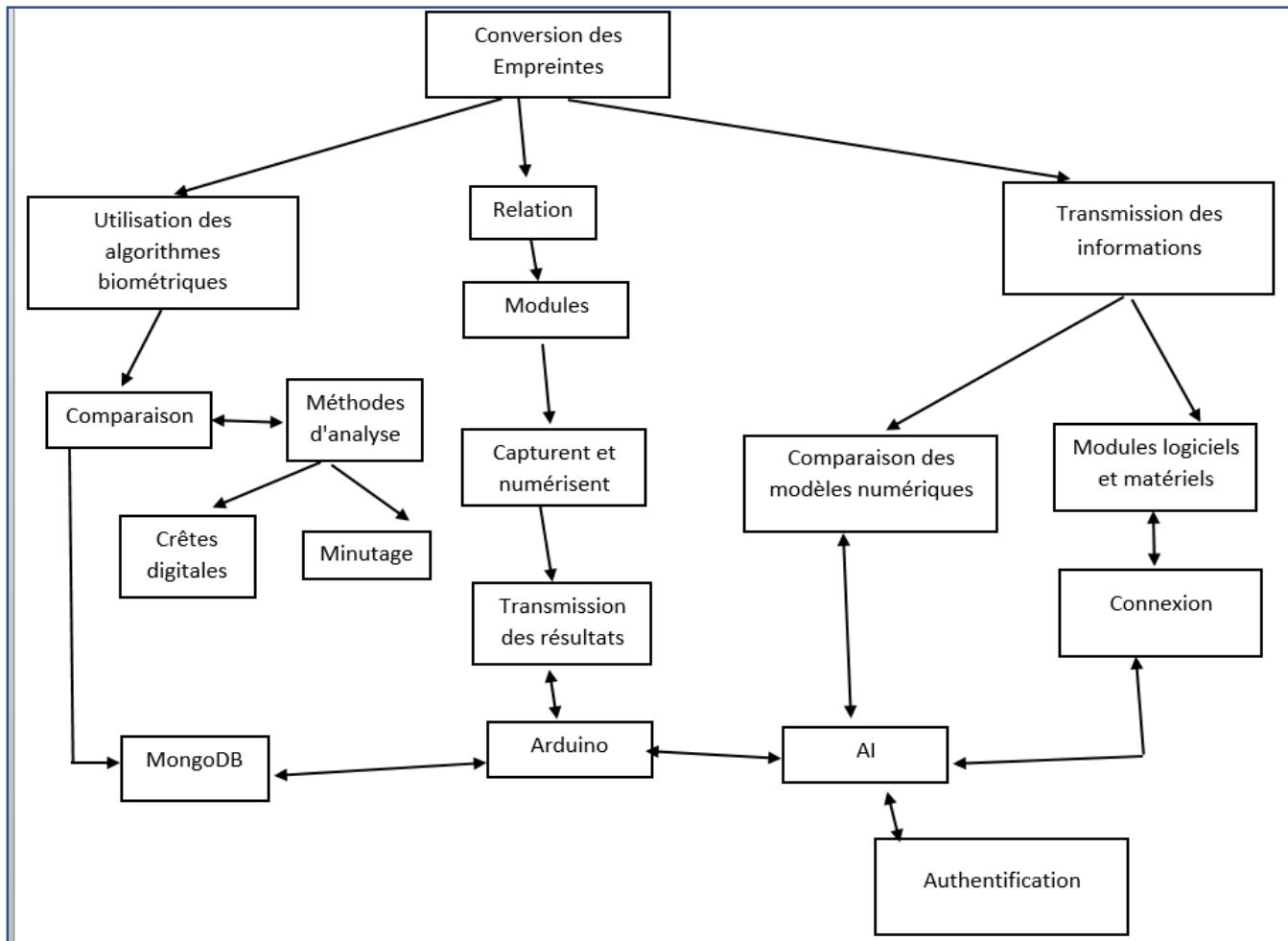


Figure 5 : la croissance dans le marché

## Schéma Synoptique



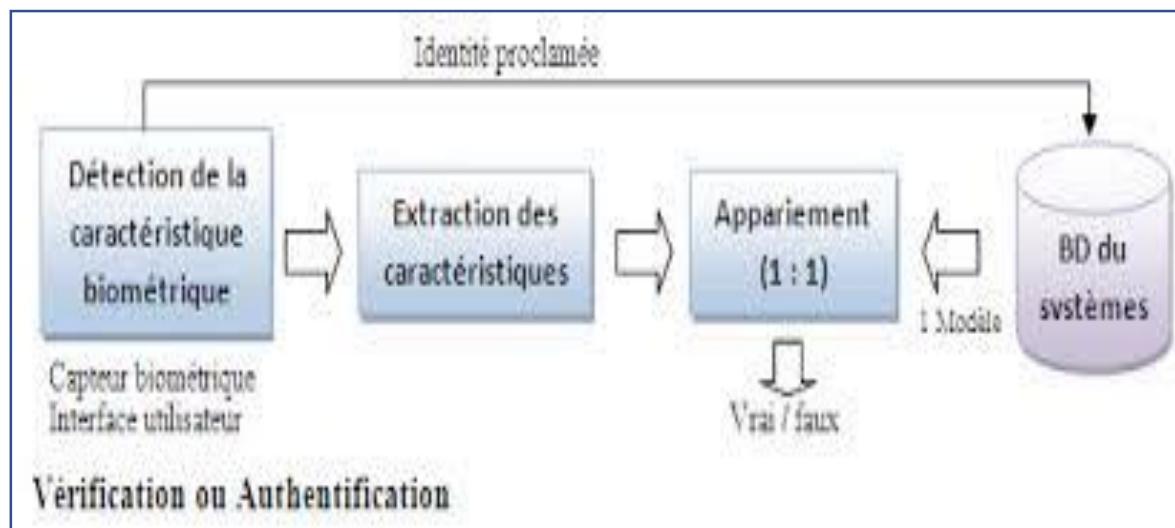
## Principe de fonctionnement

L'authentification par les empreintes digitales repose sur la concordance entre le fichier d'enregistrement, ou « signature », obtenu lors de l'enrôlement et le fichier obtenu lors de l'authentification. Ces deux fonctions se décomposent chacune en plusieurs étapes :

Fonction 1 : Enregistrement (enrôlement) :

- Capture de l'image de l'empreinte. Les données d'un doigt sont en principe suffisantes à l'enrôlement, mais la plupart des systèmes enregistrent au moins deux doigts (un par main par exemple) pour parer l'indisponibilité résultant de petites blessures.

- Numérisation de l'image afin d'extraire les minuties, ou éléments caractéristiques.
- Enregistrement sur un support, (carte à puce, Base de données...).



## Etape 1 :

### 1. Capture des empreintes digitales

- **Matériel utilisé :** Capteur biométrique connecté à une carte **Arduino**.
- **Fonctionnement :**
  - L'empreinte digitale est capturée par un **capteur biométrique** connecté à l'Arduino.
  - L'Arduino recueille l'image des empreintes digitales et l'envoie à l'unité d'analyse.
- **Rôle de l'IA :**

Une fois l'image capturée, des algorithmes basés sur l'IA (souvent implémentés sur une plateforme externe) améliorent cette image en réduisant le bruit et en garantissant la précision des **détails minutieux (crêtes et vallées de l'empreinte)**



: Les Vallée et crêtes d'une empreinte digitale

#### . Extraction des caractéristiques

- **Fonctionnement :**
  - Après capture, l'IA extrait des caractéristiques spécifiques des empreintes digitales comme les **minuties**, qui sont des points clés sur l'empreinte digitale.
  - Des algorithmes d'**apprentissage profond (Deep Learning)** sont utilisés pour reconnaître et extraire ces caractéristiques.
  - Les algorithmes peuvent être exécutés sur une plateforme externe où les capacités de calcul sont plus robustes (serveur, ordinateur).
- **Lien avec Arduino :**
  - Arduino collecte l'empreinte digitale, la transmet au module où l'IA est hébergée, puis elle retourne les **détails extraits** à l'Arduino.

#### 3. Base de Données (DB)

- **Fonctionnement :**
  - Toutes les caractéristiques biométriques (empreintes digitales) sont **stockées dans une base de données centrale**.
  - Cette base de données peut être hébergée sur un serveur Cloud comme MongoDB ou une machine locale avec une architecture robuste.
  - **L'IA compare les caractéristiques extraites des nouvelles empreintes digitales avec celles présentes dans la base de données** pour une **vérification ou une authentification**.
- **Rôle de l'IA :**
  - Des algorithmes d'IA optimisent cette comparaison, assurant une recherche rapide et une correspondance correcte des empreintes, tout en réduisant les faux positifs ou les erreurs de correspondance.

#### 4. Authentification et Appariement des Données

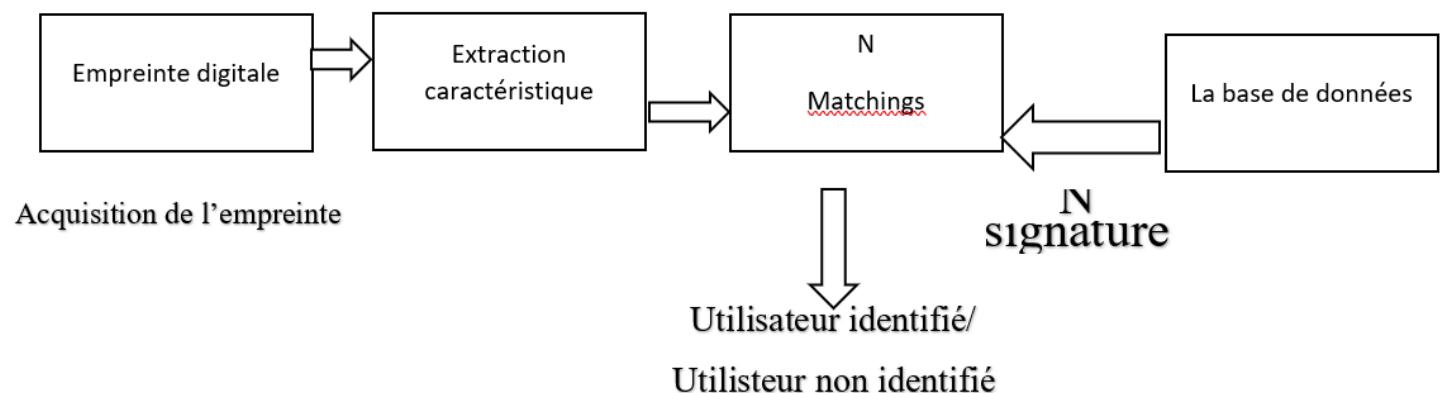
- **Fonctionnement :**
  - Une fois les caractéristiques extraites, l'IA compare cette empreinte digitale à celles présentes dans la **base de données** pour une correspondance **1:1 ou 1:N**.
  - Si la correspondance est correcte, l'utilisateur est authentifié ; sinon, une **alerte est déclenchée** (tentative d'intrusion, erreur biométrique).
- **Lien Arduino – DB – IA :**

- Arduino interagit directement avec le capteur biométrique.
- Les données sont ensuite transmises à l'IA hébergée sur une plateforme externe où l'analyse approfondie est réalisée.

La base de données, qui contient des modèles biométriques, est interrogée et comparée via des algorithmes d'IA pour confirmer l'authenticité des utilisateurs

## Fonction 2 : Authentification :

- Capture de l'image de l'empreinte.
- Numérisation de l'image afin d'extraire les minuties, ou éléments caractéristiques.
- Comparaison entre l'échantillon et le gabarit « signature ».
- Prise de décision.



## Phase authentification de l'empreinte digitale

### Etape 2 :

Ce schéma représente les différentes étapes impliquées dans le processus d'authentification d'un individu à l'aide de son empreinte digitale. Il s'agit d'une méthode biométrique de plus en plus utilisée pour sécuriser l'accès à divers systèmes (smartphones, ordinateurs, portes sécurisées, etc.).

#### Décomposition des étapes

##### 1. Acquisition de l'empreinte digitale:

- **Capture:** Un capteur biométrique (souvent optique ou capacitif) capture une image numérique de l'empreinte digitale de l'utilisateur. Cette image est souvent en haute résolution pour capturer les détails fins des crêtes papillaires.

##### 2. Extraction des caractéristiques :

- **Prétraitement :** L'image capturée est soumise à un traitement numérique pour améliorer la qualité et réduire le bruit.
- **Extraction :** Des algorithmes spécifiques extraient les caractéristiques uniques de l'empreinte digitale. Ces caractéristiques peuvent être des minuties (points terminaux, bifurcations) ou des informations sur la forme des crêtes.

##### 3. Matchings:

- **Comparaison :** Les caractéristiques extraites de l'empreinte digitale à authentifier sont comparées aux caractéristiques des empreintes digitales stockées dans une base de données (MongoDB).

- **Algorithmes de comparaison:** Divers algorithmes sont utilisés pour comparer les empreintes digitales. Ils calculent un score de similarité qui indique le degré de correspondance entre les deux empreintes.
4. **La base de données:**
- **Stockage:** La base de données contient les modèles numériques des empreintes digitales enregistrées. Chaque modèle est associé à une identité unique (par exemple, un nom d'utilisateur).
5. **N signatures:**
- **Signature numérique:** Une signature numérique est une représentation mathématique unique de l'empreinte digitale. Elle sert à vérifier l'intégrité des données biométriques et à prévenir les falsifications.

Décision finale :

En fonction du score de similarité obtenu lors de la comparaison, le système prend une décision :

- **Utilisateur identifié:** Si le score est supérieur à un seuil prédéfini, le système considère que l'empreinte digitale correspond à celle enregistrée dans la base de données et autorise l'accès.
- **Utilisateur non identifié:** Si le score est inférieur au seuil, l'accès est refusé car l'empreinte digitale ne correspond à aucune de celles enregistrées.

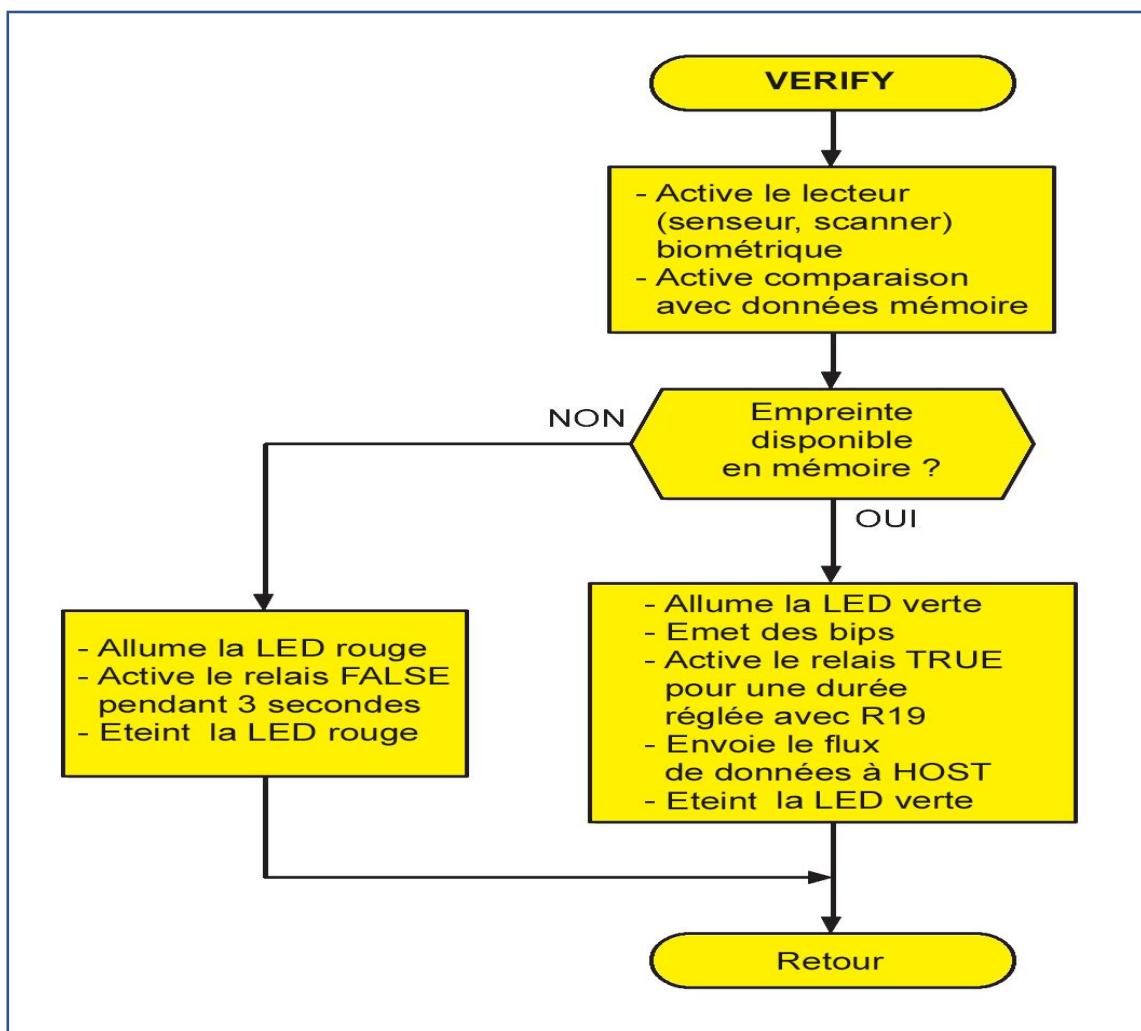
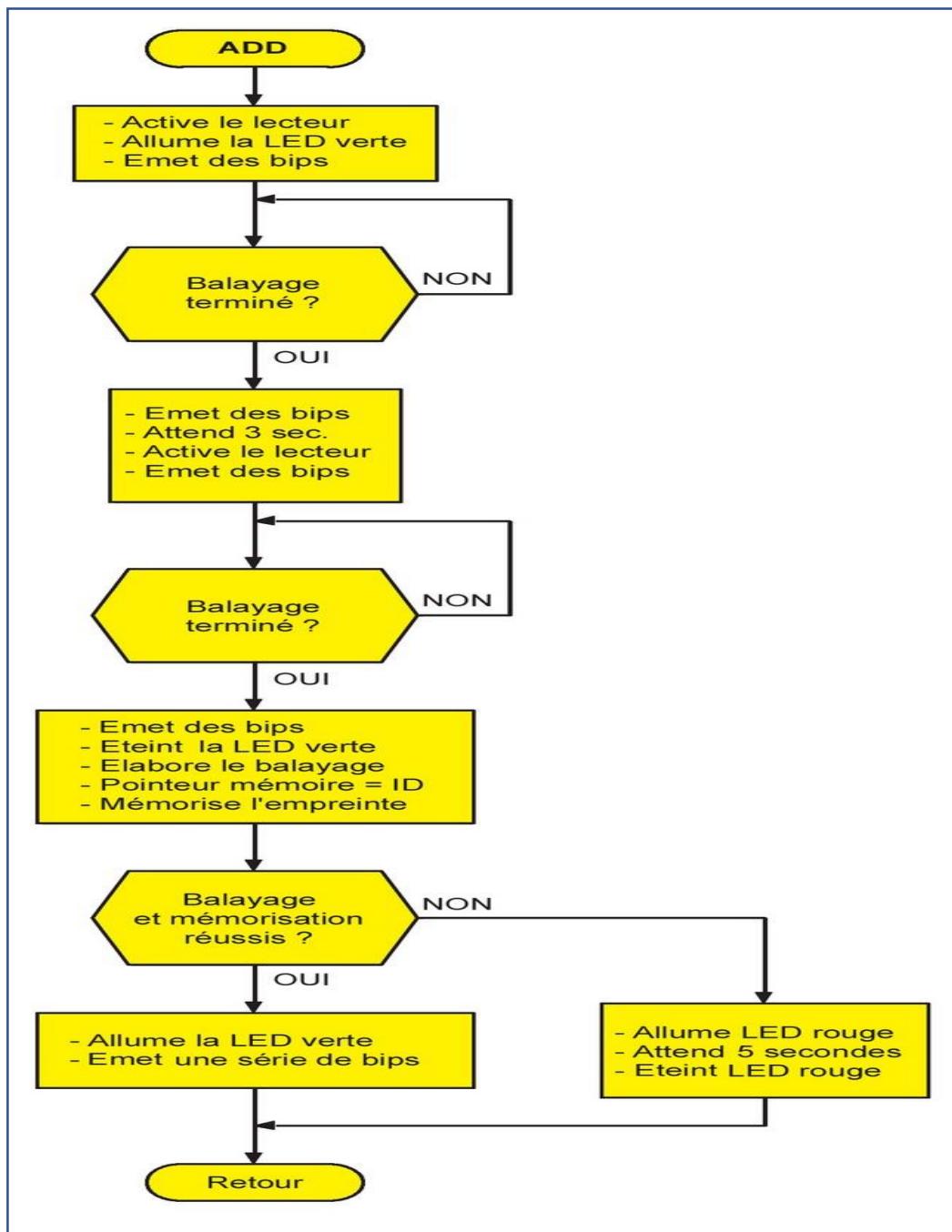


Diagramme de vérification simple de l'empreinte digitale



**Diagramme de vérification complexe de l'empreinte digitale**

Ces diagrammes visualisent les étapes décrites ci-dessus et facilite la compréhension du processus global.

## Méthodes pour transmettre les données

Il existe plusieurs moyens pour envoyer les résultats d'Arduino vers MongoDB :

### a. Transmission via un API REST

#### 1. Crédation d'une API backend :

- ✓ Vous développez une API backend (en Node.js ou React).
- ✓ Arduino envoie les empreintes au serveur via des requêtes HTTP (POST).

#### 2. Exemple de transmission :

- ✓ Arduino formate les données capturées en JSON :

```
{  
    "id_capteur": "CAP123",  
    "empreinte": "010110101010... (binnaire)",  
    "date": "2024-12-17T10:00:00Z",  
    "lieu": "Salle A2"  
}
```

- ✓ Cette requête est envoyée via un module Wi-Fi ou GSM vers l'API backend.

#### 3. Insertion dans MongoDB :

- ✓ L'API backend reçoit les données et les stocke dans MongoDB via un client Node.js, par exemple :

```
import mongoose from 'mongoose';
const Empreinte = mongoose.model('Emprinte', {
  id_capteur: String,
  empreinte: String,
  date: Date,
  lieu: String,
});
const nouvelleEmprinte = new Empreinte({
  id_capteur: "CAP123",
  empreinte: "010110101010...",
  date: new Date(),
  lieu: "Salle A2"
});
await nouvelleEmprinte.save();
```

### b. Transmission directe vers MongoDB

Si utilisation du module plus avancé comme l'ESP32 (capable de gérer des bibliothèques HTTP ou MQTT), Arduino peut :

1. Se connecter directement au serveur MongoDB via une API exposée.
2. Éviter l'utilisation d'un serveur intermédiaire, simplifiant ainsi la transmission des données.

#### II.7.4. Protocoles utilisés

##### 1. Protocole HTTP/HTTPS :

- ✓ Permet à Arduino de communiquer avec un serveur backend (API REST).
- ✓ Les données sont transmises sous forme de requêtes POST.

2. **Protocole MQTT (Message Queuing Telemetry Transport) :**
  - ✓ Idéal pour les réseaux de capteurs distribués.
  - ✓ Arduino envoie les empreintes à un broker MQTT (comme Mosquitto), qui relaye les données vers MongoDB.
3. **Protocole TCP/IP :**
  - ✓ Permet de transmettre les données directement à MongoDB dans des scénarios simples.

#### II.7.5. Intégration

Une vue d'ensemble du fonctionnement

1. **Capture dans une salle de classe :**
  - ✓ Un capteur biométrique connecté à Arduino capture l'empreinte.
2. **Transmission au réseau de capteurs distribués :**
  - ✓ Les résultats sont envoyés via Wi-Fi ou GSM vers un serveur central.
3. **Centralisation dans MongoDB :**
  - ✓ Les empreintes sont stockées dans MongoDB avec des métadonnées (localisation, heure, identifiant du capteur).
4. **Analyse par l'IA :**
  - ✓ MongoDB est utilisé comme source de données pour une IA qui :
    - Compare les empreintes aux modèles existants.
    - Déetecte les anomalies ou les accès non autorisés.
    - Génère une alerte si nécessaire.

#### II.7.6. Exemple d'application réelle

Contrôle d'accès intelligent :

Lorsqu'un utilisateur place son doigt sur le capteur :

1. Arduino capture l'empreinte et l'envoie à MongoDB.
2. MongoDB centralise les données pour comparaison et validation.
3. Si l'empreinte correspond, l'accès est accordé ; sinon, une alerte est générée.

#### II.7.7 Détection précoce des intrusions :

Si une empreinte inconnue ou suspecte est détectée par un capteur :

1. Arduino envoie immédiatement les résultats à MongoDB.
2. L'IA analyse la donnée pour détecter une tentative d'accès illégitime.
3. Une alerte est diffusée via le réseau de capteurs.

**La transmission des résultats Arduino vers MongoDB** est une étape cruciale. Elle assure la centralisation des données capturées par les capteurs distribués et permet leur exploitation par des outils avancés comme l'IA. En combinant Arduino, des protocoles réseau (HTTP, MQTT), et MongoDB, votre système devient un outil puissant pour le contrôle d'accès, la gestion des empreintes, et la génération d'alertes précocees.

**Transmission des informations IA → MongoDB**

La transmission des informations entre une intelligence artificielle (IA) et une base de données MongoDB implique plusieurs étapes, chacune jouant un rôle essentiel dans la collecte, le traitement, et le stockage des données pour une exploitation ultérieure. Dans le contexte de votre projet basé sur les empreintes digitales et les réseaux de capteurs distribués, cette transmission représente la phase où l'IA communique ses résultats (analyses, prédictions, ou décisions) vers MongoDB pour les sauvegarder ou les partager.

---

## II.8 .1. Qu'est-ce que cela signifie ?

1. **IA** : L'intelligence artificielle analyse les empreintes digitales capturées (via Arduino et les capteurs) en utilisant des algorithmes de machine learning ou de deep learning. Par exemple :
  - ✓ Détection de modèles uniques dans les empreintes.
  - ✓ Identification ou vérification d'une personne en comparant l'empreinte avec des données existantes.
  - ✓ Décision sur l'authenticité ou le rejet d'une empreinte.
2. **MongoDB** : Une base de données NoSQL flexible, utilisée pour stocker des informations structurées ou semi-structurées. Elle est idéale pour des données biométriques comme les empreintes digitales, car elle peut stocker des documents JSON contenant des matrices, des images numérisées, et des métadonnées.
3. **Transmission des informations** : Après l'analyse de l'empreinte par l'IA, les résultats (comme la correspondance avec un utilisateur, la validité d'une empreinte, ou des métriques) sont envoyés et sauvegardés dans MongoDB pour une gestion centralisée et une utilisation future.

## II.8.2. Étapes du processus

### a. Capture des données initiales

1. Les empreintes digitales sont capturées via le capteur connecté à l'Arduino.
2. Les données brutes ou partiellement traitées sont transmises à l'IA pour analyse.

### b. Traitement des données avec l'IA

1. L'IA prend les empreintes numériques et exécute des tâches comme :
  - ✓ Extraction des caractéristiques (minuties, crêtes, intersections, etc.).
  - ✓ Comparaison avec un modèle stocké (identification ou vérification).
  - ✓ Classification ou prédition (est-ce une empreinte valide ou non ?).
2. Les résultats générés incluent :
  - ✓ Un identifiant utilisateur.
  - ✓ Des métadonnées (timestamp, précision, fiabilité de la correspondance).
  - ✓ Une décision (accepté/rejeté).

### c. Formatage des résultats

1. L'IA convertit ses résultats en un format lisible pour MongoDB, généralement en **JSON** ou **BSON** (Binary JSON).  
Exemple de document JSON à envoyer :

```
{
  "user_id": "12345",
  "timestamp": "2024-12-17T10:00:00Z",
  "match_confidence": 98.7,
  "decision": "Accepted",
  "fingerprint_data": {
    "minutiae_count": 45,
    "ridge_density": 72
  }
}
```

#### d. Envoi des résultats vers MongoDB

1. Une connexion entre l'IA et MongoDB est établie via une **bibliothèque cliente** (comme Mongoose pour Node.js, PyMongo pour Python).
  2. Les informations formatées sont insérées dans une collection MongoDB.
- Exemple avec PyMongo en Python :

```
from pymongo import MongoClient

# Connexion à MongoDB
client = MongoClient("mongodb://localhost:27017/")
db = client["biometric_system"]
collection = db["fingerprint_results"]

# Document à insérer
result = {
  "user_id": "12345",
  "timestamp": "2024-12-17T10:00:00Z",
  "match_confidence": 98.7,
  "decision": "Accepted",
  "fingerprint_data": {
    "minutiae_count": 45,
    "ridge_density": 72
  }
}
collection.insert_one(result)
```

#### e. Stockage et exploitation des données

1. Les résultats sont maintenant disponibles dans MongoDB. Ils peuvent être :
  - ✓ Consultés par un tableau de bord.
  - ✓ Utilisés pour des analyses statistiques.
  - ✓ Récupérés pour des audits ou des validations ultérieures.

##### II.8.3. Exemple pratique

Contexte : Détection et alerte précoce

1. Les empreintes digitales sont capturées via des capteurs pour identifier des personnes dans des zones sensibles (par exemple, salles de classe ou zones restreintes).
2. L'IA détecte si l'utilisateur est autorisé ou si une alerte doit être déclenchée.

## Transmission des données IA → MongoDB

1. Une fois l'empreinte analysée, l'IA envoie des informations comme :
  - ✓ Identité vérifiée ou non.
  - ✓ Niveau de confiance dans la correspondance.
  - ✓ Alertes générées en cas de tentative d'accès non autorisé.

### II.8.4. Avantages de MongoDB pour votre projet

1. **Flexibilité des données** : MongoDB stocke des documents complexes (ex. empreintes, caractéristiques).
2. **Scalabilité** : Capable de gérer de grandes quantités de données biométriques.
3. **Rapidité des requêtes** : Facilite la recherche rapide dans des millions d'enregistrements d'empreintes.
4. **Sécurité** : Les données biométriques sensibles peuvent être chiffrées.

### II.8.5. Pourquoi cette transmission est essentielle

1. Elle centralise les résultats d'analyse pour une exploitation future.
2. Elle permet de créer un historique des empreintes traitées pour la traçabilité.
3. Elle alimente un système d'alerte qui repose sur les décisions prises par l'IA.

### II.8.6 Conclusion

La transmission IA → MongoDB est cruciale pour gérer efficacement les empreintes numériques, les analyses biométriques, et les décisions de sécurité. MongoDB sert de base centrale pour stocker et organiser les résultats de l'IA, permettant une consultation rapide, une scalabilité, et une intégration aisée avec d'autres systèmes de détection et d'alerte.

## Figure 7. Schéma synoptique PHASE D'ENREGISTREMENT Etapes de traitement

Capture des empreintes digitales (Arduino et capteur biométrique)

- **Matériel utilisé** : Capteur biométrique connecté à une carte **Arduino**.
- **Fonctionnement** :
  - et extraire ces caractéristiques.
  - Les algorithmes peuvent être exécutés sur une plateforme externe où les capacités de calcul sont plus robustes (serveur, ordinateur).
- **Lien avec Arduino** :
  - Arduino collecte l'empreinte digitale, la transmet au module où l'IA est hébergée, puis elle retourne les **détails extraits** à l'Arduino.

### 3. Base de Données (DB)

- **Fonctionnement :**
  - Toutes les caractéristiques biométriques (empreintes digitales) sont **stockées dans une base de données centrale**.
  - Cette base de données peut être hébergée sur un serveur cloud ou une machine locale avec une architecture robuste.
  - **L'IA compare les caractéristiques extraites des nouvelles empreintes digitales avec celles présentes dans la base de données pour une vérification ou une authentification.**
- **Rôle de l'IA :**
  - Des algorithmes d'IA optimisent cette comparaison, assurant une recherche rapide et une correspondance correcte des empreintes, tout en réduisant les faux positifs ou les erreurs de correspondance.

---

### 4. Authentification et Appariement des Données

- **Fonctionnement :**
  - Une fois les caractéristiques extraites, l'IA compare cette empreinte digitale à celles présentes dans la **base de données** pour une correspondance **1:1 ou 1:N**.
  - Si la correspondance est correcte, l'utilisateur est authentifié ; sinon, une **alerte est déclenchée** (tentative d'intrusion, erreur biométrique).
- **Lien Arduino – DB – IA :**
  - Arduino interagit directement avec le capteur biométrique.
  - Les données sont ensuite transmises à l'IA hébergée sur une plateforme externe où l'analyse approfondie est réalisée.
  - La base de données, qui contient des modèles biométriques, est interrogée et comparée via des algorithmes d'IA pour confirmer l'authenticité des utilisateurs.

---

### Résumé des interactions

- **Arduino** capture et envoie l'empreinte digitale.
- **L'IA** traite l'image des empreintes, extrait les caractéristiques et compare les résultats.
- **La base de données** conserve toutes les empreintes digitales et permet leur comparaison rapide.

Ce système complet garantit un **contrôle d'accès fiable**, une **sécurité renforcée** et une authentification rapide en utilisant une combinaison de l'Arduino, des capteurs biométriques, des réseaux IA, et une base de données performante

# Sortie de l'empreinte digitale

## Confirmation d'Identification

### Objectif de la Sortie des Données des Empreintes

Le but principal du système est de confirmer l'identité des étudiants et des professeurs grâce à une analyse des **empreintes biométriques**. Lorsqu'une empreinte digitale est capturée par les capteurs biométriques, elle doit être comparée aux empreintes enregistrées dans votre base de données MongoDB. La sortie doit alors confirmer si l'identité correspond aux informations stockées.

## Structure des Résultats

Lorsque le système compare une empreinte capturée à celles enregistrées, il renvoie des informations structurées qui donnent des résultats clairs et exploitables. Voici comment cela doit être structuré :

### III.1.1. Confirmation des Données

Les résultats des empreintes peuvent être répartis en deux scénarios :

#### 1. Correspondance Confirmée (Identification Confirmée) :

- ✓ Lorsque l'empreinte capturée correspond à une empreinte existante dans la base de données MongoDB.
- ✓ Cela signifie que l'identité de l'utilisateur est **validée**.

#### Exemple des Données de Sortie :

##### 1. Statut : Confirmation réussie

##### 2. Message : "Identification Confirmée"

##### 3. Informations Utilisateur :

- ✓ Nom
- ✓ Identifiant unique (ID)
- ✓ Email
- ✓ Photo de profil
- ✓ Date et heure de la tentative d'identification

#### 2. Correspondance Échouée (Identification Non Confirmée) :

- ✓ Lorsque l'empreinte capturée ne correspond à aucune empreinte enregistrée.
- ✓ Cela peut être dû à une empreinte absente ou une tentative frauduleuse.

#### Exemple des Données :

##### 1. Statut : Échec de Correspondance

##### 2. Message : "Aucun utilisateur correspondant trouvé"

## Exigences Fonctionnelles

### Exactitude des Données

1. Les empreintes biométriques doivent être comparées **avec une grande précision** pour éviter les erreurs d'identification.

#### Temps Réel

- ✓ Les résultats doivent être transmis et affichés **en temps réel**, garantissant une réponse instantanée.

#### Sécurisation des Données

- ✓ Toutes les informations personnelles et biométriques doivent être **chiffrées et protégées**, conformément aux standards de sécurité.

#### Interactions avec MongoDB

MongoDB jouera un rôle essentiel dans :

1. **Stockage des Empreintes Biométriques**
  - ✓ Chaque utilisateur aura une **empreinte digitale enregistrée avec ses informations personnelles** (nom, email, photo, etc.).
2. **Correspondance des Empreintes**
  - ✓ Le système compare une empreinte capturée avec celles existantes dans la collection MongoDB.
  - ✓ Si une correspondance est trouvée, une réponse positive est renvoyée.

#### III.1.4 Résumé des Sorties des Données

Paramètre	Valeur
<b>Statut</b>	success / failure
<b>Message</b>	Confirmation d'identité
<b>Nom</b>	Nom de l'étudiant/professeur
<b>Identifiant (ID)</b>	Identifiant unique
<b>Email</b>	Adresse email
<b>Photo de Profil</b>	Lien vers l'image
<b>Date et Heure</b>	Horodatage des tentatives

#### III.1.4 Conclusion

Dans ce système ENTACIG, l'**analyse des empreintes biométriques doit permettre une confirmation rapide et précise des identités des utilisateurs**. MongoDB assurera un stockage robuste et rapide des données

biométriques, tandis que l'interface utilisateur React affichera clairement les résultats des correspondances pour permettre une validation visuelle et en temps réel des confirmations d'identité.

## Génération des Résultats Authentifiés

### Objectif du Système

Le système d'ENTACIG, l'**objectif de la génération des résultats authentifiés pour les empreintes digitales** est de confirmer et valider l'identité des **étudiants, professeurs et cadres** présents dans une salle de classe. En comparant l'empreinte digitale capturée en temps réel avec celles enregistrées dans votre base de données MongoDB, votre système doit assurer une **identification rapide et fiable** des utilisateurs

## Génération des Résultats Authentifiés

### Objectif du Système

Le système d'ENTACIG, l'**objectif de la génération des résultats authentifiés pour les empreintes digitales** est de confirmer et valider l'identité des **étudiants, professeurs et cadres** présents dans une salle de classe. En comparant l'empreinte digitale capturée en temps réel avec celles enregistrées dans votre base de données MongoDB, votre système doit assurer une **identification rapide et fiable** des utilisateurs.

### Vue d'Ensemble des Résultats Authentifiés pour Empreintes Digitales

Les résultats authentifiés sont des **confirmations précises et sécurisées des identités** basées sur des correspondances d'empreintes digitales. Cette authentification peut être utilisée pour :

1. **Contrôle d'accès à la salle de classe.**
2. **Validation des présences.**
3. **Participation aux activités et examens.**
4. **Authentification des droits d'accès pour les professeurs et étudiants.**

### Composants du Système pour la Génération des Résultats Authentifiés

#### III.2.1. Capteurs Biométriques (Empreintes digitales)

Les capteurs installés dans la salle capturent l'**empreinte digitale** des utilisateurs. Ils transmettent ensuite ces informations au système pour correspondance avec les empreintes enregistrées.

#### Base de Données MongoDB

Les empreintes digitales et les informations personnelles sont stockées dans une collection MongoDB.

#### Exemple des enregistrements :

```
{  
  "_id": "unique_id",  
  "nom": "Moussa BERETE",  
  "email": "beretemoussa@gmail.com",
```

```

"identifiant": "12345",
"empreinte": {
  "pattern": "empreinte_unique_digital"
}
}

```

## Serveur Node.js

Il reçoit les empreintes capturées, les compare avec celles de MongoDB et génère les résultats authentifiés.

## . Interface React (Frontend)

L'interface affiche les résultats authentifiés, comme la confirmation des identités, les photos et autres informations.

## Résultats Authentifiés pour Empreinte Digitale

Lorsque le système identifie une empreinte digitale, il renvoie une **réponse structurée qui contient toutes les informations nécessaires** pour confirmer l'identité.

## Exemple de Réponse Authentifiée JSON

```
{
  "status": "success",
  "message": "Authentification Confirmée",
  "userDetails": {
    "nom": "Moussa BERETE",
    "identifiant": "12345",
    "email": "beroussa@gmail.com",
    "photo": "url_photo",
    "timestamp": "2024-09-25T14:20:00"
  }
}
```

## Analyse des Champs

Paramètre	Signification
status	success : Identification réussie, failure : Identification échouée.
message	Confirmation de l'identification. Exemple : "Authentification Confirmée".
identifiant	Un identifiant unique pour chaque utilisateur.
nom	Nom et prénom de l'utilisateur authentifié.
email	L'adresse email associée à l'utilisateur.
photo	L'URL pointant à la photo authentifiée.
timestamp	Horodatage des tentatives d'authentification.

## Sécurité des Résultats Authentifiés

Les résultats générés garantissent :

1. **Chiffrement des Empreintes :**  
Les empreintes digitales sont **cryptées**, assurant leur protection totale.
2. **Authentification en Temps Réel :**  
Les correspondances des empreintes sont réalisées instantanément, assurant une **réponse en temps réel**.
3. **Protection des Données Personnelles :**  
Seules les informations nécessaires sont affichées, tout en garantissant la **confidentialité des données personnelles** des utilisateurs.

### Étapes de Correspondance et Génération des Résultats

#### 1. Capture des Empreintes Digitales

Les capteurs installés dans la salle capturent les empreintes digitales des étudiants, professeurs et cadres.

#### 2. Transmission des Données au Système Backend

Les empreintes capturées sont transmises via l'API Node.js pour être comparées avec celles de MongoDB.

#### 3. Correspondance des Empreintes

Le système backend recherche l'empreinte capturée dans la collection MongoDB.  
Si une correspondance est trouvée, l'identité est confirmée.

#### 4. Génération des Résultats Authentifiés

Les résultats sont structurés et renvoyés au frontend React.

Ces informations sont affichées à l'utilisateur en temps réel, garantissant la transparence et la précision.

### Résumé des Avantages des Résultats Authentifiés

Avantages	Description
Rapidité de Réponse	Résultats disponibles en temps réel.
Sécurisation des Données	Les empreintes sont chiffrées et protégées.
Précision des Identités	Identification fiable des professeurs et étudiants.
Interface Clarity	Vue utilisateur intuitive et facile à comprendre.

### Conclusion

Dans le projet ENTACIG, la **génération des résultats authentifiés des empreintes digitales** doit offrir une validation rapide et sécurisée des identités des utilisateurs. MongoDB assure un stockage solide des données, tandis que React et Node.js garantissent une transmission fluide des informations authentifiées. Le système

respecte des standards de sécurité élevés, assurant la confidentialité des utilisateurs tout en maintenant des performances optimales.

### III.3. Stockage des empreintes biométriques sur MongoDB

#### III.3.1 Contexte Général du Projet

Un **système de détection et d'alerte précoce basé sur des réseaux de capteurs distribués** qui intègrent des capteurs biométriques pour assurer la surveillance, la détection et l'authentification des individus. MongoDB joue un rôle crucial dans le stockage et l'organisation des données biométriques collectées. Cela garantit la rapidité et la précision des correspondances des empreintes digitales tout en assurant la sécurité et la confidentialité des informations sensibles.

#### III.3.2 Objectif

L'objectif du stockage des empreintes biométriques sur MongoDB est :

1. **Enregistrer et stocker** les empreintes digitales des utilisateurs (étudiants, professeurs, cadres) recueillies par les capteurs biométriques.
2. Assurer une **correspondance rapide et précise** des identités avec des requêtes efficaces.
3. Garantir la **sécurisation des informations biométriques** pour éviter tout accès non autorisé.

#### III.3.3 Structure des Données sur MongoDB pour le Stockage des Empreintes Biométriques

Les empreintes digitales sont généralement stockées sous une **forme hachée ou compressée**, garantissant ainsi que seules les correspondances sont possibles et non une reconstitution des informations biométriques.

##### Exemple des Champs du Document MongoDB

```
{  
  "_id": ObjectId("unique_id"),  
  "nom": "Moussa BERETE",  
  "identifiant": "12345",  
  "email": "beroussa@gmail.com",  
  "role": "étudiant",  
  "empreinte": {  
    "pattern": "empreinte_hachee_unique",  
    "type": "hash_sha256"  
  },  
  "timestamp": "2024-09-25T14:20:00"  
}
```

### Analyse des Champs

Paramètre	Description
_id	Identifiant unique généré automatiquement par MongoDB.
nom	Nom et prénom de l'utilisateur authentifié.
identifiant	Identifiant unique qui distingue chaque utilisateur.
email	Adresse email de l'utilisateur.
role	Rôle de l'utilisateur (professeur, étudiant, cadre).
empreinte	Données biométriques hachées.
timestamp	Date et heure où l'empreinte a été collectée et enregistrée.

## Sécurisation des Données Biométriques

Les empreintes biométriques sont des informations sensibles. Il est donc impératif de les sécuriser en utilisant des méthodes robustes.

### Méthodes de Sécurisation

#### 1. Hachage SHA-256

- ✓ Chaque empreinte digitale est transformée en une **chaîne de caractères unique** en utilisant des algorithmes de hachage comme **SHA-256**.
- ✓ Cela empêche la reconstitution des empreintes biométriques.

#### 2. Chiffrement AES-256

- ✓ Pour une protection supplémentaire, des techniques de **chiffrement symétrique AES-256** sont utilisées.
- ✓ Seul le serveur possède la clé nécessaire pour déchiffrer ces empreintes.

### Processus de Stockage des Empreintes Biométriques

#### 1□ Capture des Empreintes Biométriques

Les capteurs biométriques installés dans votre environnement de salle collectent **en temps réel les empreintes digitales des individus** (étudiants, professeurs).

- ✓ Les capteurs biométriques génèrent une empreinte numérique ou hachée.
- ✓ Les empreintes sont immédiatement transmises au backend pour traitement.

## 2 □ Transmission des Données au Backend (Node.js)

- ✓ Le backend (Node.js) reçoit les empreintes et utilise des fonctions de **hachage (SHA-256)** et de chiffrement pour assurer la confidentialité.
- ✓ Les empreintes sont ensuite organisées dans des documents MongoDB.

## 3 □ Enregistrement des Données dans MongoDB

Le serveur Node.js utilise **Mongoose**, un package pour MongoDB, pour sauvegarder les informations.

### Exemple de Code pour le Stockage des Empreintes sur MongoDB

```
import mongoose from 'mongoose';

// Connexion à MongoDB
mongoose.connect('mongodb://localhost:27017/ENTACIG', {
  useNewUrlParser: true,
  useUnifiedTopology: true
});

// Schéma Mongoose pour l'utilisateur
const userSchema = new mongoose.Schema({
  nom: String,
  identifiant: String,
  email: String,
  role: String,
  empreinte: {
    pattern: String,
    type: String
  },
  timestamp: { type: Date, default: Date.now }
});

const User = mongoose.model('User', userSchema);

// Fonction pour enregistrer une empreinte digitale
const saveUserFingerprint = async (user) => {
  try {
    const newUser = new User(user);
    await newUser.save();
    console.log('Les empreintes ont été correctement sauvegardées.');
  } catch (error) {
    console.error('Erreur lors de l'enregistrement des empreintes :', error);
  }
};

// Exemple des informations utilisateur
saveUserFingerprint({
  nom: "Moussa BERETE",
  identifiant: "12345",
  email: "beremoussa@gmail.com",
  role: "étudiant",
  empreinte: {
    pattern: "unique_hache_sha256",
    type: "hash_sha256"
  }
});
```

## Avantages du Stockage des Empreintes Biométriques sur MongoDB

Avantage	Description
Scalabilité	MongoDB gère efficacement de grands volumes de données.
Sécurité	Hachage et chiffrement assurent la protection des empreintes biométriques.
Flexibilité des Documents	Les schémas dynamiques facilitent l'extension des attributs.
Performances Optimisées	Requêtes rapides et correspondance des empreintes en temps réel.

## **Conclusion**

Le stockage des empreintes biométriques sur MongoDB garantit :

1. Une **correspondance rapide des identités** en temps réel.
2. La **sécurisation des empreintes biométriques**, empêchant tout accès non autorisé.
3. Une **extensibilité et une performance optimisée** pour un système basé sur des capteurs distribués.

MongoDB est parfaitement adapté pour un environnement robuste et en temps réel où des données biométriques critiques sont collectées, stockées, et analysées.

### III.4 Transmission des Confirmations de Validation sur l'Interface Utilisateur

#### **Contexte Général**

Dans le **système de détection et d'alerte précoce avec réseaux de capteurs distribués**, il est essentiel d'afficher des confirmations sur l'interface utilisateur après chaque validation d'authentification des empreintes biométriques. Cela assure une meilleure **expérience utilisateur**, la **transparence des actions**, et la **confiance dans la détection des correspondances biométriques**.

Les confirmations peuvent être sous la forme de messages informatifs, notifications, ou confirmations visuelles pour assurer à l'utilisateur que la validation a réussi.

#### **Objectif**

1. **Informer l'utilisateur en temps réel** des résultats de la correspondance des empreintes biométriques.
2. Assurez la **transparence des actions** et des décisions du système.
3. Fournir des messages de confirmation clairs, concis et informatifs.

#### **III.4.1 Processus Général**

##### 1□ Capture et Validation des Empreintes

Lorsque les capteurs collectent l'empreinte biométrique :

1. **Comparaison des Empreintes** :

- Les empreintes biométriques sont comparées avec les empreintes existantes dans la base de données MongoDB.
- Le système détermine si l'empreinte est authentifiée, non reconnue, ou pose une autre validation.

## 2 □ Transmission des Résultats au Backend (Node.js)

- Le **backend Node.js** renvoie des réponses après avoir comparé les empreintes.
- Le serveur confirme à l'interface utilisateur si une empreinte est **authentifiée avec succès** ou non.
- Les résultats sont alors transmis via une **API REST**, **Socket.io** (pour la communication en temps réel), ou des **requêtes AJAX**.

## 3 □ Transmission des Résultats à l'Interface Utilisateur (React + TailwindCSS)

- Les messages sont reçus côté **frontend avec React**.
- Des composants React comme des **pop-ups, alertes, ou notifications**, affichent les résultats.
- Des confirmations visuelles en utilisant **TailwindCSS** assurent une bonne esthétique et clarté des messages.

### Exemple des Différents Messages de Confirmation

Résultat	Message Transmis à l'Interface
<b>Authentification Réussie</b>	"Validation confirmée : empreinte authentifiée avec succès."
<b>Empreinte non reconnue</b>	"Erreur : empreinte biométrique introuvable."
<b>Problème Technique</b>	"Problème avec le capteur. Veuillez réessayer."

### Exemple d'Implémentation Côté Frontend

#### Composant React avec Notifications

```
import React, { useState } from "react";

const ConfirmationMessage = ({ message, type }) => {
  return (
    <div className={`p-4 rounded mt-4 ${type === 'success' ? 'bg-green-200' : type === 'error' ? 'bg-red-200' : 'bg-yellow-200'}`}>
      {message}
    </div>
  );
};

const UserInterface = () => {
  const [confirmation, setConfirmation] = useState('');

  const handleResponse = async () => {
    const response = await fetch('/api/validateFingerprint');
    const data = await response.json();

    if (data.status === 'success') {
      setConfirmation("Validation confirmée : empreinte authentifiée avec succès.");
    } else if (data.status === 'error') {
      setConfirmation("Erreur : empreinte biométrique introuvable.");
    }
  };
};
```

```

        }
    };

    return (
        <div>
            <button onClick={handleResponse} className="p-2 bg-blue-500 text-white rounded mt-5">
                Valider l'empreinte
            </button>
            {confirmation && <ConfirmationMessage message={confirmation} type="success" />}
        </div>
    );
};

export default UserInterface;

```

## Interface Stylée avec TailwindCSS

Les messages sont stylés pour assurer une bonne visibilité :

```

.bg-green-200 {
    background-color: #a7f3c7;
}

.bg-red-200 {
    background-color: #fca5a5;
}

.bg-yellow-200 {
    background-color: #fde68a;
}

.rounded {
    border-radius: 8px;
}

.p-4 {
    padding: 16px;
}

```

### Sécurisation des Confirmations

Les messages sont :

1. **Transmis via HTTPS** pour assurer l'intégrité des requêtes.
2. Utilisent des **JSON Web Tokens (JWT)** pour valider l'authenticité des requêtes côté backend.

### Conclusion

La transmission des confirmations biométriques sur l'interface React assure :

1. Une expérience utilisateur fluide et réactive.
2. Une communication sécurisée entre **frontend** et **backend**.
3. Des messages informatifs et visuels clairs garantissant la transparence des décisions du système.

L'intégration des messages de validation en temps réel optimise ainsi la réactivité et la robustesse de ce projet basé sur des capteurs biométriques distribués et l'IA.

# Caméra de surveillance (Capteur de Mouvement)



## Introduction Générale

Dans un monde où la sécurité devient une préoccupation majeure, les environnements éducatifs tels que les salles de classe doivent être dotés de technologies modernes pour garantir la protection des personnes et des biens. Les systèmes de surveillance traditionnels, bien qu'efficaces dans certaines situations, présentent des limites importantes. Ils se contentent souvent d'enregistrer des vidéos sans offrir de capacité d'analyse intelligente ni de réaction immédiate en cas d'incident.

## Contexte et problématique

Les salles de classe, en tant qu'espaces où se concentrent les étudiants et les enseignants, nécessitent des systèmes de surveillance qui vont au-delà de la simple observation passive. Les défis actuels incluent :

1. **L'absence de traitement intelligent des données captées** : Les systèmes conventionnels ne permettent pas de différencier une activité normale d'une activité suspecte, ce qui peut conduire à des faux positifs ou des incidents non détectés.
2. **L'absence de réponse en temps réel** : Dans les situations critiques, comme une intrusion ou un comportement inhabituel, le temps de réaction est souvent trop long, ce qui peut aggraver les conséquences de l'incident.

Face à ces problématiques, ce projet vise à concevoir et à mettre en œuvre un système de surveillance moderne pour une salle de classe, basé sur des technologies avancées telles que :

1. **L'intelligence artificielle (IA)** pour analyser les données en temps réel et détecter des comportements anormaux.
2. **L'utilisation d'Arduino** pour collecter les données des capteurs et piloter les actions automatisées.
3. **Un système de notification rapide** pour alerter les responsables en cas de situation critique.

## Objectifs du projet

Dans le but de remédier aux limites des systèmes de surveillance traditionnels, ce projet ambitionne de mettre en place une solution de surveillance intelligente adaptée aux environnements éducatifs. Les objectifs principaux incluent :

1. Automatisation de la surveillance avec IA : Utiliser des algorithmes d'intelligence artificielle pour analyser les données en temps réel et détecter des événements inhabituels, tels qu'un intrus ou un comportement anormal dans la salle de classe.

2. Intégration d'Arduino : Exploiter les capacités des modules Arduino pour capter des données provenant de différents capteurs (mouvement, son, température, etc.) et déclencher des alertes automatiques en cas de besoin.

Grâce à cette automatisation, le système permettra de réduire les interventions humaines tout en assurant une surveillance fiable et constante.

## Présentation générale

Le système proposé repose sur une combinaison de technologies modernes et robustes qui assurent une surveillance proactive :

1. Technologies utilisées :
  - ✓ Caméras : Capturent des images et vidéos haute définition pour une surveillance visuelle.
  - ✓ Capteurs : DéTECTEURS de mouvement et de son intégrés pour identifier les activités suspectes et les anomalies.
  - ✓ Intelligence artificielle : Analyse des flux vidéo et des données des capteurs pour reconnaître les comportements inhabituels.
  - ✓ MongoDB : Gestion des données capturées, permettant leur stockage sécurisé et leur accès rapide pour les analyses et historiques.
2. Avantages du système :
  - ✓ Sécurité renforcée : Une identification et une gestion rapide des incidents grâce à des notifications en temps réel.
  - ✓ Surveillance proactive : Le système n'attend pas une intervention humaine pour réagir ; il prend automatiquement des décisions, comme l'envoi d'alertes ou l'activation d'alarmes, en cas de détection d'un événement anormal.

## Capteurs et Caméras

### Types de Caméras

Dans un système de surveillance, différents types de caméras sont utilisés pour répondre à des besoins spécifiques. Les caméras sont choisies en fonction des caractéristiques de l'environnement, des conditions de luminosité et des objectifs de surveillance. Quelques les principaux types de caméras utilisés dans les systèmes modernes :

#### 1. Caméras numériques (IP Cameras)

Les caméras numériques, également appelées caméras IP (Internet Protocol), sont des dispositifs de surveillance connectés à un réseau informatique. Contrairement aux caméras analogiques, les caméras numériques capturent des images sous forme de fichiers numériques, qui peuvent être facilement stockés, analysés et partagés via des systèmes informatiques. Elles offrent une haute résolution, ce qui permet de capturer des images détaillées, et peuvent être accédées à distance à travers Internet, ce qui les rend particulièrement adaptées aux environnements modernes nécessitant une surveillance à distance. Ces caméras sont capables de fournir des vidéos en haute définition (HD), voire en ultra-haute définition (4K), offrant ainsi une clarté d'image qui facilite l'identification d'individus ou d'événements suspects.

#### 2. Caméras infrarouges (IR Cameras)

Les caméras infrarouges sont spécifiquement conçues pour fonctionner dans des environnements à faible luminosité, ou même dans l'obscurité totale. Elles utilisent la technologie infrarouge pour détecter et capturer des images basées sur la chaleur émise par les objets dans leur champ de vision. Ces caméras peuvent être particulièrement utiles dans des environnements où les éclairages sont faibles ou inexistant, car elles peuvent capter la chaleur émise par les personnes ou les objets dans l'obscurité.

comme dans des parkings souterrains, des entrepôts ou des zones extérieures la nuit. Les caméras IR sont équipées de LEDs infrarouges qui illuminent la scène sans être visibles à l'œil nu, permettant ainsi à la caméra de produire une image claire, même dans l'obscurité complète.

### Vision Nocturne et Détection Thermique

En complément de la technologie infrarouge, certaines caméras modernes sont également équipées de **vision nocturne**. La vision nocturne se base sur des capteurs qui captent la lumière disponible (même très faible) et l'amplifient, permettant à la caméra de produire une image visible, même dans des conditions de faible luminosité. Ces caméras sont souvent utilisées dans les systèmes de surveillance de sécurité, car elles peuvent fonctionner 24h/24, de jour comme de nuit, en offrant une visibilité continue.

Les caméras infrarouges sont également utilisées dans des systèmes de **détection thermique**, qui mesurent la chaleur corporelle des objets et des personnes. Ces caméras ne capturent pas des images visibles, mais des représentations thermiques (ou cartes thermiques), montrant la température des objets dans le champ de vision. La détection thermique peut être utilisée pour identifier des intrus qui se déplacent dans des zones non éclairées ou pour détecter des anomalies thermiques (comme des incendies ou des équipements défectueux).

### Caméras PTZ (Pan-Tilt-Zoom)

Les caméras PTZ sont des caméras motorisées qui peuvent pivoter horizontalement (pan), s'incliner verticalement (tilt), et zoomer sur des zones spécifiques de manière contrôlable à distance. Ces caméras sont idéales pour les environnements vastes, comme les parkings, les entrepôts ou les espaces publics, où il est nécessaire de suivre un individu ou de zoomer sur une zone précise pour obtenir plus de détails.

### Caméras 360° ou Caméras omnidirectionnelles

Les caméras à 360° ou omnidirectionnelles permettent de capturer des images panoramiques de toute la zone qui les entoure, sans angle mort. Ces caméras sont souvent utilisées pour les zones ouvertes ou dans des espaces où une couverture complète est nécessaire. Elles permettent de réduire le nombre de caméras requises pour une surveillance efficace.

### Capteurs de Mouvement

Outre les caméras, les capteurs de mouvement sont essentiels pour la détection des événements. Ces capteurs utilisent diverses technologies pour détecter des changements dans leur environnement, par exemple :

1. **Capteurs à infrarouge passif (PIR)** : Ces capteurs détectent les variations de chaleur (mouvement d'une personne ou d'un animal) en mesurant les infrarouges émis par le corps humain. Ils sont souvent utilisés pour activer des caméras de surveillance ou des systèmes d'alarme lorsque quelqu'un pénètre dans une zone surveillée.
2. **Capteurs à ultrasons et micro-ondes** : Ces capteurs détectent le mouvement en envoyant des ondes sonores ou des ondes radio qui se réfléchissent sur les objets en mouvement, permettant de détecter des changements dans l'environnement.

### Importance des Caméras et Capteurs dans la Détection

Les caméras et capteurs sont des éléments clés pour détecter les événements suspects dans un environnement surveillé. En associant la vidéo en temps réel, la détection thermique, la vision nocturne et les capteurs de mouvement, les systèmes de surveillance modernes peuvent non seulement fournir une image claire mais aussi activer des alertes instantanées dès qu'un comportement suspect ou un événement non autorisé est détecté.

L'association de ces technologies avec des systèmes d'IA (intelligence artificielle) permet de créer des systèmes de surveillance proactifs, capables d'analyser en temps réel les images capturées et de prendre des décisions instantanées, telles que l'envoi d'alertes ou le déclenchement d'alarme, sans intervention humaine

### **Capteurs de Mouvement et de Son** : détection d'activités inhabituelles.

Les **capteurs de Mouvement et de Son** jouent un rôle crucial dans la détection d'activités inhabituelles dans un système de surveillance, en particulier dans des environnements comme une salle de classe ou un bâtiment. Ces capteurs permettent de repérer en temps réel des changements dans l'environnement, d'identifier des comportements suspects ou de réagir à des événements qui nécessitent une attention particulière. Leur utilisation est essentielle pour compléter les systèmes de vidéosurveillance en ajoutant une couche supplémentaire de réactivité et de précision.

#### **Capteurs de Mouvement**

Les **capteurs de mouvement** sont des dispositifs électroniques conçus pour détecter les mouvements physiques dans une zone donnée. En intégrant ces capteurs à un système de surveillance, on peut détecter des intrusions, des changements d'activité ou des comportements anormaux qui nécessitent une analyse plus approfondie.

#### **1 Types de Capteurs de Mouvement**

Il existe plusieurs types de capteurs de mouvement, chacun ayant une technologie de détection spécifique.

##### **1. Capteurs à Infrarouge Passif (PIR - Passive Infrared Sensor)**

- ✓ Les capteurs PIR sont les plus répandus et fonctionnent en détectant les changements dans les niveaux de chaleur. Ils détectent la chaleur émise par les corps humains ou animaux qui se déplacent à proximité. Lorsqu'un objet ou une personne entre dans le champ de détection, il génère une variation dans les infrarouges, ce qui déclenche l'activation du capteur.
- ✓ **Avantages :** Faible consommation d'énergie, réactivité rapide, faible coût.
- ✓ **Limites :** Ne détecte que les mouvements de personnes ou d'animaux et est sensible aux changements de température dans l'environnement.

##### **2. Capteurs à Micro-ondes**

- ✓ Ces capteurs utilisent des ondes radio (micro-ondes) pour détecter le mouvement. Ils émettent des ondes et mesurent la réflexion de ces ondes lorsqu'un objet se déplace dans la zone de détection. Ils sont plus sensibles que les capteurs PIR et peuvent détecter des mouvements même à travers des murs ou des cloisons légères.
- ✓ **Avantages :** Sensibilité élevée, détection à travers des obstacles légers.
- ✓ **Limites :** Ils peuvent être plus sensibles aux interférences électromagnétiques et sont plus coûteux.

##### **3. Capteurs à Ultrasons**

- ✓ Ces capteurs utilisent des ondes sonores à haute fréquence pour détecter les objets en mouvement. En envoyant des impulsions sonores et en mesurant leur réflexion, ils peuvent créer une carte de l'environnement autour du capteur. Ils sont souvent utilisés dans des environnements où la détection de petits mouvements ou des variations subtiles est nécessaire.
- ✓ **Avantages :** Détection précise, faible coût.
- ✓ **Limites :** Moins efficaces à grande distance et peuvent être perturbés par des bruits ambients.

#### **Application dans la Détection d'Activités Inhabituelles**

Les capteurs de mouvement sont utilisés pour détecter toute forme de déplacement dans une zone donnée. Dans un contexte de surveillance, ces capteurs peuvent signaler des événements inhabituels comme :

1. **Intrusion non autorisée** : Détection de personnes entrant dans des zones restreintes ou protégées.
2. **Comportement suspect** : Par exemple, une personne qui se déplace de manière anormale ou rapide dans un espace.
3. **Anomalies de circulation** : Détection d'un grand nombre de personnes se déplaçant de manière irrégulière, comme une foule ou un attroupement dans un endroit non prévu.

Lorsqu'un mouvement est détecté, le capteur peut envoyer une alerte au système central qui, en fonction des algorithmes d'IA intégrés, peut décider d'activer d'autres fonctions, comme une caméra de surveillance pour enregistrer des images, ou une alarme sonore pour alerter en temps réel.

### Capteurs de Son : Détection d'Activités Inhabituelles

Les **capteurs de son** sont des dispositifs électroniques qui captent les sons dans leur environnement et détectent des bruits ou des sons inhabituels. Ils sont souvent utilisés pour compléter la surveillance par vidéo ou par mouvement en offrant une détection basée sur les sons de l'environnement.

#### ▪ Types de Capteurs de Son

Les capteurs de son les plus utilisés sont principalement des microphones sensibles et des détecteurs spécialisés dans des bruits spécifiques. Ces capteurs sont capables de détecter des sons dans un large éventail de fréquences et d'intensités.

##### 1. Microphones sensibles à l'environnement

- ✓ Ces microphones sont capables d'enregistrer des sons dans une large gamme de fréquences. Ils sont capables de détecter des bruits comme des voix, des pas, des bruits métalliques ou des bruits de chocs. Les microphones directionnels ou les microphones à haute sensibilité peuvent être utilisés pour capter des sons provenant de sources spécifiques.
- ✓ **Avantages** : Détection fine des bruits inhabituels, efficacité pour détecter des événements spécifiques comme des voix, des bruits de pas, des cris.
- ✓ **Limites** : Sensibilité aux bruits ambients et aux interférences.

##### 2. Détecteurs de bruits spécifiques (Ex : Verrouillage, bris de vitre)

- ✓ Ces capteurs sont conçus pour détecter des sons très spécifiques, comme des bruits de bris de verre ou des bruits de porte forcée. Ils sont capables de filtrer les bruits ambients pour ne se concentrer que sur des sons très précis.
- ✓ **Avantages** : Très efficaces pour la détection d'événements inhabituels ou dangereux (par exemple, un vol par effraction).
- ✓ **Limites** : Ne détectent qu'un type très spécifique de bruit.

#### ▪ Application dans la Détection d'Activités Inhabituelles

Les capteurs de son peuvent être utilisés pour détecter une série d'événements inhabituels ou suspects :

1. **Effraction** : Détection de bruits de bris de vitre, de porte ou de fenêtre forcée.
2. **Comportement suspect** : Par exemple, des voix élevées, des cris ou des disputes peuvent être détectés et signalés.
3. **Bris d'équipement ou chocs** : Des bruits de chocs ou de chutes peuvent signaler un accident ou un vol dans un environnement surveillé.

Les capteurs de son sont souvent couplés avec des caméras ou des capteurs de mouvement pour offrir une réponse plus précise. Par exemple, un capteur de son peut détecter un bruit de bris de vitre, tandis qu'une caméra à proximité peut immédiatement fournir une image pour confirmer l'intrusion.

## Combinaison des Capteurs de Mouvement et de Son pour une Détection Complète

L'intégration des **capteurs de mouvement** et **de son** dans un système de surveillance intelligent permet de maximiser l'efficacité de la détection d'activités inhabituelles. Ces capteurs peuvent fonctionner ensemble pour détecter des événements de manière plus précise et pour réduire les faux positifs. Par exemple :

1. Si un capteur de mouvement détecte une intrusion, un capteur de son peut être activé pour écouter des bruits suspects comme des cris ou des bris de verre.
2. Un capteur de son qui détecte un bruit suspect peut activer un capteur de mouvement pour une analyse plus approfondie de la situation.

L'intégration avec des algorithmes d'**intelligence artificielle** peut aussi être utilisée pour analyser en temps réel les données des capteurs et déterminer si un événement est effectivement suspect. Par exemple, un bruit de pas détecté par un capteur de son combiné avec un mouvement dans une zone restreinte peut être immédiatement interprété comme une tentative d'intrusion.

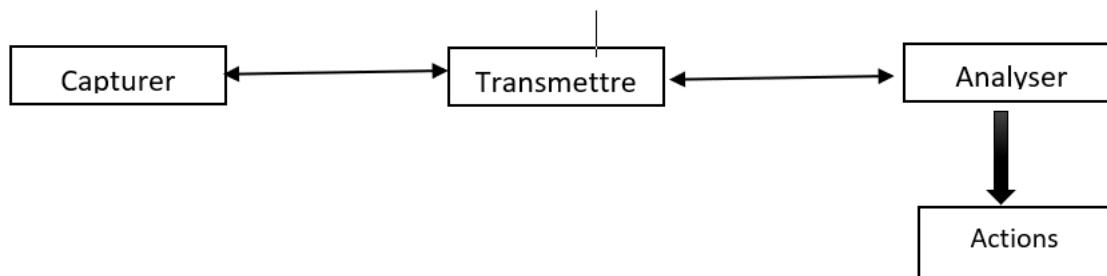
### Conclusion

Les **capteurs de mouvement** et **de son** sont des composants essentiels dans les systèmes modernes de surveillance. Leur capacité à détecter des anomalies dans leur environnement permet une détection précoce des événements inhabituels, ce qui est crucial pour la sécurité. Couplés avec des technologies avancées comme l'IA et l'analyse vidéo, ces capteurs peuvent offrir une surveillance proactive et efficace, réduisant ainsi les risques d'incidents non détectés.

### Modules Matériels Arduino : Capture et Transmission des Données Capturées par les Capteurs

Les **modules matériels Arduino** jouent un rôle fondamental dans la gestion des capteurs dans un système de surveillance domotique. Ils permettent de capturer les données fournies par les capteurs (mouvement, son, température, etc.) et de les transmettre à un système central pour analyse. L'Arduino est une plateforme ouverte et flexible qui est souvent utilisée dans les projets de domotique et d'automatisation en raison de sa simplicité et de sa capacité

### Schéma Synoptique



## Schéma Synoptique

Le schéma Synoptique de la détection est un processus en trois étapes clé qui consiste à **capturer les données** (images, sons, mouvements), à les **transmettre** pour traitement, et à les **analyser** pour identifier des événements inhabituels ou des comportements anormaux. Ce processus est crucial pour un système de surveillance intelligent, que ce soit dans un environnement éducatif, comme une salle de classe, ou dans d'autres environnements nécessitant une surveillance continue.

Voici une explication détaillée de chacune des étapes, en particulier dans le contexte d'une **salle de classe**.

### 1. Capturer (Capture des données)

La première étape consiste à **capturer les données** en temps réel à l'aide de divers dispositifs de surveillance. Dans un environnement comme une salle de classe, la capture des données se fait principalement par les **caméras de surveillance**, mais d'autres capteurs peuvent également être utilisés pour améliorer la couverture.

#### 1. Types de dispositifs utilisés pour la capture :

##### 1. Caméras vidéo :

1. **Caméras numériques** pour capturer des images claires et détaillées.
2. **Caméras infrarouges** pour permettre la vision nocturne ou dans des environnements peu éclairés, permettant de capturer des images même lorsqu'il n'y a pas de lumière ambiante.
3. **Caméras PTZ (Pan-Tilt-Zoom)** : Ces caméras peuvent pivoter (pan), s'incliner (tilt) et zoomer (zoom) pour couvrir une zone plus large et suivre des objets en mouvement.

2. **Capteurs de mouvement** : Ces capteurs détectent tout déplacement dans une zone spécifique, comme près des **entrées** et **fenêtres**. Lorsqu'un mouvement est détecté, ils envoient un signal au système pour activer d'autres actions.

3. **Capteurs de son** : Ils peuvent être utilisés pour capter des bruits suspects, comme des cris ou des bruits soudains, ce qui peut indiquer un problème, comme une altercation entre étudiants.

Exemple dans une salle de classe :

- **Surveillance des zones sensibles** comme les **entrées** (porte) et **fenêtres**. Ces zones sont souvent plus vulnérables et peuvent être des points d'entrée non autorisés. Les caméras seront installées pour surveiller ces zones spécifiques.
- Les **capteurs de mouvement** peuvent être placés près de la porte et des fenêtres pour détecter toute intrusion ou tout mouvement anormal pendant les heures de classe.

**But** : Collecter toutes les données brutes qui seront utilisées pour l'analyse.

### 2. Transmettre (Transmission des données)

Une fois que les données sont capturées, elles doivent être envoyées au module de **traitement (IA)** pour analyse. La transmission doit être rapide et fiable pour garantir une réponse en temps réel, surtout lorsqu'il s'agit de situations d'urgence.

Protocole de transmission :

1. Les données capturées par les **caméras et capteurs** sont envoyées à un serveur ou à un module de traitement pour l'analyse.
2. **Transmission via réseau local (LAN)** : Dans une salle de classe ou un environnement contrôlé, les données peuvent être envoyées via un réseau interne (LAN) à un serveur local ou à un dispositif comme un **microcontrôleur Arduino** ou un **Raspberry Pi**.
3. **Transmission via Cloud** : Si un traitement plus complexe est nécessaire, comme l'analyse vidéo avec de l'intelligence artificielle, les données peuvent être envoyées à des serveurs dans le **cloud** via Internet.

Exemple de transmission :

1. Une fois qu'un capteur de mouvement ou une caméra détecte un mouvement ou une activité suspecte dans une zone sensible (comme près de la fenêtre), la **vidéo ou les données du capteur** sont envoyées au serveur pour un traitement ultérieur.
2. Les **caméras IP** peuvent être configurées pour transmettre les données vidéo directement au système de gestion centralisé, qui est capable d'analyser les informations en temps réel.

### 3. Analyser (Traitement et Analyse des données)

L'étape finale consiste à **analyser** les données capturées et transmises, généralement à l'aide d'un **système d'intelligence artificielle (IA)** ou de **logiciels de traitement d'images** pour identifier des événements ou comportements inhabituels.

1. Types d'analyse effectuées :
2. **Analyse vidéo** : Les vidéos sont traitées par des **algorithmes de vision par ordinateur**. Cela peut inclure :
  - **Détection de mouvement** : Si le mouvement détecté dans une zone sensible (entrée ou fenêtre) dépasse un certain seuil (par exemple, si une personne entre dans la pièce en dehors des heures de cours), une alerte est envoyée.
  - **Reconnaissance faciale** : L'IA peut être utilisée pour identifier des personnes dans la salle, vérifier si des personnes non autorisées entrent ou détecter des comportements suspects.
  - **Comportements anormaux** : Si l'IA détecte un comportement inhabituel, comme une personne se déplaçant de manière erratique ou une altercation, elle peut envoyer une alerte.
3. **Analyse sonore** : Si des sons suspects sont captés, l'IA peut utiliser des **algorithmes de traitement du signal** pour distinguer des sons normaux de bruits anormaux, comme des cris, des bruits de verre brisé, ou des sons d'agression.

#### Exemple d'analyse dans une salle de classe :

1. Les **caméras de surveillance** surveillent constamment les **zones sensibles** comme l'entrée et les fenêtres.
  - Si un **mouvement suspect** est détecté dans l'une de ces zones (par exemple, une fenêtre qui est ouverte à une heure inhabituelle), l'IA déclenche une alerte et envoie une notification à l'administrateur ou à l'enseignant pour qu'il prenne des mesures immédiates.
  - Si des **bruits suspects** (cris ou objets brisés) sont détectés à proximité de ces zones, l'IA peut également signaler un incident à l'équipe de sécurité ou à l'enseignant.

## Conclusion

Ce schéma en trois étapes (Capturer → Transmettre → Analyser) montre comment un système de surveillance moderne peut fonctionner pour surveiller une salle de classe, notamment en surveillant les zones sensibles telles que les entrées et fenêtres. Chaque étape joue un rôle essentiel dans la détection rapide d'événements suspects et dans la fourniture d'une réponse proactive, renforçant ainsi la sécurité et la surveillance dans l'environnement éducatif.

### Traitement des données capturées

La phase d'analyse dans un système de surveillance domotique intelligent (comme celui que vous implémentez dans une salle de classe) joue un rôle fondamental pour transformer les données brutes collectées par les capteurs et caméras en informations exploitables. Cette analyse est effectuée principalement par des **algorithmes d'intelligence artificielle (IA)**, qui permettent de traiter des volumes massifs de données (images et vidéos) en temps réel, et d'identifier des comportements ou événements anormaux.

#### 1. Objectifs du Traitement des Données Capturées

Le principal objectif du traitement des données dans un système de surveillance intelligent est de **détecter des événements anormaux**, comme une intrusion, un comportement suspect, ou toute activité inhabituelle. L'analyse permet de filtrer les **fausses alertes** et de n'envoyer que des notifications pertinentes pour que l'utilisateur (dans ce cas, un enseignant ou un administrateur) puisse réagir rapidement.

#### 2. Types d'Algorithmes d'IA Utilisés pour l'Analyse

##### a) Algorithmes de Vision par Ordinateur (Computer Vision)

Les **algorithmes de vision par ordinateur** sont utilisés pour analyser les images et les vidéos en extrayant des informations visuelles utiles. Voici quelques techniques clés utilisées dans le contexte de la surveillance par caméra :

###### 1. Détection de mouvement :

- Les caméras capturent en continu des flux vidéo, et les **algorithmes de détection de mouvement** comparent les images successives pour détecter des différences. Si une différence significative est détectée, cela peut être interprété comme un mouvement. Par exemple, un individu qui entre dans la salle de classe ou une fenêtre qui est ouverte peut être signalé comme un mouvement suspect.

###### 2. Segmentation d'image :

- La **segmentation d'image** permet de diviser une image en différentes régions afin d'identifier les objets ou les zones spécifiques. Par exemple, l'algorithme peut distinguer entre le **fond de la pièce** et les **objets ou personnes** qui y sont présents. Cela permet de suivre des objets ou de repérer des personnes en mouvement dans la salle de classe.

###### 3. Suivi d'objets (Tracking) :

- Une fois qu'un objet ou une personne a été détecté, les **algorithmes de suivi d'objets** sont utilisés pour suivre leur position dans la vidéo au fil du temps. Par exemple, si une personne se déplace dans la salle, l'algorithme suivra sa trajectoire pour s'assurer que son mouvement est surveillé jusqu'à ce qu'il quitte le champ de vision.

###### 4. Reconnaissance faciale :

- **L'algorithme de reconnaissance faciale** est l'un des algorithmes les plus courants pour identifier les personnes dans un environnement surveillé. Ce système compare les traits du visage capturés par la caméra avec une **base de données de visages enregistrés** pour identifier des individus

autorisés ou non autorisés. Par exemple, dans une salle de classe, l'IA pourrait identifier un intrus en comparant le visage de la personne avec une liste de visages d'élèves ou de professeurs enregistrés dans le système.

### b) Analyse des Sons (Traitement Audio)

En plus de la vision par ordinateur, certains systèmes utilisent également des **algorithmes de traitement audio** pour analyser les sons capturés par des microphones ou des capteurs de son. Voici quelques applications possibles dans le cadre d'une salle de classe :

#### 1. Détection de bruits anormaux :

- Des **algorithmes d'analyse audio** peuvent être utilisés pour identifier des bruits inhabituels, comme des **cris**, des **objets tombés**, ou des **bruits de lutte**. Ces bruits peuvent indiquer une situation d'urgence, comme une altercation entre étudiants. Dès que ce type de son est détecté, une alerte peut être envoyée à l'enseignant ou à l'administrateur.

#### 2. Reconnaissance de mots-clés :

- Les systèmes peuvent être programmés pour reconnaître des **mots-clés spécifiques** ou des phrases particulières (par exemple, des **mots de panique** ou des **alertes de sécurité**) dans l'environnement. Si des termes sensibles sont détectés dans l'audio, cela pourrait déclencher une notification automatique pour signaler un danger imminent.

### c) Apprentissage Supervisé et Non Supervisé

Les algorithmes d'IA pour le traitement des données vidéo et audio sont souvent basés sur des méthodes **d'apprentissage supervisé** et **non supervisé**, selon la complexité et les besoins du système.

#### 1. Apprentissage supervisé :

- Dans l'apprentissage supervisé, un **modèle d'IA** est **formé** sur un **ensemble de données étiquetées**. Par exemple, des vidéos contenant des événements de **comportements suspects** peuvent être étiquetées comme tels, et l'IA apprendra à reconnaître ces types d'événements dans de nouvelles vidéos. Ce modèle peut ensuite être utilisé pour classer de nouvelles vidéos ou images en **comportement normal** ou **comportement suspect**.

#### 2. Apprentissage non supervisé :

- Dans l'apprentissage non supervisé, l'algorithme analyse des **données non étiquetées** pour identifier des **patterns ou des anomalies** dans les comportements. Par exemple, il pourrait détecter un changement soudain dans les habitudes des mouvements dans la salle (par exemple, des étudiants qui bougent de manière inhabituelle ou en groupe) sans avoir besoin de données étiquetées. Cela permet de détecter des anomalies dans des comportements non définis au préalable.

## 3. Processus d'Analyse des Données

### a) Prétraitement des Données

Avant que l'IA ne commence son analyse, les données capturées peuvent être **prétraitées** pour les rendre plus faciles à analyser :

1. **Amélioration de l'image** (par exemple, **réduction du bruit** dans les images ou vidéos capturées).
2. **Conversion** des vidéos en images fixes pour l'analyse, ou réduction de la résolution pour réduire les coûts de traitement tout en conservant les informations nécessaires.

## b) Identification et Classification

L'IA passe ensuite par plusieurs étapes pour **identifier et classifier** les objets ou personnes dans les images :

1. **Détection des objets** : L'algorithme identifie différents objets ou individus dans l'image ou la vidéo (par exemple, des étudiants, des chaises, des bureaux, etc.).
2. **Classification** : Une fois les objets identifiés, le système peut classer ces objets en différentes catégories, comme des **personnes autorisées**, des **objets suspects**, ou des **zones interdites**.

## c) Détection des Anomalies

Après avoir identifié les objets, le système peut détecter des **anomalies**. Par exemple :

1. Si une **personne non autorisée** est détectée dans la salle, ou si une **fenêtre** est ouverte alors qu'elle ne devrait pas l'être.
2. Si des **comportements suspects** comme des déplacements rapides ou des groupements de personnes sont détectés, le système peut en tirer la conclusion qu'une situation anormale est en train de se produire.

## 4. Réactions et Alertes

Une fois l'analyse effectuée et une anomalie détectée, le système peut **réagir** de manière automatique :

1. **Envoi d'alertes** : Une fois qu'un événement anormal est détecté, le système envoie immédiatement une alerte sous forme de **notification push, SMS, ou email** à un administrateur, un enseignant, ou à toute autre personne responsable.
2. **Prise de décision automatique** : Par exemple, si une intrusion est détectée pendant que la classe est vide, le système pourrait activer un **alarme sonore, signaler la situation au personnel de sécurité** et verrouiller les portes automatiquement.

## Conclusion

Le **traitement des données capturées** dans un système de surveillance domotique intelligent repose sur des **algorithmes d'IA** puissants qui permettent de **détecter des événements suspects** dans une salle de classe ou un environnement similaire. Ces algorithmes, tels que la **vision par ordinateur, l'apprentissage supervisé et non supervisé**, et l'**analyse audio**, permettent de filtrer les alertes pertinentes et d'assurer une surveillance proactive. Grâce à cette analyse en temps réel, la sécurité et la gestion des incidents sont considérablement améliorées.

## VI.1.2 Techniques de reconnaissance faciale et de détection d'objets (intrus, objets abandonnés)

Les **techniques de reconnaissance faciale** et de **détection d'objets** sont des domaines clés dans l'intelligence artificielle et la surveillance. Ces technologies jouent un rôle crucial dans les systèmes de sécurité modernes, notamment pour identifier des individus non autorisés (intrus) ou détecter des objets abandonnés qui pourraient représenter des menaces. Voici une explication détaillée des deux techniques :

### 1. Reconnaissance Faciale

La **reconnaissance faciale** est un procédé qui permet de **reconnaître un individu** à partir de son visage. Elle repose sur des algorithmes d'**intelligence artificielle** qui comparent les traits du visage capturés avec des bases de données pour identifier une personne.

## a) Fonctionnement de la Reconnaissance Faciale

Le processus de reconnaissance faciale peut être divisé en plusieurs étapes principales :

### i) Détection du Visage

Avant d'identifier un visage, il faut d'abord le **localiser** dans l'image ou la vidéo. Cette détection repose sur des **algorithmes de détection de visage**, qui permettent de repérer un visage humain dans une scène.

1. **Algorithmes utilisés** : L'un des algorithmes les plus populaires pour la détection des visages est le **Haar Cascade Classifier**, développé par OpenCV, qui analyse des caractéristiques simples de l'image (comme les contours ou les formes géométriques) pour détecter un visage. Un autre algorithme courant est le **HOG (Histogram of Oriented Gradients)**, qui examine les orientations des gradients d'intensité pour détecter des objets comme les visages.
2. **Réseaux neuronaux convolutifs (CNN)** : Les réseaux de neurones convolutifs sont de plus en plus utilisés pour la détection des visages, car ils permettent d'apprendre des caractéristiques complexes dans les images, rendant la détection plus précise.

### ii) Extraction des Caractéristiques Faciales

Une fois le visage détecté, le système extrait les **caractéristiques faciales** qui sont uniques à chaque individu. Ces caractéristiques peuvent inclure :

1. **La forme du visage** (ovale, carré, rond).
2. **Les traits du visage** : La distance entre les yeux, la forme du nez, la courbure des lèvres, etc.
3. **Les points caractéristiques** (landmarks) : Des points précis sur le visage, comme les coins des yeux, la ligne de la mâchoire, le sommet du nez, etc.

Les caractéristiques extraites du visage permettent de créer une **empreinte faciale** (aussi appelée **vecteur facial**), qui représente l'identité d'une personne sous forme numérique.

### iii) Comparaison avec la Base de Données

Une fois que l'empreinte faciale est extraite, elle est comparée à une base de données d'**empreintes faciales** existantes pour identifier si cette personne est déjà enregistrée. Les techniques de **matching** (correspondance) les plus courantes sont :

1. **Distance Euclidienne** : Mesure la similarité entre les empreintes faciales en comparant la distance entre leurs vecteurs dans l'espace des caractéristiques.
2. **Métrique de Similarité Cosinus** : Compare l'angle entre les vecteurs faciaux dans un espace multidimensionnel, ce qui donne une idée de la similarité entre les deux visages.

### iv) Identification et Vérification

- **Identification** : Chercher l'identification parmi une base de données d'utilisateurs. Le système essaie d'identifier qui est la personne dans la vidéo ou la photo.
- **Vérification** : Confirmer si une personne donnée (par exemple, un élève ou un intrus potentiel) correspond à un identifiant spécifique dans la base de données. Dans ce cas, le système recherche un seul visage et le compare à un seul enregistrement.

## b) Applications de la Reconnaissance Faciale

1. **Sécurité** : Identifier les intrus dans un bâtiment ou une salle de classe en temps réel. Si une personne non autorisée entre dans un espace protégé, l'algorithme de reconnaissance faciale peut émettre une alerte.
2. **Contrôle d'accès** : Remplacer ou compléter des systèmes de badges d'identification dans des environnements sécurisés, en permettant un accès automatique basé sur la reconnaissance du visage.
3. **Surveillance automatisée** : Analyser les flux vidéo en temps réel pour repérer des individus spécifiques parmi une foule ou un groupe d'étudiants.

### 2. Détection d'Objets : Intrus et Objets Abandonnés

La **détection d'objets** dans les systèmes de surveillance permet de repérer des objets ou des personnes anormaux dans une zone surveillée. En particulier, la **détection d'intrus** et la **détection d'objets abandonnés** sont des applications courantes dans des environnements comme des écoles, des bâtiments publics, ou des espaces commerciaux.

#### a) Détection d'Intrus

La **détection d'intrus** fait partie des systèmes de sécurité qui surveillent les zones pour repérer des individus qui n'ont pas l'autorisation d'être présents. Ces intrus peuvent être détectés grâce à des techniques d'**apprentissage automatique** et de **vision par ordinateur**.

##### i) Détection par Mouvement

Une méthode courante de détection d'intrus repose sur la **détection de mouvement**. Les caméras capturent des flux vidéo en continu, et les algorithmes analysent les images pour repérer des **mouvements suspects** dans des zones spécifiques (par exemple, les portes, fenêtres, ou passages secrets).

##### ii) Détection d'Intrus par Profilage Comportemental

Les algorithmes plus avancés peuvent analyser les **comportements** des individus dans une zone donnée. Par exemple, un intrus pourrait être repéré s'il adopte un comportement **inhabituel**, comme une **personne qui entre dans une zone non autorisée**, ou une **personne qui reste trop longtemps dans une zone particulière** sans raison apparente.

#### b) Détection d'Objets Abandonnés

La **détection d'objets abandonnés** permet de repérer des objets laissés sans surveillance dans un endroit, ce qui peut constituer un risque pour la sécurité (ex. : un sac abandonné, un objet suspect).

### i) Identification des Objets Abandonnés

Les systèmes de détection d'objets abandonnés utilisent des **réseaux neuronaux convolutionnels (CNN)**, capables de détecter et classifier des objets dans les images (par exemple, un sac, un colis, ou une valise). Lorsqu'un objet est détecté dans une zone où il ne devrait pas se trouver, le système signale une alerte.

### ii) Techniques de Suivi d'Objet

En plus de la détection initiale, les systèmes de surveillance peuvent utiliser des techniques de **suivi d'objets** pour suivre le déplacement d'un objet suspect, afin de déterminer s'il a été abandonné ou s'il est simplement déplacé par une personne autorisée.

### iii) Détection de Comportements Anormaux

L'algorithme peut également détecter des **comportements suspects** associés aux objets, comme une personne qui dépose un objet et s'éloigne rapidement sans surveillance.

## c) Applications de la Détection d'Intrus et d'Objets Abandonnés

1. **Surveillance de la sécurité** : Repérer des intrus qui pénètrent dans une zone protégée, ou des objets laissés sans surveillance dans des lieux publics.
2. **Gestion des risques** : Identifier rapidement des objets potentiellement dangereux (comme des sacs abandonnés dans un aéroport ou une école).
3. **Réponse automatisée** : En cas de détection d'un intrus ou d'un objet abandonné, des actions automatiques peuvent être entreprises (par exemple, verrouillage des portes, envoi d'une alerte à la sécurité).

## Conclusion

Les **techniques de reconnaissance faciale** et de **détection d'objets** sont des outils puissants pour renforcer la sécurité dans les environnements surveillés. La reconnaissance faciale permet d'identifier des individus spécifiques, tandis que la détection d'objets permet de repérer des anomalies dans l'environnement, comme des intrus ou des objets abandonnés. Ces technologies, combinées à l'intelligence artificielle, offrent une surveillance proactive et réactive, capable de répondre en temps réel aux situations de sécurité.

## VI.1.3 Classification des événements : activité normale vs activité suspecte

La **classification des événements** est un processus clé dans les systèmes de surveillance et de sécurité automatisés. Elle permet de distinguer les activités **normales** des **activités suspectes** en fonction de comportements ou d'événements capturés par des caméras ou des capteurs. L'objectif principal de cette classification est de réduire le nombre de fausses alertes et de focaliser l'attention du système de sécurité sur des événements réellement importants.

Cette classification repose sur des techniques avancées d'**intelligence artificielle (IA)** et de **machine learning**, qui analysent les données collectées par les caméras et capteurs pour évaluer si un événement observé est normal ou s'il présente des signes de danger potentiel.

### 1. Activité Normale vs Activité Suspecte

#### a) Activité Normale

Une **activité normale** fait référence à des comportements ou des événements qui se produisent régulièrement et qui ne posent pas de risque immédiat pour la sécurité. Ces événements sont souvent associés à des **comportements prévisibles** ou attendus dans un environnement donné. Par exemple, dans une salle de classe ou un bâtiment, une activité normale peut inclure :

1. Des personnes se déplaçant de manière ordonnée (entrer, sortir ou se déplacer d'un endroit à un autre).
2. Des étudiants ou des employés en train de travailler dans leurs bureaux ou de se réunir dans une salle de réunion.
3. Des mouvements réguliers, comme des employés arrivant et sortant du bâtiment pendant les heures de travail.
4. La présence d'objets dans des zones spécifiques (par exemple, des sacs laissés dans des coins de la salle mais qui ne semblent pas avoir été abandonnés de manière suspecte).

Critères de détection des activités normales :

Les systèmes utilisent des modèles basés sur des **comportements répétitifs** pour identifier ce qui est considéré comme normal. Cela peut inclure :

1. **Le nombre et la position des personnes** présentes dans une zone.
2. **Les horaires habituels** d'activité dans un espace (par exemple, les heures de travail ou de classe).
3. **Les trajectoires de mouvement** régulières des personnes.

## b) Activité Suspecte

Une **activité suspecte** fait référence à des comportements ou événements qui diffèrent de l'activité normale et qui peuvent indiquer un risque pour la sécurité, une intrusion ou une menace potentielle. Ces comportements peuvent inclure :

1. **Intrusion dans des zones interdites** : Par exemple, une personne qui pénètre dans une zone réservée au personnel ou dans une zone où elle n'est pas censée être.
2. **Comportement étrange ou inattendu** : Une personne qui se déplace de manière erratique, qui semble se cacher, ou qui reste immobile dans une zone pendant un temps inhabituellement long.
3. **Objets abandonnés** : Des sacs, valises ou autres objets laissés sans surveillance, surtout dans des endroits à forte sécurité comme des écoles, des aéroports ou des bâtiments gouvernementaux.
4. **Comportements violents ou agressifs** : Par exemple, une bagarre ou un individu qui agit de manière menaçante.
5. **Mouvements anormaux** : Un individu qui prend un chemin inhabituel dans une zone surveillée, ou qui entre et sort rapidement d'une zone sans raison apparente.
6. **Tentatives de sabotage ou de destruction** : Par exemple, des gestes qui semblent indiquer qu'une personne tente de détruire des équipements de sécurité ou de causer des dommages à la propriété.

Critères de détection des activités suspectes :

Pour identifier une activité suspecte, les systèmes de surveillance exploitent plusieurs indicateurs d'anomalie, souvent à travers l'IA, qui sont comparés à un modèle **comportemental normal** :

1. **Vitesse de déplacement anormale** : Si une personne se déplace trop rapidement ou trop lentement, cela peut indiquer un comportement suspect.
2. **Mouvements erratiques** : Une personne qui se déplace de manière irrégulière ou qui semble suivre un itinéraire non linéaire, peut être un signe de comportement étrange.

3. **Zones interdites** : Une personne pénétrant dans une zone où elle n'est pas censée être déclenche une alerte.
4. **Temps passé dans une zone** : Un individu qui reste trop longtemps dans une zone particulière sans raison apparente peut être suspect.
5. **Vitesse d'apparition de nouveaux objets** : Un objet soudainement placé dans une zone sensible sans raison apparente (sac abandonné, colis oublié) peut signaler un danger.

## 2. Méthodes de Classification : Comment Identifier l'Activité Normale vs Suspecte

La **classification des événements** entre normal et suspect se fait principalement à l'aide de deux approches principales :

### a) Apprentissage Supervisé

L'apprentissage supervisé repose sur l'utilisation de jeux de données étiquetés, dans lesquels les événements sont déjà classés comme « normal » ou « suspect ». Les modèles d'IA sont formés pour **apprendre** à partir de ces données et à **généraliser** ce qu'est une activité normale et suspecte. Ce processus se déroule en trois étapes :

1. **Collecte des données** : Les données sont collectées via des capteurs et caméras (par exemple, les images ou vidéos des événements).
2. **Annotation des événements** : Les événements sont classifiés par des experts ou des annotateurs pour indiquer s'ils sont considérés comme normaux ou suspects.
3. **Entraînement du modèle** : Un algorithme (comme un **réseau neuronal convolutionnel (CNN)** pour l'analyse des images ou des **forêts aléatoires**) est utilisé pour entraîner le système à identifier les motifs d'activités normales et suspectes.

Exemples d'algorithmes d'apprentissage supervisé :

1. **Réseaux neuronaux profonds** : Ces réseaux sont utilisés pour la détection d'objets ou de visages, ou pour l'analyse des comportements visuels dans les vidéos.
2. **Forêts aléatoires** : Un modèle populaire dans les systèmes de classification, où plusieurs arbres de décision sont construits pour prédire si un événement est normal ou suspect.
3. **Support Vector Machines (SVM)** : Un autre modèle d'apprentissage supervisé qui peut être utilisé pour classer des événements en fonction des caractéristiques extraites des données.

### b) Apprentissage Non Supervisé

L'apprentissage non supervisé ne nécessite pas de données étiquetées, ce qui est utile lorsque les données ne sont pas préalablement classées ou lorsque les comportements nouveaux sont détectés. L'objectif est d'identifier des **anomalies** ou des **déviations** par rapport aux habitudes normales sans intervention humaine.

Les algorithmes non supervisés recherchent des motifs dans les données et détectent les événements qui diffèrent de la norme. Cela permet au système d'identifier des **comportements inconnus**.

Exemples d'algorithmes d'apprentissage non supervisé :

1. **Clustering** (par exemple, **k-means**) : Cette méthode permet de grouper des événements similaires et d'identifier ceux qui s'éloignent des groupes.
2. **Méthodes de détection des anomalies** : Des techniques comme **Isolation Forests** ou **One-Class SVM** permettent de détecter les événements qui s'écartent significativement des comportements observés.

### 3. Amélioration Continue de la Classification

La **classification des événements** évolue continuellement grâce à l'utilisation de **données en temps réel** et de systèmes de feedback. Lorsqu'un événement suspect est détecté, une alerte peut être envoyée aux opérateurs humains pour évaluation. Ces opérateurs peuvent ensuite confirmer ou infirmer l'activité suspecte, ce qui permet de **réajuster** le modèle d'IA pour mieux détecter des événements futurs.

En outre, les systèmes modernes intègrent des mécanismes de **réapprentissage automatique**. Cela permet au système d'apprendre de ses erreurs et d'améliorer sa capacité à détecter de nouvelles anomalies au fur et à mesure qu'elles se produisent dans l'environnement surveillé.

#### Conclusion

La **classification des événements** entre activité normale et suspecte est essentielle pour automatiser la surveillance et éviter les fausses alertes. Grâce à des techniques avancées d'intelligence artificielle, notamment l'apprentissage supervisé et non supervisé, les systèmes modernes de sécurité peuvent détecter en temps réel les comportements et événements anormaux qui nécessitent une intervention. Cette approche permet une **réactivité accrue** tout en réduisant la charge de travail des opérateurs de sécurité, en se concentrant sur les incidents réels plutôt que sur des événements banals.

VI.2 Gestion et filtrage des anomalies :

VI.2.1 Analyse des mouvements - Vitesse, direction, taille

Dans le cadre des systèmes de surveillance et de sécurité automatisés, la **gestion et le filtrage des anomalies** jouent un rôle crucial dans la détection des comportements suspects ou des incidents potentiellement dangereux. L'analyse des **mouvements** est l'une des composantes fondamentales de cette gestion, permettant de comprendre et de différencier les comportements normaux des comportements anormaux en fonction de critères spécifiques comme la **vitesse**, la **direction** et la **taille** des objets ou des personnes en mouvement.

L'objectif principal de cette analyse est de détecter toute **anomalie** ou activité suspecte qui pourrait indiquer un risque, une intrusion ou un comportement inhabituel, tout en minimisant les fausses alertes dues à des événements non menaçant.

##### a) Vitesse des mouvements

La **vitesse** d'un objet ou d'une personne en mouvement est un critère fondamental dans l'analyse des comportements. Une **vitesse de déplacement anormale** peut être le premier signe qu'une personne agit de manière suspecte. Par exemple, une personne qui se déplace trop rapidement dans un endroit où les gens marchent normalement à un rythme plus lent, ou quelqu'un qui court dans un bâtiment peut être un signe de danger potentiel.

Comment mesurer et analyser la vitesse ?

La vitesse est généralement calculée en mesurant la **distance parcourue** par un objet ou une personne pendant un certain intervalle de temps. Les capteurs ou caméras de surveillance capturent des images ou des vidéos à intervalles réguliers. En utilisant des algorithmes de traitement d'image et de suivi d'objets (comme la détection de mouvement ou l'algorithme de suivi des objets en temps réel - **object tracking**), le système peut calculer la distance parcourue entre chaque image et en déduire la vitesse.

Exemples d'analyses de vitesse :

1. **Comportement suspect** : Une personne qui marche très rapidement vers une zone interdite.
2. **Intrusion** : Un individu courant dans une zone où la course est inhabituelle (par exemple, dans un musée, un hôpital ou une école).
3. **Anomalies dans une foule** : Un individu qui court ou se précipite dans une foule peut indiquer une fuite ou un incident (comme une bagarre).

Critères de vitesse suspects :

1. **Vitesse trop élevée** : Si la vitesse d'une personne dépasse un seuil défini (en fonction du contexte du lieu), elle peut être signalée comme anormale.
2. **Vitesse trop faible** : Un mouvement excessivement lent, ou un individu qui se déplace de manière molle et délibérée dans une zone où la vitesse normale est plus rapide, peut également être un signe de comportement suspect.

### b) Direction des mouvements

La **direction** du mouvement est un autre critère essentiel pour identifier les anomalies. Dans un environnement surveillé, certaines directions sont considérées comme normales, tandis que d'autres peuvent indiquer des comportements indésirables. Par exemple, un individu se dirigeant vers une porte d'accès interdite ou vers un lieu clos sans raison apparente peut déclencher une alerte.

#### Comment analyser la direction ?

Les systèmes de surveillance modernes utilisent des techniques de **suivi d'objet** pour suivre le déplacement d'une personne ou d'un objet sur un certain trajet. À chaque frame ou image capturée, l'algorithme calcule les **coordonnées de départ et d'arrivée**, ce qui permet de déterminer si la personne suit un trajet habituel ou s'écarte d'un parcours standard.

Exemples d'analyse de la direction :

1. **Comportement suspect** : Une personne qui se déplace vers une **zone interdite** (par exemple, une porte verrouillée ou une zone de stockage).
2. **Comportement anormal** : Un individu qui fait des allers-retours incessants dans un espace spécifique sans raison apparente (ce qui pourrait être un signe de furtivité ou de préparation à un vol).
3. **Comportement de fuite** : Un individu qui change rapidement de direction pour échapper à une situation (comme une personne qui court vers une sortie lors d'un incident de sécurité).

Critères directionnels suspects :

1. **Changement brusque de direction** : Si une personne change soudainement de direction de manière inexplicable, cela peut indiquer qu'elle cherche à éviter une zone de surveillance.
2. **Déviation d'un itinéraire normal** : Si un individu dévie de son chemin prévu sans raison apparente (par exemple, en entrant dans une zone interdite), l'alerte peut être déclenchée.

### c) Taille des objets ou des personnes en mouvement

La **taille** des objets ou des personnes en mouvement permet de déterminer s'il y a des anomalies en fonction de l'**échelle** des objets observés. Par exemple, une personne plus grande ou plus petite que la normale dans un

environnement pourrait être détectée comme un événement suspect. De même, la taille peut être utilisée pour détecter des objets qui ne devraient pas être présents dans une zone surveillée (comme un bagage abandonné dans un aéroport).

Comment analyser la taille ?

Les systèmes de surveillance basés sur l'IA ou le **deep learning** peuvent estimer la taille d'un objet ou d'une personne en fonction de leur **position dans l'image** et de la **distorsion de perspective**. La taille relative d'un objet dans une image est souvent calculée en fonction de la distance entre la caméra et l'objet, en utilisant des algorithmes de **stéréovision** ou de **calcul de profondeur**.

Exemples d'analyse de la taille :

1. **Comportement suspect** : Un objet de grande taille laissé dans une zone où il ne devrait pas se trouver (par exemple, un sac volumineux oublié près d'une porte de sécurité dans un aéroport).
2. **Intrusion suspecte** : Un objet suspect de petite taille qui se déplace de manière inhabituelle dans une zone sensible.
3. **Comportement incongru** : Une personne de taille anormalement petite ou grande dans un endroit où l'on attend des individus de taille standard (par exemple, une personne très petite qui tente d'accéder à une zone réservée).

Critères de taille suspects :

1. **Objet trop grand pour l'environnement** : Si un objet ou une personne semble occuper plus d'espace que la moyenne dans une zone donnée, cela peut déclencher une alerte.
2. **Objet suspect dans une zone non surveillée** : La taille peut aussi être utilisée pour différencier un petit objet posé de manière inhabituelle (comme un paquet) d'une personne qui se cache dans un espace clos.

### 3. Filtrage des anomalies : Gestion des fausses alertes

L'un des défis majeurs dans la gestion des anomalies est de **réduire les fausses alertes**, qui sont souvent générées par des comportements non menaçant, mais interprétés à tort comme suspects. Le filtrage des anomalies repose sur l'analyse des **mouvements**, en utilisant les critères de vitesse, direction et taille pour éliminer les événements qui ne sont pas significatifs.

Techniques de filtrage :

1. **Combinaison de critères** : En croisant la vitesse, la direction et la taille, le système peut déterminer si un événement est réellement suspect ou s'il s'agit d'une simple activité normale. Par exemple, une personne marchant lentement vers une porte de sortie pendant les heures normales d'activité peut être ignorée, tandis qu'une personne courant dans la même direction peut être signalée comme suspecte.
2. **Évaluation contextuelle** : Le système analyse les mouvements dans le contexte de l'environnement surveillé. Par exemple, dans un entrepôt, les mouvements des employés sont considérés comme normaux, mais un mouvement soudain dans une zone de stockage peut être détecté comme suspect.
3. **Apprentissage automatique** : Les modèles d'IA peuvent être entraînés pour différencier les anomalies des comportements ordinaires en apprenant à partir de données historiques et en ajustant les seuils de détection pour chaque type de mouvement.

## Conclusion

L'analyse des mouvements, en se basant sur des critères tels que la **vitesse**, la **direction** et la **taille**, est essentielle pour la détection d'anomalies dans les systèmes de surveillance modernes. Cette analyse permet de différencier les comportements normaux des comportements potentiellement suspects, ce qui améliore l'efficacité du système en réduisant les fausses alertes et en augmentant la réactivité face aux menaces réelles. L'intégration de ces critères dans un système intelligent de surveillance permet une gestion efficace des risques tout en assurant une surveillance continue et proactive des espaces sensibles.

### VI.2.2 Détection des comportements inhabituels : Regroupement de personnes

La détection des **comportements inhabituels** dans les systèmes de surveillance intelligents est une composante clé pour identifier des situations de risque, d'anomalie ou d'urgence. Un des comportements les plus pertinents à surveiller est le **regroupement de personnes**, qui peut être le signe d'un événement anormal, comme une altercation, une agitation ou un rassemblement non autorisé. Ce type de détection s'appuie sur l'analyse de données visuelles ou sensorielles pour identifier les comportements qui diffèrent de l'activité normale dans un environnement donné.

Pourquoi surveiller le regroupement de personnes ?

Le **regroupement de personnes** peut être un indicateur important pour plusieurs raisons :

1. **Incidents sociaux ou violents** : Un regroupement soudain de personnes peut signifier une bagarre, un attroupement ou même une manifestation, ce qui peut devenir une situation de sécurité préoccupante.
2. **Rassemblement non autorisé** : Dans des lieux tels que des établissements scolaires, des bureaux ou des événements publics, un regroupement de personnes non autorisé peut être une violation des règles de sécurité ou des lois.
3. **Conditions dangereuses** : Lors d'une situation d'urgence, comme une alerte incendie, un regroupement de personnes près des sorties ou des zones sensibles peut être potentiellement dangereux si les gens bloquent les passages ou agissent de manière chaotique.

### Détection et analyse des regroupements de personnes

La détection d'un **regroupement de personnes** repose sur l'utilisation de technologies avancées telles que la vision par ordinateur, l'intelligence artificielle (IA) et l'analyse de flux de données. Plusieurs techniques sont utilisées pour détecter les comportements inhabituels dans les foules.

#### 1. Vision par ordinateur et suivi d'objets

L'analyse du **comportement des foules** commence souvent par la capture des données visuelles à l'aide de **caméras de surveillance**. Les algorithmes de **vision par ordinateur** permettent d'extraire et de suivre les **mouvements** des individus dans une zone surveillée, en analysant les flux de personnes, leur densité et leur comportement.

Comment ça fonctionne ?

1. **Détection des personnes** : Tout d'abord, un algorithme d'**identification des objets** (comme le modèle YOLO, Faster R-CNN ou SSD) détecte les personnes dans les images ou les vidéos en capturant les contours et les formes caractéristiques des individus.

- Suivi d'objets** : Ensuite, les personnes détectées sont suivies à travers les différentes frames vidéo à l'aide de techniques de **suivi d'objets** (par exemple, **Kalman Filter** ou **DeepSORT**), ce qui permet de connaître leur trajectoire, leur vitesse et leur position relative.
- Identification du regroupement** : En analysant la densité de personnes dans une zone donnée, il est possible de déterminer si plusieurs individus sont proches les uns des autres, ce qui peut indiquer un regroupement ou une formation de groupe. Lorsque le nombre d'individus dans un espace donné dépasse un certain seuil, une alerte peut être déclenchée.

### **Exemple de cas de regroupement détecté :**

- Une **foule dense** qui se forme soudainement dans un espace restreint, comme près d'une porte de sortie, peut être identifiée grâce à l'analyse des positions des individus et de la proximité entre eux.

#### 2. Détection de la densité de la foule

La **densité** d'une foule est une mesure importante qui permet de détecter si un groupe de personnes devient trop compact, ce qui peut indiquer un **comportement anormal**. Un regroupement de personnes dans une zone où la densité est généralement faible peut être considéré comme suspect.

Comment ça fonctionne ?

- Calcul de la densité** : En mesurant la distance entre chaque individu et en calculant le nombre d'individus par unité d'espace (par exemple, par pixel ou par mètre carré), il est possible de déterminer si la zone est densément peuplée.
- Seuils de densité** : Si la densité dépasse un certain seuil préétabli, indiquant qu'un nombre inhabituellement élevé de personnes se regroupe dans une zone spécifique, le système déclenche une alerte. Ces seuils peuvent être dynamiques et ajustés en fonction de l'environnement ou de l'heure de la journée (par exemple, une densité plus élevée peut être normale dans un auditorium pendant une conférence, mais pas dans un couloir vide).

### **Exemple de cas de détection de densité anormale :**

- Un **regroupement soudain** de personnes dans un espace étroit, comme une salle d'attente ou un couloir, où la densité dépasse les normes de sécurité ou le comportement habituel des personnes dans cet espace.

#### 3. Analyse des trajectoires de mouvement et des interactions

L'**analyse des trajectoires de mouvement** des personnes peut également fournir des indices sur un regroupement de personnes. Par exemple, si plusieurs individus convergent vers un point donné (comme une porte ou une zone restreinte), cela peut être interprété comme un regroupement.

Comment ça fonctionne ?

- Suivi des trajectoires** : Les algorithmes de suivi d'objets identifient les trajectoires de plusieurs individus qui se déplacent vers un même endroit, ce qui peut indiquer qu'ils ont l'intention de se rassembler. Les trajectoires peuvent être analysées pour détecter des motifs inhabituels, comme un grand nombre de personnes qui convergent vers une zone spécifique.
- Interaction entre individus** : Les interactions entre les individus peuvent être détectées par l'analyse de leurs trajectoires. Par exemple, si plusieurs personnes se déplacent simultanément dans une direction, cela peut être interprété comme une forme de **rassemblement intentionnel**.

Exemple de regroupement via les trajectoires :

1. Plusieurs personnes qui se dirigent simultanément vers une même zone (comme une sortie d'urgence) peuvent être détectées comme un rassemblement non autorisé ou comme un comportement suspect (comme une tentative d'évasion pendant une situation de danger).

#### 4. Comportements de groupe et analyse contextuelle

Les comportements de groupe peuvent être analysés dans le contexte global de l'environnement. Par exemple, si un regroupement de personnes se forme dans un endroit où cela ne se produit généralement pas (comme une salle de classe vide, une zone de passage ou un hall d'entrée), cela peut déclencher une alerte.

Comment ça fonctionne ?

1. **Analyse contextuelle** : Le système analyse les **comportements typiques** dans un environnement particulier. Par exemple, dans un bâtiment de bureaux, les regroupements spontanés peuvent être plus fréquents lors des pauses, mais un rassemblement dans une zone normalement vide (comme un couloir ou une salle de stockage) peut être suspect.
2. **Utilisation des alertes** : En combinant les données de trajectoires, de densité et d'indices contextuels, le système peut déterminer si le regroupement est **naturel** (par exemple, pendant un événement social) ou **inhabituel** (par exemple, un rassemblement rapide de personnes dans un endroit fermé).

#### 5. Filtrage et gestion des fausses alertes

Il est essentiel d'éviter les fausses alertes, surtout lorsque des groupes de personnes peuvent se regrouper pour des raisons totalement innocentes, comme une pause-déjeuner ou un échange de mots dans un environnement social. Les systèmes utilisent des critères supplémentaires pour **filtrer les fausses alertes**, par exemple, en comparant les comportements suspects à un **modèle d'activité normale**.

### Conclusion

La détection du **regroupement de personnes** est un élément clé dans l'identification des comportements inhabituels ou suspects dans les systèmes de surveillance. Grâce à des technologies avancées comme la **vision par ordinateur**, l'**intelligence artificielle** et l'**analyse de densité de la foule**, il est possible de détecter rapidement les situations anormales et de déclencher des alertes en temps réel. L'**analyse des trajectoires de mouvement**, des interactions entre individus et des contextes spécifiques renforce la capacité à identifier des comportements problématiques tout en minimisant les fausses alertes, ce qui améliore la sécurité des environnements surveillés.

#### VI.2.3 Précision accrue grâce à l'apprentissage supervisé et aux modèles d'IA prédictifs

L'**apprentissage supervisé** et les **modèles d'intelligence artificielle (IA) prédictifs** sont des techniques avancées qui permettent d'améliorer considérablement la **précision** des systèmes de détection et de surveillance. Ces technologies sont au cœur des systèmes modernes de surveillance et de sécurité, car elles permettent d'**analyser des données complexes** (images, vidéos, sons, mouvements, etc.) et d'en extraire des informations significatives pour prédire et détecter des événements inhabituels avec un haut degré de fiabilité.

## 1. L'apprentissage supervisé

L'apprentissage supervisé est une technique d'IA où un modèle est entraîné sur un ensemble de données d'entrée étiquetées pour apprendre à effectuer des prédictions ou des classifications. Dans un contexte de surveillance, cela implique l'utilisation de **données étiquetées** qui contiennent à la fois des **exemples d'événements normaux** (comportement attendu) et **d'événements anormaux** (comportement suspect ou atypique), que le modèle apprend à distinguer.

Comment ça fonctionne dans la surveillance ?

1. **Ensemble de données étiquetées** : Pour entraîner un modèle supervisé, on utilise un **ensemble de données d'images ou de vidéos annotées**, où chaque image ou vidéo est étiquetée avec une catégorie (par exemple, « normal », « suspect », « intrusion », « regroupement de personnes », etc.).
2. **Entraînement du modèle** : Le modèle apprend à **reconnaître des caractéristiques spécifiques** dans les données, comme les mouvements inhabituels, les comportements suspects ou les regroupements de personnes. Il s'agit d'un processus d'entraînement où le modèle ajuste ses paramètres pour minimiser l'erreur de prédiction sur les exemples d'apprentissage.
3. **Test et évaluation** : Une fois le modèle entraîné, il est testé sur un ensemble de **données non étiquetées** (test set) pour vérifier sa capacité à prédire correctement les catégories des événements nouveaux. Cela permet d'évaluer sa **précision** (taux de bonnes prédictions) et d'améliorer le modèle si nécessaire.

Exemple d'application dans la détection d'intrusion :

Dans un système de surveillance de sécurité, un modèle supervisé peut être utilisé pour **classer des mouvements** détectés par des caméras comme « normal » ou « suspect ». Par exemple, si le modèle a été formé pour reconnaître des intrusions dans une zone protégée, il sera capable de signaler toute activité qui ne correspond pas aux comportements habituels observés dans l'environnement.

## 2. Modèles d'IA Prédicifs

Les **modèles prédictifs** en IA utilisent des **algorithmes d'apprentissage automatique** pour analyser des données historiques et en déduire des **comportements futurs** ou **des tendances potentielles**. Ces modèles sont capables d'anticiper des événements futurs en fonction de modèles de comportement observés dans le passé.

Comment ça fonctionne dans la surveillance ?

1. **Analyse des données historiques** : En surveillant les événements passés et les comportements des individus dans un environnement donné (par exemple, une salle de classe, un bureau, un entrepôt), les modèles d'IA prédictifs peuvent identifier des **patterns** récurrents et des tendances comportementales. Ces **données historiques** peuvent inclure des informations sur la position des individus, leur vitesse de déplacement, leur direction, etc.
2. **Prévision des comportements futurs** : Une fois qu'un modèle prédictif est entraîné avec ces données, il peut prédire des **événements futurs**. Par exemple, il pourrait prédire qu'un regroupement de personnes est susceptible de se produire dans une certaine zone, ou qu'un individu pourrait s'aventurer dans une zone interdite, basé sur les comportements précédents.
3. **Détection proactive des anomalies** : Ces modèles peuvent être utilisés pour anticiper des situations inhabituelles avant qu'elles ne se produisent, ce qui permet de **réagir proactivelement**. Par exemple, si le système détecte qu'une foule commence à se former dans une zone habituellement vide, il peut prédire que cela pourrait devenir un rassemblement dangereux et alerter la sécurité avant qu'un incident ne survienne.

Exemple d'application dans la prédiction de comportements suspects :

Un modèle prédictif dans un système de surveillance pourrait analyser les comportements des individus dans une zone donnée et prévoir des **changements de comportement**. Si plusieurs individus commencent à se regrouper à un certain endroit ou se déplacent dans une direction inhabituelle, le modèle pourrait prédire une **anomalie potentielle** (par exemple, un rassemblement non autorisé) et alerter les responsables de la sécurité avant que le regroupement ne devienne plus important.

### 3. Précision accrue grâce à la combinaison des deux approches

Lorsque l'apprentissage supervisé et les modèles prédictifs sont utilisés ensemble, ils offrent des avantages supplémentaires :

1. **Apprentissage supervisé** : Permet à l'IA de **classer précisément** les événements en temps réel en fonction des étiquettes d'entraînement et des caractéristiques des données.
2. **Modèles prédictifs** : Permettent à l'IA de **prévoir des événements futurs** et d'identifier des comportements suspects avant qu'ils ne se produisent, en fonction des tendances observées dans les données historiques.

Cette combinaison renforce la **précision du système** et lui permet de détecter des événements inhabituels, de prévenir des incidents avant qu'ils ne se produisent et d'adapter ses prédictions en fonction des nouveaux comportements observés. Cela permet d'augmenter l'efficacité des systèmes de surveillance intelligents et d'assurer une **sécurité proactive**.

Exemple d'utilisation conjointe :

Dans un environnement scolaire, un système de surveillance pourrait détecter un **regroupement de personnes** dans une zone spécifique (via l'apprentissage supervisé) et utiliser des modèles prédictifs pour **anticiper l'intention de ces personnes** (par exemple, se diriger vers une sortie, se rendre dans une zone interdite). En combinant ces deux approches, le système devient plus réactif et intelligent, en détectant les anomalies de manière plus précise et en prévoyant des comportements futurs potentiellement dangereux.

### 4. Amélioration continue avec des modèles adaptatifs

Les systèmes d'IA peuvent continuer à améliorer leur **précision** avec le temps grâce à des **algorithmes adaptatifs**. L'apprentissage supervisé et les modèles prédictifs peuvent être **réajustés** en fonction des nouvelles données et des événements réels détectés, ce qui permet au système de **s'adapter aux changements** dans l'environnement et les comportements observés.

## Conclusion

L'utilisation combinée de l'**apprentissage supervisé** et des **modèles d'IA prédictifs** permet d'augmenter considérablement la précision des systèmes de surveillance intelligents. Grâce à ces techniques, les systèmes peuvent détecter des événements anormaux en temps réel, prédire des comportements futurs et réagir de manière proactive avant que des situations de sécurité ne surviennent. L'intégration de ces modèles dans les dispositifs de surveillance est essentielle pour garantir une sécurité optimale dans divers environnements, en rendant la surveillance non seulement réactive mais aussi **proactive**.

## VI.3. Base de données MongoDB

### VI.3.1. Base de données MongoDB dans un système de surveillance intelligent

MongoDB est une base de données **NoSQL**, orientée document, qui est particulièrement adaptée aux applications nécessitant des performances élevées, une gestion flexible des données, et une évolutivité horizontale. Dans un système de surveillance intelligent, MongoDB joue un rôle clé dans le **stockage des données pertinentes** issues des caméras, capteurs et systèmes de traitement. Ces données peuvent inclure des vidéos, des journaux d'alertes, des présences quotidiennes, ainsi que les identités des individus détectés.

#### 1. Stockage des vidéos

Les vidéos capturées par les caméras de surveillance représentent une source de données volumineuse et continue. MongoDB peut être utilisé pour **stockage et gestion** de ces fichiers multimédia, notamment en utilisant son système de **grande capacité** pour stocker des documents binaires comme des vidéos.

Stockage des vidéos avec MongoDB :

1. **GridFS** : MongoDB dispose d'un système appelé **GridFS** qui permet de stocker de **grands fichiers binaires**, comme des vidéos ou des images, dans des segments de taille gérable. GridFS divise un fichier volumineux en **plusieurs fragments** et les stocke dans des collections séparées. Cela permet de gérer des fichiers dont la taille dépasse la capacité maximale d'un document MongoDB classique (16 Mo).
2. **Références aux vidéos** : Plutôt que de stocker directement les vidéos dans la base de données, MongoDB peut conserver des **références** vers les fichiers vidéo qui sont stockés dans un système de fichiers ou un service de cloud (par exemple, AWS S3). Cela peut être plus efficace, car la gestion des fichiers vidéo est alors optimisée pour les systèmes dédiés au stockage de fichiers volumineux, tout en maintenant des **métadonnées utiles** dans MongoDB (date de capture, identification de la caméra, durée de la vidéo, etc.).
3. **Exemple d'implémentation** : Lorsqu'une vidéo est capturée par une caméra, MongoDB peut stocker la vidéo elle-même via GridFS ou juste une référence au fichier vidéo stocké ailleurs. Les métadonnées associées à la vidéo (par exemple, l'heure, le lieu, le type d'événement enregistré) sont également stockées dans un **document MongoDB**, ce qui permet une recherche rapide et un accès facile aux informations.

#### 2. Journaux des alertes

Dans un système de surveillance intelligent, les **alertes** sont générées en fonction de l'analyse des données (mouvement détecté, intrusion, comportement suspect, etc.). Ces alertes doivent être enregistrées et stockées pour une gestion ultérieure, et peuvent être utilisées pour analyser la sécurité en temps réel ou pour des audits après un incident.

Stockage des journaux d'alertes :

1. **Collecte d'événements** : Chaque alerte générée par les caméras ou capteurs peut être enregistrée sous forme de document dans MongoDB. Chaque **document d'alerte** contiendra des informations sur l'événement (par exemple, type d'alerte, timestamp, localisation, etc.), le niveau de gravité de l'alerte, et éventuellement l'identification de l'utilisateur ou de l'administrateur qui a réagi à l'alerte.
2. **Structure d'un document d'alerte** :

```
{  
  "id_alerte": "12345",  
  "type": "intrusion",  
  "date": "2024-12-21T10:00:00Z",  
  "localisation": "entrée principale",  
}
```

```

    "niveau": "haut",
    "résolu": false,
    "actions": [
        { "action": "alerte sonore", "date": "2024-12-21T10:05:00Z" },
        { "action": "envoi d'email", "date": "2024-12-21T10:06:00Z" }
    ]
}

```

3. **Exemple d'utilisation :** Lorsqu'un mouvement suspect est détecté, une alerte peut être générée et stockée dans MongoDB avec toutes les informations pertinentes (type d'événement, emplacement, gravité, actions entreprises). Cela permet de suivre les incidents et de garantir que des mesures correctives sont prises.

### 3. Liste des présences journalières

Une fonctionnalité clé dans de nombreux systèmes de surveillance est le **suivi de la présence des individus** dans des zones spécifiques, comme une salle de classe, un bâtiment ou une zone de sécurité. En utilisant des caméras et des capteurs, le système peut **suivre la présence** des personnes et enregistrer les informations pertinentes dans une base de données.

Stockage des présences :

1. **Suivi des individus :** MongoDB peut être utilisé pour stocker les **entrées et sorties** des individus dans une zone surveillée. Lorsque quelqu'un entre ou sort d'une zone, le système peut détecter cette activité, enregistrer l'événement dans la base de données, et associer des **identifiants uniques** (par exemple, les identités des personnes détectées, les dates et heures d'entrée et de sortie).
2. **Structure d'un document de présence :**

```
{
    "id_individu": "98765",
    "nom": "John Doe",
    "date": "2024-12-21",
    "heure_arrivée": "08:00:00",
    "heure_sortie": "16:00:00",
    "zone": "salle 101",
    "identité_detectée": true
}
```

- Dans cet exemple, le système enregistre les informations sur l'identité de l'individu, ainsi que l'heure d'arrivée et de départ, permettant ainsi de **suivre les présences** dans un bâtiment ou une zone spécifique. Ces données peuvent être utilisées pour des rapports d'absence ou de présence, ou pour des fins de sécurité.

### 4. Identités détectées

Un autre aspect important du stockage des données dans MongoDB est la gestion des **identités détectées**, notamment dans les systèmes de reconnaissance faciale. Lorsqu'un individu est détecté par la caméra, son **identité** peut être capturée et stockée dans la base de données.

Stockage des identités détectées :

1. **Reconnaissance faciale :** Si le système utilise des **algorithmes de reconnaissance faciale**, chaque visage détecté peut être associé à une **identité unique** (par exemple, un étudiant ou un membre du

personnel). Cette identité peut être stockée sous forme de **métadonnées** dans MongoDB, comprenant des informations telles que le nom, l'image du visage (stockée dans un format sécurisé), et les zones où l'individu a été détecté.

## 2. Structure d'un document d'identité détectée :

```
{  
    "id_individu": "98765",  
    "nom": "John Doe",  
    "image": "image_du_visage_base64",  
    "zones_detectées": [  
        "entrée principale",  
        "salle de réunion",  
        "cuisine"  
    ],  
    "heure": "2024-12-21T09:30:00Z"  
}
```

## 3. Exemple d'application : Dans un environnement scolaire, lorsque l'identité d'un étudiant est détectée par une caméra (par exemple, via reconnaissance faciale ou autre méthode), l'identité et les informations associées peuvent être stockées dans MongoDB. Cela permet de générer des rapports détaillés sur la **présence** et les **mouvements** des individus dans les différentes zones surveillées.

### Conclusion

En résumé, MongoDB offre une flexibilité et une performance exceptionnelles pour stocker des données variées et volumineuses dans un système de surveillance intelligent. Grâce à son architecture orientée document et sa capacité à gérer des données non structurées, MongoDB peut stocker efficacement des vidéos, des journaux d'alertes, des présences et des identités détectées, permettant ainsi une gestion optimale des informations et des événements critiques dans un environnement de sécurité.

## VI.3.2 Sécurisation et organisation pour un accès rapide dans une base de données MongoDB

La **sécurisation** et l'**organisation** des données sont des aspects essentiels pour garantir non seulement la **protection des informations sensibles**, mais aussi un **accès rapide et efficace** aux données dans une base de données comme MongoDB. Dans le cadre d'un système de surveillance intelligent, où les informations doivent être traitées en temps réel (par exemple, vidéos, alertes, présences), ces deux éléments sont cruciaux.

### 1. Sécurisation des données dans MongoDB

La sécurisation des données dans MongoDB repose sur plusieurs mécanismes pour protéger l'intégrité des informations stockées, garantir l'accès approprié, et se conformer aux bonnes pratiques de sécurité.

#### A. Authentification et autorisation

1. **Authentification** : MongoDB permet de mettre en place un système d'**authentification** pour vérifier l'identité des utilisateurs qui se connectent à la base de données. Cela peut inclure l'utilisation de **mots de passe**, de **certificats SSL**, ou d'autres mécanismes d'authentification comme **LDAP** (Lightweight Directory Access Protocol).

- Exemple : Un utilisateur administrateur doit se connecter avec un mot de passe fort ou un certificat validé avant d'avoir accès aux données sensibles.
2. **Rôles et autorisations** : MongoDB permet de définir des **rôles** d'accès pour chaque utilisateur. Chaque rôle détermine ce qu'un utilisateur peut faire avec la base de données, par exemple :
- **Lecteur** : Peut lire les données, mais ne peut pas effectuer de modifications.
  - **Écrivain** : Peut lire et écrire dans la base de données.
  - **Administrateur** : A un contrôle total sur la base de données, y compris la gestion des utilisateurs et des rôles.

Cela permet de limiter l'accès à des informations sensibles, comme les vidéos ou les identités des personnes détectées, aux seuls utilisateurs autorisés.

## B. Cryptage des données

Le cryptage est une méthode essentielle pour protéger les données, notamment lorsque les informations sensibles sont stockées dans la base de données.

1. **Cryptage au repos** : MongoDB propose des fonctionnalités de cryptage pour les **données au repos**, c'est-à-dire les données qui sont stockées sur le disque. Cela garantit que même si un attaquant accède au serveur de base de données, les données seront illisibles sans la clé de décryptage.
2. **Cryptage en transit** : MongoDB prend également en charge le **cryptage en transit** avec SSL/TLS, ce qui signifie que les données échangées entre les clients et le serveur sont cryptées pour éviter qu'elles ne soient interceptées pendant leur transmission.

## C. Sauvegarde et récupération

Pour assurer la disponibilité et l'intégrité des données en cas de défaillance système, MongoDB offre des outils de **sauvegarde**. Les sauvegardes régulières des données critiques (comme les vidéos et les journaux d'alertes) permettent de récupérer les informations en cas d'incident ou d'attaque.

1. **Sauvegardes incrémentielles** : Ces sauvegardes ne stockent que les changements effectués depuis la dernière sauvegarde, ce qui permet d'économiser de l'espace disque tout en garantissant la protection des données.
2. **Répliques et clusters** : MongoDB utilise des **réplicas sets** pour garantir la disponibilité des données même en cas de panne d'un serveur. Les répliques créent des copies des données sur plusieurs serveurs, ce qui permet de récupérer rapidement les informations en cas de problème.

## D. Journalisation et audit

MongoDB permet de mettre en place des mécanismes de **journalisation** et de **audit** pour suivre toutes les actions effectuées dans la base de données, ce qui est particulièrement utile pour détecter toute activité suspecte ou non autorisée.

1. **Audit MongoDB** : Il s'agit d'un module qui enregistre les actions importantes effectuées dans la base de données, comme les modifications des données sensibles, les connexions, ou l'exécution de requêtes.
2. **Exemple d'utilisation** : Si un utilisateur non autorisé tente d'accéder à des données critiques (par exemple, des vidéos sensibles ou des informations d'identité), l'audit enregistrera cette tentative pour une analyse ultérieure.

## 2. Organisation des données pour un accès rapide

Une fois que la **sécurisation** des données est mise en place, il est crucial d'**organiser les données** dans MongoDB de manière optimale pour un **accès rapide et efficace**. Une bonne organisation des données facilite les requêtes, améliore la performance et permet de traiter rapidement les informations en temps réel.

### A. Modélisation des données

La **modélisation des données** dans MongoDB dépend des types de données à gérer. Dans le cas d'un système de surveillance, les données peuvent être volumineuses et complexes (images, vidéos, alertes, etc.), et il est important de les organiser sous forme de **documents** ou de **collections**.

1. **Collections** : Organiser les données dans différentes **collections** permet de les séparer selon leur type. Par exemple, une collection pour les **vidéos**, une autre pour les **alertes**, une autre pour les **présences**, etc. Chaque collection est optimisée pour contenir un type de données homogène.
  - o Exemple :
    - Collection `videos` : Contient les documents avec les informations sur les vidéos capturées (ID, chemin du fichier, métadonnées).
    - Collection `alertes` : Contient les documents avec les informations sur les alertes générées (ID, type, niveau de gravité, date).
    - Collection `presences` : Contient les documents enregistrant les présences des individus (ID, nom, date et heure).

### B. Indexation des données

L'**indexation** est un élément clé pour optimiser les performances des requêtes. MongoDB permet de créer des **index** sur des champs spécifiques afin d'améliorer la vitesse d'accès aux données.

1. **Index sur les identifiants** : Il est courant de créer un index sur des champs comme l'**ID** des vidéos ou des alertes pour accéder rapidement à un document spécifique.
2. **Index sur les dates** : Dans un système de surveillance, les événements sont souvent recherchés par **date**. Créer un index sur la **date de capture** des vidéos ou des **heures d'alerte** permet d'effectuer des requêtes rapides pour accéder aux données les plus récentes.
  - o Exemple de création d'index sur MongoDB :

```
db.alertes.createIndex({ "date": 1 });
```

### C. Partitionnement des données

MongoDB permet de **partitionner** (ou "sharder") les données pour améliorer la performance des grandes bases de données. Le partitionnement permet de diviser les données en plusieurs **segments** répartis sur différents serveurs ou clusters. Cela permet d'accélérer l'accès aux données et d'améliorer la **scalabilité**.

1. **Sharding par zone géographique** : Dans un environnement de surveillance, si plusieurs zones géographiques ou bâtiments sont surveillés, les données peuvent être partitionnées par **zone** pour améliorer l'accès aux informations spécifiques à chaque site.
  - o Exemple : Les vidéos ou les alertes provenant de différentes salles ou bâtiments peuvent être réparties sur plusieurs shards.

### D. Cache des données fréquemment demandées

Enfin, l'utilisation d'un **cache** peut être cruciale pour accéder rapidement aux informations fréquemment demandées, comme les vidéos récentes ou les alertes récurrentes.

1. **Caching des résultats** : Les résultats des requêtes fréquemment utilisées, comme les alertes récentes ou les présences, peuvent être **mis en cache** dans une couche intermédiaire (par exemple, en utilisant **Redis** ou **Memcached**) afin de réduire la charge sur la base de données et accélérer l'accès.

## Conclusion

La sécurisation et l'organisation efficaces des données dans MongoDB sont essentielles pour un système de surveillance intelligent. En sécurisant l'accès via des mécanismes d'authentification, en cryptant les données sensibles et en mettant en place un système d'audit, on garantit la protection des informations. L'organisation des données à travers des collections appropriées, des index, et un partitionnement efficace permet un accès rapide, ce qui est primordial pour un système de surveillance réactif et performant.

### VI.3.3 Analyse des données historiques pour améliorer les modèles d'IA

L'**analyse des données historiques** joue un rôle crucial dans l'amélioration des modèles d'**intelligence artificielle (IA)**, notamment dans des systèmes tels que ceux utilisés pour la surveillance. Ces données fournissent des informations essentielles pour comprendre les tendances, identifier des anomalies et affiner les modèles prédictifs afin de mieux détecter les événements inhabituels ou suspects. Voici une explication détaillée de la manière dont l'analyse des données historiques peut être utilisée pour améliorer les modèles d'IA.

#### 1. Importance des données historiques dans les systèmes d'IA

Les **données historiques** désignent les informations recueillies sur une période passée, comme les vidéos de surveillance, les alertes générées, les mouvements des personnes, les événements suspects, etc. Ces données peuvent être analysées pour extraire des informations utiles qui permettent de mieux comprendre le comportement normal et anormal dans un environnement surveillé.

##### A. Amélioration de la précision des modèles prédictifs

Les modèles d'IA, comme ceux utilisés pour la **détection d'anomalies**, s'améliorent grâce à l'exposition à des **données historiques**. Plus les modèles sont nourris de données historiques représentatives des différents scénarios et comportements, plus leur capacité à **prédir** ou à **détecter des événements suspects** sera précise. Les données historiques permettent donc de « former » l'IA à reconnaître des **patterns** ou des **comportements récurrents**.

##### B. Identification des tendances récurrentes et des comportements normaux

En analysant les données historiques, les modèles d'IA peuvent apprendre ce qui constitue un comportement **normal** ou attendu dans un environnement donné. Par exemple, la fréquence des mouvements dans une zone spécifique à des moments donnés (comme les heures de pointe) peut être utilisée pour apprendre aux modèles à reconnaître ce qui est habituel et ce qui ne l'est pas.

##### C. Détection d'anomalies

Les données historiques permettent aussi de mieux comprendre ce qui constitue une **anomalie**. Par exemple, si un comportement inhabituel est observé à un certain moment de la journée ou dans une zone spécifique (comme

un regroupement de personnes dans un endroit normalement désert), les données passées permettent de définir des **seuils d'anomalie** et d'apprendre à l'IA à détecter ces événements en temps réel.

## 2. Types de données historiques utilisées pour l'amélioration des modèles d'IA

### A. Données de vidéos et images

Les systèmes de vidéosurveillance génèrent une quantité massive de données sous forme d'images ou de vidéos. Ces données peuvent être analysées de plusieurs façons pour entraîner l'IA :

1. **Identification des objets et des personnes** : En analysant des vidéos passées, l'IA peut apprendre à reconnaître certains objets (par exemple, sacs, objets abandonnés) ou des comportements suspects (regroupement de personnes).
2. **Suivi des mouvements** : Les vidéos permettent de suivre le déplacement des personnes et d'analyser des **comportements de mobilité** anormaux. Par exemple, un individu qui se déplace trop rapidement ou à des horaires inhabituels peut être signalé comme suspect.

### B. Données des capteurs (*mouvement, son*)

Les capteurs de mouvement et de son captent des événements dans l'environnement. Ces capteurs génèrent également une grande quantité de données, que l'IA peut utiliser pour identifier des motifs d'activité :

1. **Mouvement** : Les capteurs détectent les déplacements et la vitesse des objets dans l'environnement. Par l'analyse de ces données dans le temps, l'IA peut mieux comprendre les **comportements de groupe** ou identifier des **comportements inhabituels**.
2. **Son** : Les capteurs sonores détectent des bruits spécifiques, comme des voix humaines ou des bruits de chocs. L'IA peut analyser des enregistrements sonores passés pour reconnaître des situations suspectes, comme des cris, des bruits de verre brisé, ou des discussions de groupe dans des endroits où cela ne devrait pas se produire.

### C. Journaux d'alertes et événements

Les systèmes de surveillance génèrent des alertes chaque fois qu'une activité suspecte est détectée. Ces alertes, qui incluent des informations telles que l'heure, le type d'événement (ex. : mouvement détecté, bruit, regroupement), sont des données précieuses pour l'analyse historique.

1. **Analyse des alertes** : En examinant l'historique des alertes, l'IA peut identifier les moments où les alertes étaient trop fréquentes ou inutiles, et ajuster les seuils de déclenchement pour réduire les faux positifs. Par exemple, une alerte de mouvement peut être considérée comme anormale à une heure spécifique ou dans une zone généralement peu fréquentée.

## 3. Techniques d'analyse des données historiques pour améliorer l'IA

### A. Apprentissage supervisé

L'**apprentissage supervisé** repose sur l'utilisation de données étiquetées pour entraîner les modèles d'IA. En analysant les événements passés où les anomalies ont été étiquetées par des humains, l'IA peut apprendre à classer correctement les événements futurs.

- **Exemple** : Si les vidéos historiques montrent un groupe de personnes se rassemblant dans une zone sans autorisation, ce type d'événement sera étiqueté comme **suspect**. L'IA apprendra ainsi à détecter un comportement similaire dans des vidéos ultérieures.

## B. Apprentissage non supervisé

L'**apprentissage non supervisé** est une méthode où l'IA apprend à **déetecter des anomalies sans supervision humaine préalable**. Cela permet de détecter des comportements suspects qui n'ont pas encore été rencontrés.

- **Clustering** : L'IA peut utiliser des techniques de **clustering** pour regrouper des événements similaires dans les données historiques. Si un comportement inhabituel survient dans un groupe de données non identifié, il sera traité comme une **anomalie**.

## C. Réseaux de neurones profonds (Deep Learning)

Les **réseaux de neurones profonds** sont des modèles d'IA capables d'apprendre des représentations complexes des données. L'utilisation de modèles de **deep learning** pour l'analyse des vidéos, des images et des sons historiques permet à l'IA d'améliorer sa capacité à détecter des objets spécifiques, à reconnaître des comportements suspects et à faire des prédictions plus précises.

- **Exemple** : L'IA peut apprendre à identifier non seulement des personnes dans une vidéo, mais aussi leurs comportements (ex. : mouvement rapide, interaction avec d'autres personnes), ce qui permet de détecter des anomalies.

## D. Modèles prédictifs

Les modèles d'IA peuvent utiliser les données historiques pour prédire les événements futurs ou les comportements suspects. Par exemple, si l'IA détecte un certain type d'activité anormale à une heure spécifique chaque jour, elle peut prédire que cet événement se produira à nouveau à l'avenir.

### 4. Application de l'analyse des données historiques dans un système de surveillance

Dans un système de surveillance, l'analyse des données historiques permet d'améliorer les alertes générées par le système, d'affiner les processus de détection et de réduire les faux positifs. Par exemple :

1. **Réduction des faux positifs** : En étudiant les alertes passées et les comportements récurrents, l'IA peut apprendre à ne pas déclencher une alerte pour des comportements normaux.
2. **Réaction proactive** : Une fois qu'une tendance est identifiée dans les données historiques (par exemple, une zone habituellement vide qui devient soudainement occupée), le système peut prendre des mesures préventives pour avertir les opérateurs avant qu'un incident ne se produise.

## Conclusion

L'analyse des données historiques permet d'améliorer considérablement les modèles d'IA, en fournissant des informations contextuelles et des exemples réels d'événements ou de comportements dans un environnement surveillé. En utilisant des techniques d'apprentissage supervisé, non supervisé, et des réseaux de neurones profonds, l'IA devient plus précise dans la détection des anomalies, et peut prédire des événements futurs avec une meilleure fiabilité. Cela renforce l'efficacité du système de surveillance et contribue à la sécurité proactive de l'environnement surveillé.

## CHAPITRE VII : PARTIE DE LA SORTIE

**VII.1** Déclenchement d'alarmes (sonores et visuelles) en cas d'intrusion détectée.

## 1. Alertes et Notifications

Les alertes et notifications jouent un rôle crucial dans les systèmes de surveillance, car elles permettent de réagir rapidement en cas d'incidents ou d'activités suspectes. Lorsqu'une intrusion ou un événement anormal est détecté, il est essentiel d'alerter les responsables de la sécurité ou les utilisateurs concernés, afin qu'ils puissent prendre les mesures nécessaires. Cela peut inclure des **alarms sonores et visuelles**, des notifications par **SMS**, **e-mail**, ou **applications mobiles**. Ce processus de gestion des alertes est une partie essentielle de la chaîne de surveillance, et il repose sur une interaction bien synchronisée entre la détection, l'analyse et la sortie des informations.

### A. Déclenchement des alarmes (sonores et visuelles)

Les **alarmantes sonores et visuelles** sont des outils de communication instantanée qui alertent les responsables de la sécurité ou les personnes présentes dans la zone surveillée en cas d'événements suspects. Voici une explication détaillée sur le **déclenchement de ces alarmes** :

#### 1. Alarme sonore

Une **alarme sonore** est généralement un signal acoustique fort (sirène, buzzer, etc.) qui est émis lorsque le système de surveillance détecte une intrusion ou un comportement suspect. Elle est utilisée pour attirer l'attention immédiatement et alerter les personnes autour de la zone surveillée.

##### 1. Caractéristiques :

- Le son est souvent conçu pour être **strident et percutant**, ce qui permet à l'alarme d'être entendue sur une large distance.
- L'intensité et la durée du son peuvent varier selon la **gravité de l'intrusion** ou de l'événement.
- Parfois, l'alarme sonore peut être configurée pour être activée **en continu** ou **par intermittence**, en fonction de la priorité ou du niveau d'urgence.

##### 2. Fonctionnement :

- Lorsqu'un capteur de **mouvement** ou une **caméra** détecte une **intrusion** (par exemple, une personne entrant dans une zone non autorisée), une **commande est envoyée à un module d'alarme sonore** pour émettre un signal.
  - Certaines alarmes sonores peuvent être **programmées pour durer un certain temps** (ex. : 30 secondes à 1 minute), ou peuvent être **désactivées manuellement** une fois l'incident résolu.
3. **Exemple d'application** : Si un capteur de mouvement détecte un intrus entrant dans une zone interdite (comme une salle d'examen pendant la nuit), l'alarme sonore se déclenche immédiatement pour signaler l'intrusion.

#### 2. Alarme visuelle

Les alarmes **visuelles** sont des dispositifs de signalisation visuelle qui peuvent inclure des **lumières clignotantes**, des **écrans d'affichage**, ou des **projecteurs lumineux**, qui attirent l'attention de manière visible sur l'événement suspect.

##### 1. Caractéristiques :

- Souvent, une **lumière clignotante** ou un **panneau d'affichage** peut être utilisé pour signaler un incident.
- La couleur de la lumière peut varier en fonction du type d'alerte (rouge pour une **intrusion**, verte ou bleue pour un événement résolu).
- Ces lumières ou affichages peuvent être activés localement (sur place) ou à distance (comme dans une salle de contrôle).

2. **Fonctionnement :**
  - Une fois qu'un événement anormal est détecté, l'alarme visuelle peut être activée pour attirer l'attention des personnes dans la zone surveillée ou à proximité. Par exemple, un **flash lumineux** ou un **clignotement rapide** peut signaler qu'une alerte est en cours.
  - Cela peut aussi être utilisé pour **dissuader** une intrusion, en signalant immédiatement à l'intrus qu'il a été repéré.
3. **Exemple d'application :** Lorsqu'un détecteur de mouvement repère une personne en dehors des heures d'accès dans un bâtiment, un **flash lumineux** rouge peut être activé pour indiquer qu'une alerte est en cours, tout en envoyant un signal au système de surveillance pour alerter les agents de sécurité.

## B. Types d'Alertes et Notifications

Outre les alarmes sonores et visuelles, des notifications supplémentaires peuvent être envoyées pour permettre une gestion **à distance** ou **en temps réel** de l'incident. Cela inclut des **notifications mobiles**, **emails** ou encore des messages instantanés. Ces notifications permettent aux utilisateurs autorisés de réagir rapidement, même s'ils ne sont pas à proximité de la zone surveillée.

### 1. *Notifications mobiles (via des applications)*

De nos jours, il est courant d'utiliser des **applications mobiles** pour recevoir des alertes en temps réel. Une notification peut être envoyée directement sur le téléphone mobile des responsables de la sécurité ou des administrateurs via une **application de surveillance**.

1. **Caractéristiques :**
  - La notification peut **inclure un résumé de l'incident**, y compris la date, l'heure, et la localisation de l'événement détecté.
  - L'application peut permettre de **visualiser l'incident** en temps réel, grâce à la transmission des vidéos ou des images capturées par les caméras de surveillance.
  - Le responsable peut alors décider d'agir, comme envoyer du personnel sur place ou désactiver l'alarme.
2. **Exemple d'application :** Un responsable de sécurité reçoit une notification sur son téléphone indiquant qu'une personne a été détectée à une heure inhabituelle dans une zone sécurisée. Il peut alors accéder à la vidéo en direct de la caméra pour évaluer la situation.

### 2. *Notifications par Email ou SMS*

Les alertes peuvent également être envoyées sous forme de **messages par e-mail** ou **SMS**, particulièrement utiles lorsque les responsables de la sécurité ou les administrateurs ne sont pas constamment connectés à une application mobile.

1. **Caractéristiques :**
  - Les **emails** peuvent contenir un résumé détaillé de l'événement, y compris une image ou une vidéo de l'incident, et un lien pour accéder aux informations détaillées sur un portail sécurisé.
  - Les **SMS** peuvent être plus immédiats et succincts, contenant seulement des informations essentielles, telles que la nature de l'incident et un lien vers une application ou un portail.
2. **Exemple d'application :** Lorsqu'un mouvement suspect est détecté dans un entrepôt la nuit, un email ou un SMS est envoyé à l'administrateur de la sécurité, lui indiquant l'événement avec des détails et des instructions sur la façon de vérifier ou intervenir.

### *3. Système de gestion centralisée (surveillance à distance)*

Un **système de gestion centralisée** permet de surveiller en temps réel toutes les alarmes et notifications provenant de différentes zones, offrant une vue d'ensemble sur la situation.

#### 1. Caractéristiques :

- Un **tableau de bord centralisé** affichant toutes les alertes en cours et les événements passés.
- L'intégration d'une **vidéo en direct** pour permettre aux opérateurs de visualiser les zones sous surveillance, en cas d'alerte.

2. **Exemple d'application** : Dans un centre commercial, un opérateur surveille toutes les caméras et reçoit des alertes en temps réel. Si une intrusion est détectée dans une zone non surveillée, l'opérateur peut immédiatement observer la vidéo et envoyer une équipe de sécurité.

#### C. Importance des alertes pour la réactivité et la sécurité

Le déclenchement rapide et efficace des alarmes sonores et visuelles est essentiel pour assurer une **réactivité immédiate** aux intrusions ou aux comportements suspects. Les **alertes mobiles** et **emails** jouent un rôle complémentaire pour garantir que les responsables de la sécurité peuvent intervenir même s'ils ne sont pas sur place. Ces mécanismes contribuent non seulement à **minimiser les risques** et à **limiter les dommages**, mais aussi à améliorer la gestion proactive de la sécurité dans l'ensemble de l'environnement surveillé.

#### Conclusion

Les **alertes et notifications** sont des composants essentiels dans un système de surveillance efficace. Le déclenchement d'alarmes sonores et visuelles permet d'alerter immédiatement les personnes présentes dans une zone, tandis que les notifications mobiles et par email assurent un suivi à distance. Grâce à l'intégration de ces différentes formes de communication, le système permet une gestion rapide et efficace des incidents, renforçant ainsi la sécurité de l'environnement surveillé.

### VII.1.2 Envoi de Notifications en Temps Réel aux Administrateurs via Email/SMS

L'envoi de notifications en temps réel aux administrateurs est une partie fondamentale de tout système de surveillance moderne. Cela permet aux responsables de la sécurité d'être immédiatement informés des événements importants ou des incidents détectés par les systèmes de surveillance, qu'ils soient sur place ou à distance. Ces notifications peuvent être envoyées par **email** et **SMS**, offrant ainsi une flexibilité maximale dans la réception des alertes.

#### A. Principe de Fonctionnement des Notifications en Temps Réel

Lorsqu'un événement suspect ou anormal est détecté par un capteur, une caméra ou un système d'intelligence artificielle (IA), il est crucial que l'administrateur ou la personne responsable de la sécurité soit informé sans délai. Cela permet de réagir rapidement et efficacement pour éviter que la situation ne dégénère.

**Exemple d'événements** pouvant déclencher une notification :

1. **Intrusion dans une zone interdite**.
2. **Comportement suspect** (comme un regroupement de personnes dans une zone non autorisée).
3. **Détection de mouvement** dans des zones normalement vides ou après des heures de travail.
4. **Objet abandonné** détecté dans un lieu sensible (par exemple, un sac à dos laissé sans surveillance).

Le processus d'envoi de notifications suit généralement ces étapes :

1. **Détection** : Un capteur (mouvement, chaleur, son) ou une caméra détecte un événement suspect.
2. **Traitement** : L'événement est traité par un **algorithme d'IA** ou un **système de gestion centralisé**.
3. **Génération de notification** : Si l'événement est jugé important ou suspect, une notification est générée.
4. **Envoi de notification** : La notification est envoyée en temps réel à un administrateur via **email** ou **SMS**.

## B. Envoi de Notifications par Email

### 1. Structure de la Notification par Email

Une notification envoyée par **email** doit contenir des informations précises et utiles pour permettre à l'administrateur de prendre des décisions rapidement. Voici les éléments typiques inclus dans un email de notification :

1. **Objet de l'email** : Un titre concis qui résume l'incident (par exemple, "Intrusion détectée dans la salle 3").
2. **Détails de l'incident** :
  - o **Date et heure de l'événement** : Pour savoir quand l'incident s'est produit.
  - o **Lieu de l'incident** : Indication de l'endroit exact où l'événement a été détecté (par exemple, "zone de stockage").
  - o **Type d'événement** : Une brève description de l'incident (par exemple, "détecté de mouvement dans une zone interdite").
  - o **Capture d'écran ou vidéo** : L'email peut inclure une **image** ou un **clip vidéo** de l'incident, capturé par les caméras, pour que l'administrateur puisse évaluer la situation immédiatement.
3. **Actions recommandées** : Si applicable, des recommandations ou actions à suivre (par exemple, "Vérifier les caméras de surveillance", "Appeler l'équipe de sécurité").

Exemple de contenu d'email :

Objet : Intrusion détectée dans la salle 3 à 22h15

Détails de l'incident :

- **Date/Heure** : 15 décembre 2024, 22h15
- **Lieu** : Salle 3, zone sécurisée
- **Événement** : Mouvement détecté en dehors des horaires autorisés
- **Vidéo** : [lien vers la vidéo de l'incident]

Actions recommandées :

- Vérifier immédiatement l'incident à l'aide des caméras en direct.
- Envoyer une équipe de sécurité sur place.

### 2. Technologie derrière l'envoi de l'Email

L'envoi d'un **email** de notification se fait généralement à travers un **serveur SMTP** (Simple Mail Transfer Protocol). Un programme ou un service (comme Node.js, Python, ou un service de messagerie tiers tel que SendGrid ou Mailgun) est utilisé pour créer l'email et l'envoyer aux administrateurs.

- **Exemple d'implémentation dans Node.js** : Pour envoyer un email avec Node.js, vous pouvez utiliser un module comme **Nodemailer**, qui permet d'envoyer des emails de manière simple.

```
const nodemailer = require('nodemailer');
```

```

// Créer un transporteur SMTP
let transporter = nodemailer.createTransport({
  service: 'gmail',
  auth: {
    user: 'votre-email@gmail.com',
    pass: 'votre-mot-de-passe'
  }
});

// Définir l'email
let mailOptions = {
  from: 'votre-email@gmail.com',
  to: 'admin@example.com',
  subject: 'Intrusion détectée dans la salle 3',
  text: 'Une intrusion a été détectée à 22h15 dans la salle 3. Veuillez vérifier les caméras.',
  html: '<p>Une intrusion a été détectée à <strong>22h15</strong> dans la salle 3. Veuillez vérifier les caméras.</p><video src="video_link" controls></video>'
};

// Envoyer l'email
transporter.sendMail(mailOptions, (error, info) => {
  if (error) {
    console.log('Erreur lors de l\'envoi de l\'email:', error);
  } else {
    console.log('Email envoyé:', info.response);
  }
});

```

### *3. Avantages des Notifications par Email*

1. **Accessibilité à distance** : Les administrateurs peuvent recevoir et consulter les notifications par email partout où ils ont accès à leur boîte de réception.
2. **Archivage des incidents** : Les emails permettent de garder un **historique des alertes**, ce qui peut être utile pour l'analyse des incidents à long terme.
3. **Facilité de mise en place** : La configuration des notifications par email est relativement simple à mettre en œuvre.

## C. Envoi de Notifications par SMS

Les notifications par **SMS** sont idéales pour un envoi rapide et immédiat à des administrateurs ou responsables qui peuvent ne pas consulter leurs emails en temps réel. Les SMS permettent d'alerter plus directement les utilisateurs, car les messages sont souvent lus très rapidement après leur réception.

### *1. Structure du SMS*

Contrairement aux emails, le SMS est plus succinct et direct. Il inclut généralement :

1. **Un résumé court** de l'incident (par exemple, "Intrusion détectée à 22h15 dans la salle 3").
2. **L'action requise** : Par exemple, "Vérifiez immédiatement la caméra".

Exemple de contenu d'un SMS :

Intrusion détectée à 22h15 dans la salle 3. Vérifiez la caméra maintenant.

## 2. Technologie derrière l'envoi des SMS

L'envoi de SMS se fait généralement par l'intermédiaire d'un service **API SMS** comme **Twilio**, **Nexmo**, ou **Plivo**, qui offre une interface pour envoyer des messages depuis une application web ou un serveur.

- **Exemple avec Twilio en Node.js :**

```
const twilio = require('twilio');

// Vos informations Twilio
const accountSid = 'VOTRE_ACCOUNT_SID';
const authToken = 'VOTRE_AUTH_TOKEN';
const client = new twilio(accountSid, authToken);

// Envoyer le SMS
client.messages.create({
    body: 'Intrusion détectée à 22h15 dans la salle 3. Vérifiez la caméra maintenant.',
    from: '+224663919633', // Numéro Twilio
    to: '+224625841484' // Numéro de l'administrateur
})
.then((message) => console.log('Message envoyé :', message.sid));
```

## 3. Avantages des Notifications par SMS

1. **Réception instantanée** : Les SMS sont reçus presque instantanément sur le téléphone mobile de l'administrateur.
2. **Haute visibilité** : Les utilisateurs lisent souvent les SMS immédiatement après les avoir reçus, ce qui garantit que l'incident sera traité sans délai.
3. **Idéal pour les alertes urgentes** : Les SMS sont parfaits pour des alertes critiques ou urgentes nécessitant une réponse rapide.

## D. Conclusion

Les notifications en temps réel via **email** et **SMS** sont essentielles pour permettre aux administrateurs de prendre des mesures immédiates lors d'incidents détectés par les systèmes de surveillance. L'email est plus adapté pour des notifications détaillées et archivées, tandis que le SMS permet une réactivité instantanée. Grâce à ces deux méthodes, les responsables de la sécurité peuvent être alertés de manière efficace, quel que soit leur emplacement.

## VII.2 Actions Automatisées :

### VII.2.1 Activation des Lumières ou Verrouillage des Portes en Cas de Détection d'un Intrus

Les **actions automatisées** dans un système de sécurité moderne jouent un rôle crucial pour réagir rapidement aux incidents détectés, en minimisant les risques et en maximisant l'efficacité des mesures prises. Ces actions sont souvent déclenchées automatiquement par des systèmes intelligents, comme ceux utilisant l'intelligence artificielle (IA), et se produisent dès qu'un intrus ou un événement suspect est détecté.

## A. Principe des Actions Automatisées

Lorsqu'un **intrus** est détecté par les **capteurs** ou les **caméras de surveillance**, des actions physiques telles que **l'activation des lumières** ou **le verrouillage des portes** peuvent être prises automatiquement, en fonction de la nature de l'incident. Ces actions sont conçues pour :

1. **Dissuader l'intrus** de poursuivre son action.
2. **Confirmer la présence d'un incident** et alerter les personnes concernées (ex. : les administrateurs ou le personnel de sécurité).
3. **Sécuriser immédiatement les zones sensibles** en restreignant l'accès ou en attirant l'attention.

Ces actions sont souvent intégrées dans un **système de gestion automatisée** qui prend des décisions en fonction de l'analyse des données provenant des capteurs et des caméras.

## B. Détail des Actions Automatisées

### 1. Activation des Lumières

L'activation des lumières en cas d'intrusion détectée est une mesure de sécurité courante qui vise à plusieurs objectifs :

1. **Dissuader l'intrus** : La lumière soudaine peut effrayer un intrus ou signaler qu'il est observé.
2. **Améliorer la visibilité** : Lorsqu'un intrus est détecté dans une zone, activer les lumières permet d'améliorer la visibilité pour la surveillance (caméras de sécurité) et pour les gardes ou administrateurs qui pourraient intervenir rapidement.
3. **Attirer l'attention** : Dans certains cas, l'activation des lumières peut être un signal pour alerter les personnes présentes dans l'environnement (par exemple, dans un bâtiment ou une salle de classe).

### Mécanisme de fonctionnement :

1. **Détection de l'intrus** : Un capteur de mouvement ou une caméra équipée d'un système de traitement d'image (IA) détecte la présence d'un intrus.
2. **Événement déclencheur** : Lorsque l'intrus est localisé dans une zone sensible, un signal est envoyé au système de gestion pour activer les lumières.
3. **Action physique** : Le **contrôleur de lumières** (souvent un relais connecté à des lumières LED ou à des ampoules intelligentes) est déclenché par le signal pour allumer les lumières dans la zone ciblée.

**Exemple avec un système domotique (smart home)** : Si un mouvement est détecté par un capteur dans un couloir, le système domotique envoie un signal aux **ampoules intelligentes** connectées à un réseau de gestion de la maison pour allumer les lumières, et même ajuster leur intensité.

### 2. Verrouillage des Portes

Le **verrouillage des portes** est une mesure de sécurité efficace pour restreindre l'accès à certaines zones après une détection d'intrusion. Cela permet de sécuriser les espaces sensibles, en particulier dans les environnements où les intrus peuvent être susceptibles de se déplacer rapidement dans le bâtiment.

### Mécanisme de fonctionnement :

1. **Détection de l'intrus** : Une fois que l'intrus est détecté par un capteur (par exemple, un détecteur de mouvement, une caméra de surveillance ou un capteur de porte), l'événement déclenche une action.
2. **Événement déclencheur** : Le système de gestion analyse les données des capteurs et identifie que l'intrus est dans une zone nécessitant une sécurité renforcée (par exemple, près d'une porte).
3. **Action physique** : Le système envoie un signal au **moteur de verrouillage électronique** de la porte pour fermer immédiatement le mécanisme de la serrure, empêchant ainsi l'intrus de pénétrer dans la zone sécurisée.

## **Exemple avec un verrou connecté :**

- Si un intrus est détecté près de l'entrée principale, un **verrou connecté** (par exemple, un verrou intelligent Z-Wave ou Bluetooth) reçoit le signal du système de surveillance pour activer le verrouillage de la porte.

## C. Technologies Utilisées pour les Actions Automatisées

### *1. Capteurs et Caméras*

Les capteurs et caméras jouent un rôle essentiel dans le déclenchement des actions automatisées. Parmi les technologies utilisées :

1. **Capteurs de mouvement** : Ils détectent tout mouvement dans une zone donnée (par exemple, capteurs à infrarouge passif ou capteurs à micro-ondes).
2. **Caméras de surveillance** : Des caméras avec des capacités de détection intelligentes peuvent identifier les intrus sur la base de modèles d'images, de reconnaissance faciale ou de détection de comportements suspects.

### *2. Systèmes de Gestion Automatisée*

Les systèmes qui gèrent les actions automatisées, souvent appelés **systèmes domotiques** ou **systèmes de gestion de la sécurité** (SGS), sont des plateformes logicielles qui traitent les données des capteurs et des caméras et déclenchent les actions appropriées en réponse à des événements. Ces systèmes peuvent être intégrés avec des **API de smart home** (maison intelligente), ce qui permet un contrôle automatisé de divers équipements physiques tels que les lumières, les verrous de portes, ou même les systèmes d'alarme.

Exemple :

- **Plateformes domotiques** comme **Home Assistant** ou **SmartThings** peuvent être programmées pour activer automatiquement des actions (lumières, portes) lorsque des intrusions sont détectées par les caméras ou les capteurs.

### *3. IoT (Internet des Objets) et Automatisation*

L'IoT permet aux différents appareils et capteurs (comme les lumières et les verrous) de communiquer entre eux via des protocoles comme **Wi-Fi**, **Zigbee**, ou **Z-Wave**. Ces technologies facilitent l'automatisation des actions en reliant les objets physiques (verrous, lumières) à un **système centralisé** qui gère les réponses aux incidents.

Exemple de plateforme IoT :

1. **Z-Wave et Zigbee** sont utilisés pour connecter des **serrures intelligentes** et des **ampoules connectées** aux systèmes de sécurité, permettant l'envoi d'ordres de verrouillage ou d'activation des lumières dès qu'un événement est détecté.

## D. Avantages des Actions Automatisées

1. **Réduction des délais de réponse** : Les actions sont déclenchées instantanément sans nécessiter une intervention humaine, réduisant ainsi le temps de réponse face à une menace.
2. **Dissuasion des intrus** : Les actions comme l'allumage des lumières ou le verrouillage des portes peuvent surprendre ou effrayer un intrus, le forçant à fuir.

3. **Sécurisation continue** : Le système fonctionne en continu, garantissant une sécurité constante sans dépendre de la présence ou de l'action d'un garde ou d'un opérateur.
4. **Amélioration de la réactivité** : Lorsque des actions automatisées sont mises en place, elles permettent de sécuriser les zones sensibles (portes, fenêtres) immédiatement, sans attendre un personnel de sécurité ou une décision manuelle.

## E. Conclusion

Les **actions automatisées** telles que **l'activation des lumières** et **le verrouillage des portes** sont des éléments essentiels d'un système de sécurité intelligent. En réagissant immédiatement à la détection d'un intrus, ces actions renforcent la sécurité, réduisent les risques d'intrusion et augmentent l'efficacité des systèmes de surveillance. Elles s'appuient sur des technologies avancées comme l'IA, l'IoT, et des systèmes domotiques pour garantir une réactivité optimale face aux incidents de sécurité.

### VII.2.2 Enregistrement Automatique des Vidéos d'Incidents

L'enregistrement automatique des vidéos d'incidents est une fonctionnalité essentielle dans un système de surveillance moderne, permettant de capturer et de conserver les images ou les vidéos dès qu'une situation anormale ou une intrusion est détectée. Cela permet non seulement de fournir des preuves visuelles en cas d'incident, mais aussi de faciliter l'analyse post-incident et d'assurer une traçabilité des événements pour renforcer la sécurité.

#### A. Principe de l'Enregistrement Automatique des Vidéos

Lorsque le système de surveillance détecte un événement anormal (comme une intrusion, un mouvement suspect ou une activité inhabituelle), il déclenche immédiatement un enregistrement vidéo, qui peut être stocké localement sur un appareil (comme un disque dur ou un serveur), ou dans le cloud pour un accès à distance. L'objectif est de fournir une vidéo de l'incident, permettant de recueillir des informations cruciales pour l'analyse et la prise de décision rapide.

#### B. Mécanisme de Fonctionnement de l'Enregistrement Automatique

##### 1. Détection de l'Incident :

- ✓ **Capteurs de mouvement** ou **caméras intelligentes** (équipées d'IA) détectent des comportements suspects ou anormaux. Cela peut inclure des mouvements dans une zone surveillée, une intrusion dans une zone protégée, ou des changements dans les conditions de la scène (par exemple, un objet abandonné).
- ✓ Une fois que l'événement est détecté, un **signal est envoyé au système de gestion** pour activer l'enregistrement vidéo.

##### 2. Activation de l'Enregistrement Vidéo :

- ✓ Dès que l'événement est validé comme anormal ou suspect, l'enregistrement vidéo commence automatiquement. Cela peut être configuré pour enregistrer en **continu** ou pour **enregistrer uniquement quand un événement est détecté**.
- ✓ Les caméras sont généralement configurées pour **enregistrer en boucle**, ce qui signifie que lorsque l'espace de stockage est presque plein, les anciennes vidéos sont écrasées, mais les vidéos des incidents restent sauvegardées et protégées.
- ✓ L'enregistrement peut être **local** (sur un serveur ou un disque dur interne) ou **dans le cloud** via des solutions comme AWS, Google Cloud, ou des serveurs privés.

##### 3. Stockage des Vidéos :

- ✓ **Stockage local** : Si l'enregistrement est effectué localement, les vidéos sont stockées sur un serveur, un disque dur réseau (NAS) ou un enregistreur vidéo numérique (NVR) qui gère plusieurs caméras.
- ✓ **Stockage dans le Cloud** : Pour des raisons de sécurité et d'accessibilité, beaucoup de systèmes optent pour le stockage **cloud**. Les vidéos peuvent être transférées en temps réel vers des serveurs distants, permettant une consultation à distance et la sauvegarde des données sans risque de perte due à un vol ou à un dommage physique de l'équipement local.

#### 4. Durée d'Enregistrement et Conservation :

- ✓ L'enregistrement peut être configuré pour **capturer des vidéos pendant un certain laps de temps** après le déclenchement de l'événement (par exemple, pendant 30 minutes ou 1 heure après l'incident).
- ✓ Les vidéos sont souvent classées et **étiquetées avec des informations de métadonnées** telles que la **date, l'heure, le type d'incident, et les caméras associées**. Cela permet une organisation facile et une recherche rapide dans la base de données.

#### 5. Protection des Vidéos :

- ✓ Les vidéos associées à des événements détectés sont souvent protégées contre l'écrasement afin de garantir qu'elles ne seront pas effacées lors de l'enregistrement de nouvelles vidéos. Des mécanismes sont mis en place pour **verrouiller** ces vidéos et les rendre inaltérables jusqu'à ce qu'elles soient analysées ou examinées par les administrateurs.

### C. Avantages de l'Enregistrement Automatique des Vidéos

#### 1. Preuves et Traçabilité :

- ✓ L'enregistrement des vidéos fournit des **preuves visuelles** cruciales pour les enquêtes après un incident, permettant de documenter précisément ce qui s'est passé.
- ✓ Les vidéos peuvent être utilisées pour des **rapports de sécurité**, pour les **forces de l'ordre** ou pour des **audits internes**.

#### 2. Amélioration de la Sécurité :

- ✓ L'enregistrement automatique améliore la **réactivité** du système de surveillance. Dès qu'un incident est détecté, les vidéos sont immédiatement capturées et stockées, réduisant le risque de perdre des informations cruciales.
- ✓ Cela permet également d'**analyser des événements** après coup pour évaluer les actions entreprises et ajuster les systèmes de sécurité en fonction des incidents.

#### 3. Réduction du Risque de Fausse Détection :

- ✓ En enregistrant continuellement et en réagissant immédiatement aux événements détectés, le système réduit les risques de manquer une **détérioration progressive des conditions de sécurité** (comme un cambriolage en cours ou un comportement étrange non détecté).

#### 4. Accès à Distance et Flexibilité :

- ✓ Lorsque les vidéos sont stockées dans le **cloud**, les administrateurs peuvent accéder aux vidéos à distance via une plateforme sécurisée. Cela permet d'examiner les vidéos de manière flexible, sans être limité par la géographie ou le temps.

#### 5. Analyse post-incident :

- ✓ Les vidéos peuvent être utilisées pour une analyse détaillée du **comportement des intrus**, de l'**évolution de l'incident**, et des **zones vulnérables** pour ajuster les stratégies de sécurité.
- ✓ Elles servent également à identifier les **failles dans le système de sécurité**, comme des zones non couvertes ou des comportements inattendus.

### D. Technologies Utilisées pour l'Enregistrement Automatique

#### 1. Caméras de Surveillance :

- ✓ **Caméras IP** : Ces caméras sont capables de transmettre des données vidéo en temps réel via Internet. Elles sont souvent utilisées dans les systèmes de surveillance modernes pour leur **flexibilité et facilité d'intégration** dans des solutions cloud.
  - ✓ **Caméras avec stockage local** : Ces caméras peuvent enregistrer sur une **carte SD** ou un **disque dur local**, et sont particulièrement adaptées aux systèmes de surveillance autonomes ou de petite échelle.
2. **Serveurs et NVR (Network Video Recorder)** :
- ✓ Les **enregistreurs vidéo numériques (NVR)** gèrent plusieurs caméras en même temps, en stockant et en traitant les vidéos. Ils permettent d'accéder à l'enregistrement en temps réel et d'assurer la gestion de l'espace de stockage.
  - ✓ Un **serveur de stockage local** peut également être utilisé pour héberger les vidéos enregistrées, mais cela nécessite des ressources de gestion supplémentaires.
3. **Solutions Cloud** :
- ✓ Des **services cloud** comme **Google Cloud**, **AWS**, ou **Microsoft Azure** peuvent être utilisés pour enregistrer et gérer des vidéos de manière sécurisée. Ces solutions offrent une **scalabilité élevée**, une **accessibilité à distance**, et des **mécanismes de sécurité** pour protéger les données sensibles.
4. **Logiciels de Surveillance Vidéo** :
- ✓ Des logiciels de gestion vidéo comme **Blue Iris**, **Milestone XProtect**, ou **Luxriot** permettent la gestion et la visualisation des vidéos, et intègrent souvent des fonctionnalités avancées comme la détection de mouvement, l'analyse en temps réel et la recherche vidéo basée sur des critères spécifiques.

## E. Conclusion

L'**enregistrement automatique des vidéos d'incidents** est un élément clé pour renforcer la sécurité dans des environnements sensibles. En permettant une réaction immédiate et une collecte systématique de preuves, ce processus facilite la gestion des incidents, l'analyse des événements et la mise en œuvre de mesures correctives. Grâce à des technologies avancées telles que les caméras IP, les systèmes de stockage dans le cloud, et l'intégration de logiciels d'analytique vidéo, les systèmes modernes offrent une surveillance proactive et réactive efficace.

### Interface Utilisateur :

#### Affichage en Temps Réel des Flux Vidéo et des Alertes

L'interface utilisateur (UI) est une partie essentielle d'un système de surveillance vidéo, car elle permet aux administrateurs, surveillants ou responsables de la sécurité de suivre les événements en temps réel, d'interagir avec les vidéos capturées, et de répondre rapidement aux alertes de sécurité. L'affichage des flux vidéo et des alertes sur l'interface est crucial pour garantir que les utilisateurs aient toutes les informations nécessaires pour prendre des décisions éclairées. Voici une explication détaillée de la manière dont ce système fonctionne.

#### Affichage en Temps Réel des Flux Vidéo

##### 1. Visualisation en Direct des Caméras de Surveillance

- ✓ L'interface utilisateur doit permettre la visualisation en temps réel des flux vidéo provenant de plusieurs caméras de surveillance.
- ✓ Chaque caméra peut être affichée dans une grille ou sous forme de miniatures (thumbnails), de manière à ce que l'utilisateur puisse rapidement voir plusieurs caméras en simultané.

- ✓ L'affichage des flux vidéo peut être configuré pour être plein écran ou réduit selon les besoins de l'utilisateur, avec une option pour basculer entre les caméras en cliquant sur une miniature.
2. Flux Vidéo en Direct (Live Stream)
    - ✓ Les caméras envoient les données vidéo à l'interface en temps réel via des protocoles de streaming vidéo comme RTSP (Real-Time Streaming Protocol) ou HLS (HTTP Live Streaming).
    - ✓ Ces flux vidéo peuvent être compressés pour optimiser la bande passante, tout en maintenant une qualité d'image suffisante pour l'analyse visuelle.
    - ✓ L'interface doit afficher les vidéos sans délai significatif, garantissant une surveillance fluide et continue des zones sensibles.
  3. Contrôle et Interactivité
    - ✓ Les utilisateurs peuvent mettre en pause, zoomer ou dézoomer sur une vidéo en direct pour observer un événement spécifique.
    - ✓ Des options comme l'enregistrement manuel ou la capture d'images peuvent être intégrées pour permettre à l'utilisateur de sauvegarder des moments spécifiques en temps réel.
    - ✓ Il peut y avoir des options pour changer la vue en mode quadrillage (plusieurs caméras affichées à la fois) ou en mode plein écran pour une surveillance plus concentrée sur une seule caméra.

## B. Affichage des Alertes et Notifications

1. Notifications en Temps Réel
  - ✓ Lorsque le système de surveillance détecte un événement anormal (comme une intrusion, un mouvement suspect, ou un comportement inhabituel), une alerte doit être immédiatement envoyée à l'utilisateur via l'interface.
  - ✓ Ces alertes peuvent être sous forme de pop-up, notifications visuelles (telles que des bannières clignotantes ou des indicateurs de couleur sur la vidéo ou l'écran de l'interface), ou même sonores pour attirer l'attention des utilisateurs.
2. Affichage des Alertes sur l'Interface
  - ✓ Les alertes sont généralement affichées sous forme de boîtes de dialogue ou de listes d'événements sur le côté de l'écran, en indiquant la nature de l'alerte, la caméra concernée, l'heure, et une description de l'événement détecté.
  - ✓ Les alertes peuvent également être classées par niveau de gravité. Par exemple, les intrusions peuvent avoir une couleur rouge, les mouvements suspects une couleur jaune, et les événements non critiques une couleur verte.
3. Historique des Alertes
  - ✓ L'interface peut offrir un historique des alertes dans une section dédiée, permettant aux utilisateurs de revoir les événements passés, de les analyser et de prendre des mesures sur la base de ces données.
  - ✓ Ces alertes sont cliquables, de manière à ce que l'utilisateur puisse accéder à la vidéo en lien avec l'alerte, pour une analyse plus détaillée du contexte de l'incident.
4. Priorité et Gestion des Alertes
  - ✓ Les alertes peuvent être priorisées selon l'importance de l'événement détecté. Par exemple, une alerte liée à une intrusion pourrait prendre priorité par rapport à une alerte liée à un mouvement de fond ou à un événement non critique.
  - ✓ Les administrateurs ou les responsables de la sécurité peuvent être alertés par des moyens supplémentaires, tels que des emails, des SMS, ou des notifications push sur des appareils mobiles, en plus de l'affichage sur l'interface.

## C. Interaction et Réaction en Temps Réel

1. Réponse à une Alerté

- Une fois qu'une alerte est affichée sur l'interface, les utilisateurs peuvent répondre immédiatement à l'événement. Par exemple, activer une alarme sonore, verrouiller les portes, allumer les lumières ou activer des caméras supplémentaires pour suivre l'incident de manière plus détaillée.
  - L'interface permet d'enregistrer une réponse à l'incident ou de marquer l'alerte comme résolue ou en cours.
2. Contrôle des Éléments de Sécurité
    - Certains systèmes permettent à l'interface de contrôler des dispositifs de sécurité supplémentaires, tels que des barrières physiques, des systèmes d'éclairage automatisés, des sirènes, ou même de démarrer l'enregistrement de vidéos supplémentaires selon l'incident détecté.
  3. Vue des Zones Sensibles
    - L'interface peut aussi permettre une surveillance des zones sensibles en affichant en surbrillance les zones qui sont les plus exposées ou les plus susceptibles d'être intrusées.
    - Cette fonctionnalité peut être utilisée en combinaison avec des cartes interactives de l'environnement ou des plans de bâtiments, permettant aux utilisateurs de voir en un coup d'œil les zones qui nécessitent une attention particulière.

## D. Gestion de l'Interface Utilisateur

1. Accessibilité et Configuration
  - L'interface doit être ergonomique et facile à utiliser, avec des options de personnalisation pour répondre aux besoins spécifiques des utilisateurs (par exemple, un mode sombre, une disposition flexible des fenêtres, etc.).
  - L'interface peut être responsive pour s'adapter à différents appareils, comme les tablettes et les smartphones, permettant ainsi aux responsables de sécurité de surveiller les vidéos et alertes à tout moment et depuis n'importe quel endroit.
2. Multi-Utilisateurs et Permissions
  - Selon les rôles des utilisateurs (par exemple, administrateurs, surveillants, ou agents de sécurité), l'interface peut permettre une gestion des droits d'accès pour déterminer qui peut voir quelles caméras, qui peut répondre aux alertes, et qui peut configurer les paramètres du système.
  - Un tableau de bord personnalisé pourrait être mis à disposition, permettant à chaque utilisateur de suivre uniquement les informations pertinentes pour son rôle.

## E. Technologies et Outils Utilisés

1. Technologies de Streaming Vidéo :
  - RTSP ou HLS sont souvent utilisés pour diffuser les flux vidéo en direct.
  - Les WebRTC ou WebSockets peuvent être utilisés pour une transmission vidéo en temps réel et avec une faible latence, spécialement pour les systèmes nécessitant un suivi immédiat.
2. Frameworks Frontend :
  - Des frameworks comme React, Vue.js, ou Angular sont utilisés pour construire des interfaces utilisateurs interactives et dynamiques.
  - TailwindCSS ou Bootstrap peuvent être utilisés pour la mise en page et la réactivité de l'interface.
3. Plateformes de Notification :
  - Firebase Cloud Messaging (FCM) ou des services d'emailing/SMS comme Twilio ou SendGrid peuvent être utilisés pour envoyer des notifications en temps réel à l'administrateur ou aux utilisateurs.
4. Gestion de l'Authentification et des Permissions :
  - OAuth ou JWT peuvent être utilisés pour gérer les sessions utilisateur et garantir la sécurité des accès à l'interface.

## F. Conclusion

L'affichage en temps réel des flux vidéo et des alertes sur une interface utilisateur joue un rôle crucial dans la surveillance moderne, permettant une gestion proactive de la sécurité. Une bonne interface doit fournir des flux vidéo de haute qualité, des alertes claires et visibles, et des outils d'interaction efficaces pour une réaction rapide. Grâce à l'intégration de technologies avancées et de bonnes pratiques de conception UI, l'interface devient un outil puissant pour assurer la sécurité des lieux surveillés.

### Possibilité d'Interagir avec les Alertes : Confirmer ou Rejeter une Anomalie

Dans un système de surveillance et de sécurité utilisant des caméras, des capteurs et de l'intelligence artificielle (IA), la gestion des alertes est une partie cruciale du processus de surveillance. Lorsque le système détecte une anomalie, une alerte est générée. Toutefois, ces alertes doivent être validées par un opérateur humain pour éviter les faux positifs et garantir que les mesures appropriées soient prises. L'interaction avec les alertes (confirmer ou rejeter une anomalie) permet aux utilisateurs de gérer efficacement ces événements de sécurité.

#### 1. Vue d'Ensemble du Processus d'Interaction avec les Alertes

Lorsqu'une alerte est générée par le système de surveillance (par exemple, une intrusion détectée par la caméra ou un comportement suspect), l'utilisateur a la possibilité de :

1. **Confirmer** si l'anomalie est réelle et nécessite une action.
2. **Rejeter** si l'anomalie est une fausse alerte ou si l'événement ne présente aucun danger.

Cette interaction permet de réduire le nombre de fausses alertes (faux positifs) et de se concentrer sur les menaces réelles.

#### 2. Affichage de l'Alerte sur l'Interface Utilisateur

Les alertes sont affichées de manière visible et claire sur l'interface utilisateur, souvent sous forme de **fenêtres pop-up**, de **bannières** ou de **cartes d'alerte** qui contiennent les informations suivantes :

1. **Nature de l'événement** (par exemple, intrusion, mouvement suspect, regroupement de personnes, objet abandonné).
2. **Caméra ou capteur responsable** de la détection de l'anomalie.
3. **Heure et date** de l'événement.
4. **Description détaillée** de l'incident (par exemple, "mouvement rapide détecté dans la zone 3", "comportement inhabituel enregistré").
5. **Lien vers la vidéo** associée à l'alerte pour examiner l'événement.

#### 3. Interactions avec l'Alerte

##### a) Confirmer une Alerta

Lorsqu'une alerte est reçue et qu'un utilisateur pense que l'événement détecté est légitime (c'est-à-dire qu'il s'agit d'une **intrusion réelle** ou d'un **comportement suspect**), il peut :

1. **Cliquer sur le bouton "Confirmer"** : Ce bouton permet à l'utilisateur de valider que l'alerte est correcte. Une fois confirmée, l'alerte peut déclencher une action (par exemple, alarme sonore, verrouillage des portes, notification aux autorités).

2. **Accéder à la vidéo en temps réel ou enregistrée** : L'utilisateur peut regarder la vidéo associée à l'alerte en temps réel ou lire l'**historique vidéo** pour mieux comprendre le contexte de l'événement et s'assurer de la pertinence de l'alerte.
3. **Classification de l'Alerte** : Une fois confirmée, l'alerte peut être classée comme une **alerte réelle** (intrusion, acte criminel, etc.) et le système peut passer à une série d'actions automatisées ou envoyer des notifications aux personnes concernées.
4. **Prendre des mesures immédiates** : En fonction de la nature de l'alerte, l'utilisateur peut être invité à prendre des mesures spécifiques, telles que :
  - ✓ **Activer une alarme.**
  - ✓ **Verrouiller ou déverrouiller des portes.**
  - ✓ **Notifier les administrateurs ou les responsables** via email/SMS.
  - ✓ **Démarrer l'enregistrement** ou la capture d'images supplémentaires de l'événement pour documenter l'incident.

#### *b) Rejeter une Alerta*

Dans certains cas, l'alerte peut être un **faux positif** ou un événement qui ne nécessite pas d'intervention. L'utilisateur peut alors **rejeter l'alerte**. Voici les étapes associées :

1. **Cliquer sur le bouton "Rejeter"** : Ce bouton permet à l'utilisateur de signaler que l'alerte est une fausse alerte ou qu'elle n'est pas suffisamment critique pour être traitée. L'alerte est ensuite supprimée de la liste des événements en cours ou marquée comme **non pertinente**.
2. **Système de justification** : Dans certains systèmes, l'utilisateur peut être invité à fournir une **justification** pour rejeter l'alerte (par exemple : "faux positif", "mouvement causé par un animal", "événement planifié").
3. **Amélioration des modèles d'IA** : Après avoir rejeté une alerte, les utilisateurs peuvent être invités à indiquer la raison de cette fausse alerte (par exemple, "mouvement causé par une personne portant un sac lourd"), ce qui permet de **réentraîner le modèle d'IA** pour éviter que ce type de fausse alerte se reproduise à l'avenir.

#### *c) Enregistrement de l'Alerte comme Résolue ou Ignorée*

Une fois qu'une alerte est confirmée ou rejetée, elle peut être marquée comme **résolue** ou **ignorée** dans le système :

1. **Alerte Résolue** : Si l'alerte a été confirmée comme valide et que les mesures appropriées ont été prises, elle est enregistrée comme **résolue**, et une **action corrective** ou de sécurité est effectuée.
2. **Alerte Ignorée** : Si l'alerte est rejetée, elle peut être simplement ignorée et **archivée** dans le système comme **faux positif** ou **non pertinent**, afin que les administrateurs puissent y revenir si nécessaire.

### 4. Impact sur les Modèles d'IA

1. **Amélioration Continue** : Chaque interaction avec les alertes (qu'elles soient confirmées ou rejetées) offre des **données supplémentaires** au système. Ces interactions peuvent être utilisées pour **améliorer les modèles d'IA** de détection d'anomalies, réduisant ainsi les faux positifs et améliorant la précision des futures alertes.
2. **Feedback Utilisateur** : Les utilisateurs qui rejettent les alertes peuvent fournir des retours utiles qui permettent d'affiner les algorithmes de détection en prenant en compte des contextes spécifiques à l'environnement surveillé (par exemple, un mouvement causé par un employé ou un objet inoffensif).

## 5. Précautions et Limites

1. **Humanisation de l'analyse** : Les systèmes de surveillance assistée par IA doivent toujours garder un aspect **humain dans la validation des alertes** pour éviter une dépendance excessive à la machine, qui pourrait manquer des nuances contextuelles.
2. **Formation des Utilisateurs** : Les utilisateurs doivent être formés pour interpréter correctement les alertes, éviter les décisions impulsives et suivre une procédure standardisée de gestion des alertes (confirmer, rejeter, enquêter).
3. **Réponse rapide** : Les systèmes doivent garantir que l'interaction avec les alertes soit **rapide et intuitive**, permettant aux utilisateurs de confirmer ou rejeter une anomalie en un temps réduit, afin de pouvoir agir en temps réel en cas de menace.

## 6. Conclusion

L'interaction avec les alertes est essentielle pour assurer que le système de surveillance fonctionne de manière fiable et efficace. Permettre aux utilisateurs de **confirmer ou rejeter** une anomalie est crucial pour minimiser les **faux positifs** et garantir que les mesures de sécurité appropriées sont prises. En intégrant ces interactions dans l'interface utilisateur de manière fluide et intuitive, le système peut améliorer à la fois la réactivité et la précision des alertes, tout en permettant d'adapter continuellement les modèles d'IA pour mieux répondre aux besoins spécifiques de l'environnement surveillé.

## Historique des Incidents pour Analyse Ultérieure

L'**historique des incidents** est une composante essentielle dans un système de surveillance moderne. Il permet non seulement de **documenter** et de **suivre les événements de sécurité** au fil du temps, mais aussi de **fournir des informations précieuses pour les analyses futures**. Cela aide à améliorer la sécurité, à identifier les tendances, et à ajuster les paramètres de surveillance pour des performances optimales.

Voici une explication détaillée de l'importance de l'historique des incidents, de son fonctionnement et des bénéfices qu'il offre.

### 1. Collecte et Stockage des Incidents

Les incidents détectés par les capteurs et caméras de surveillance sont enregistrés sous forme de **données structurées** dans une **base de données**. Cette base de données conserve une trace complète des événements, y compris les informations suivantes :

1. **Date et Heure** : Le moment exact où l'incident a été détecté.
2. **Type d'Incident** : S'il s'agit d'une intrusion, d'un comportement suspect, d'un objet abandonné, d'un mouvement inhabituelle, etc.
3. **Identifiants des Caméras et Capteurs** : Quelle caméra ou capteur a capturé l'incident (par exemple, caméra 1, capteur de mouvement 2).
4. **Zone Concernnée** : L'emplacement spécifique où l'incident a été détecté, comme la porte d'entrée, le couloir, une fenêtre.
5. **Caractéristiques de l'Incident** : Les détails concernant l'événement (mouvement rapide, présence humaine, objet abandonné, etc.).
6. **Médias Associés** : Vidéos, images ou enregistrements audio associés à l'incident.
7. **Actions Entreprises** : Si l'alerte a été confirmée ou rejetée, et quelles actions ont été entreprises en réponse (verrouillage de porte, envoi d'une notification, alarme sonore, etc.).

8. **Utilisateur Responsable** : L'identifiant de l'opérateur ou de l'administrateur ayant confirmé ou rejeté l'alerte.

## 2. Accès à l'Historique des Incidents

L'historique des incidents doit être facilement accessible aux **responsables de la sécurité** ou aux **administrateurs** pour une consultation et une analyse approfondie. Voici comment les données peuvent être organisées et consultées :

1. **Interface d'Affichage** : Un tableau de bord ou une interface dédiée permet de **visualiser tous les incidents passés** dans un format clair. L'interface peut afficher une **chronologie** des événements, triée par date et heure, ou permettre de rechercher des incidents spécifiques par type, zone, ou caméra.
2. **Filtres de Recherche** : Des filtres peuvent être appliqués pour **rechercher des incidents spécifiques**, par exemple, en fonction du type d'incident (intrusion, comportement suspect), de la zone géographique (entrée, zone de stockage), ou de l'état de l'alerte (confirmée, rejetée).
3. **Accès aux Médias Associés** : Pour chaque incident, des liens peuvent être fournis pour accéder aux vidéos ou images enregistrées. Cela permet de revoir les événements en question pour une analyse plus détaillée.

## 3. Analyse des Incidents

L'historique des incidents est une mine d'informations pour l'analyse des comportements et des tendances de sécurité. Les données peuvent être utilisées de plusieurs manières :

### a) Détection de Tendances et Modèles Comportementaux

Les administrateurs peuvent analyser les incidents historiques pour **détecter des motifs récurrents** ou des **tendances** dans les événements de sécurité. Par exemple, si des intrusions ont lieu fréquemment à des heures spécifiques ou dans certaines zones (comme près de l'entrée ou près des fenêtres), cela peut indiquer un **point faible dans la sécurité** qui nécessite une attention particulière.

Les **comportements suspects récurrents**, comme des mouvements inhabituels dans des zones normalement vides ou un regroupement de personnes dans des espaces non autorisés, peuvent également être identifiés et surveillés plus étroitement dans le futur.

### b) Identification des Zones Vulnérables

En analysant les incidents historiques, il est possible de **repérer les zones les plus vulnérables**. Par exemple, si plusieurs incidents sont détectés autour d'une porte ou d'une fenêtre, cela peut suggérer que cette zone est particulièrement exposée à des intrusions et nécessite une meilleure surveillance, avec éventuellement l'ajout de caméras supplémentaires ou de capteurs plus sensibles.

### c) Ajustement des Paramètres de Surveillance

En étudiant les types d'incidents et leur fréquence, l'équipe de sécurité peut ajuster les **paramètres du système de surveillance**. Par exemple :

1. **Sensibilité des capteurs** : Si le système détecte trop de faux positifs, les paramètres de sensibilité des caméras ou capteurs peuvent être ajustés pour ne capturer que les événements les plus significatifs.

2. **Alertes et Notifications** : Si certaines alertes sont jugées moins pertinentes, les **notifications peuvent être personnalisées** pour ne signaler que les incidents réellement importants.

#### d) Apprentissage Automatique et Amélioration des Modèles IA

Les données historiques peuvent être utilisées pour **former et améliorer les modèles d'IA** du système de surveillance. En utilisant des **techniques d'apprentissage supervisé**, les incidents passés (par exemple, les fausses alertes et les vrais incidents) peuvent être utilisés pour améliorer la capacité du système à faire des prédictions plus précises sur de futurs événements.

### 4. Suivi et Rapports

L'historique des incidents permet également de générer des **rapports détaillés** pour un suivi continu de la sécurité. Les rapports peuvent inclure des **analyses de fréquence** des incidents, des **réponses aux alertes**, et des statistiques sur le **temps de réponse**.

1. **Rapports périodiques** : Des rapports peuvent être générés sur une base quotidienne, hebdomadaire ou mensuelle pour fournir aux administrateurs un aperçu complet de la performance du système de surveillance.
2. **Mesure de la Réactivité** : Le temps de réponse aux alertes (temps entre la détection d'une anomalie et l'action entreprise) peut être mesuré pour identifier les points d'amélioration dans la gestion des incidents.

### 5. Sécurisation et Accès

Les informations relatives à l'historique des incidents sont sensibles et doivent être protégées pour assurer la confidentialité et l'intégrité des données. Cela inclut :

1. **Contrôle d'accès basé sur les rôles** : Seuls les utilisateurs autorisés (administrateurs ou responsables de la sécurité) doivent avoir accès à ces données.
2. **Sauvegarde régulière des données** : L'historique des incidents doit être **sauvegardé régulièrement** pour éviter toute perte de données.
3. **Cryptage des données sensibles** : Les vidéos, images et informations personnelles liées aux incidents doivent être cryptées pour garantir leur sécurité.

### 6. Conclusion

L'historique des incidents joue un rôle central dans l'analyse continue et l'amélioration des systèmes de surveillance. Non seulement il permet de mieux comprendre et de documenter les événements de sécurité passés, mais il est également essentiel pour identifier les **tendances et zones vulnérables**. Cette analyse continue permet de rendre le système de surveillance plus **réactif, précis et adapté** aux défis futurs.

#### VII.3.4 Exemple d'Intégration : De la Détection à la Réaction

Dans ce processus, nous allons illustrer les étapes clés d'un système de surveillance intelligent qui détecte, analyse et réagit à un incident de sécurité, en l'occurrence, l'intrusion d'un individu dans une zone surveillée. Ce système utilise des capteurs et des caméras, couplés à des algorithmes d'intelligence artificielle (IA), pour fournir une surveillance proactive et efficace.

## 1. Détection → Capture d'un Intrus

### *Étape 1 : Activation des capteurs et caméras*

La première étape du processus de détection consiste à activer les capteurs et caméras installés dans la zone surveillée. Ce système utilise :

1. **Capteurs de mouvement** : Qui détectent tout mouvement dans des zones spécifiques comme les portes, fenêtres ou zones sensibles.
2. **Caméras de surveillance (y compris des caméras infrarouges)** : Pour capturer des images ou vidéos en temps réel de la zone surveillée, même dans l'obscurité.

Les caméras sont programmées pour **capturer des images** en continu ou selon les mouvements détectés par les capteurs. Lorsqu'un mouvement est détecté, la caméra se déclenche pour **enregistrer la scène en vidéo**.

### *Étape 2 : Détection d'un intrus*

Lorsqu'un **intrus** entre dans une zone sensible ou active un capteur, la caméra et le système de détection capturent cet événement. La caméra enregistre la scène et commence à envoyer les données à un module de traitement pour une analyse plus approfondie.

## 2. Analyse → Identification via IA

### *Étape 1 : Envoi des données au module d'analyse*

Les images ou vidéos capturées sont immédiatement envoyées à un **module d'analyse basé sur l'intelligence artificielle (IA)**. Ce module peut être alimenté par des algorithmes avancés d'apprentissage automatique, spécialement entraînés pour traiter des données visuelles.

### *Étape 2 : Reconnaissance de l'intrus*

Le module IA procède à l'analyse des images ou vidéos pour identifier s'il s'agit d'un **intrus** ou non. L'IA utilise deux principales techniques :

1. **Reconnaissance faciale** : Si un visage est visible dans la vidéo, l'IA peut tenter de **comparer ce visage à une base de données** de personnes autorisées (par exemple, les employés ou les étudiants). Si le visage n'est pas reconnu, l'IA marque l'événement comme suspect.
2. **Détection d'objets** : Si l'intrus porte un objet suspect ou se trouve dans une zone interdite, l'IA peut identifier cet objet et l'associer à une alerte de **comportement suspect**.

Les critères d'analyse sont les suivants :

1. **Intrusion dans une zone interdite** : Si l'intrus pénètre une zone où son accès est restreint.
2. **Comportement suspect** : Un comportement comme un regroupement de personnes dans une zone non autorisée ou des mouvements brusques et rapides qui ne correspondent pas aux habitudes.

### *Étape 3 : Catégorisation de l'événement*

L'IA **catégorise** l'événement comme **normal** ou **suspect**. En cas de détection d'un comportement suspect ou d'une intrusion, l'alerte est générée et l'événement est marqué pour une attention immédiate.

### 3. Sortie → Envoi d'une Alerta et Enregistrement de la Vidéo

#### Étape 1 : Envoi d'une alerte

Une fois que l'intrusion est identifiée et catégorisée comme suspecte, le système **envoie immédiatement une alerte** aux administrateurs ou responsables de la sécurité. L'alerte est envoyée par les moyens suivants :

1. **Email/SMS** : Un message est envoyé en temps réel aux administrateurs ou à la personne en charge, avec tous les détails de l'incident, y compris l'heure de l'intrusion, la zone concernée, et un lien vers les vidéos capturées.
2. **Notification dans l'interface utilisateur** : Une notification apparaît également dans l'interface du système de surveillance, affichant l'incident et la vidéo associée.

#### Étape 2 : Enregistrement vidéo automatique

En parallèle, une **vidéo de l'incident** est enregistrée et stockée dans la base de données pour un accès futur. Cela permet de garder une trace des événements et de disposer d'une preuve vidéo en cas de besoin (par exemple, pour des enquêtes futures).

1. La vidéo peut être stockée localement ou dans une **base de données cloud** sécurisée pour un accès rapide et une récupération facile.
2. L'enregistrement peut être complété par des **métadonnées** associées (date, heure, zone géographique, type d'incident).

#### Étape 3 : Actions automatisées

En fonction de la configuration du système, des **actions automatisées** peuvent être déclenchées immédiatement en réponse à l'intrusion :

1. **Activation des alarmes sonores et visuelles** : Pour alerter le personnel ou les personnes dans la zone que l'intrusion a été détectée.
2. **Activation de la sécurité physique** : Par exemple, le **verrouillage des portes** ou l'**activation de lumières** dans des zones spécifiques pour dissuader l'intrus.

### 4. Récapitulation des Étapes de l'Intégration

Voici un récapitulatif détaillé des étapes du système :

1. **Détection** : Le capteur détecte un mouvement ou un objet suspect et active la caméra pour capturer une vidéo.
2. **Analyse** : L'IA analyse la vidéo en temps réel pour identifier l'intrus, soit par reconnaissance faciale, soit par détection d'objets.
3. **Sortie** :
  - **Alerte** : Envoi immédiat de notifications (email/SMS) aux administrateurs et affichage de l'alerte dans l'interface utilisateur.
  - **Enregistrement vidéo** : Stockage des vidéos capturées pour une consultation ultérieure.
  - **Actions automatisées** : Déclenchement de mesures de sécurité telles que l'activation des alarmes, le verrouillage des portes, ou l'activation de lumières.

# Capteur de température et d'humidité DHT22



DHT22

# Introduction

La température est un paramètre environnemental fondamental dans tout espace fermé, en particulier dans des environnements comme les salles de classe universitaires, où le confort des étudiants et du personnel est crucial. Une température trop élevée peut provoquer de l'inconfort, affecter la concentration des étudiants, ou même causer des problèmes liés à la santé. À l'inverse, une température trop basse peut rendre l'environnement désagréable et inefficace pour l'apprentissage. C'est pourquoi la surveillance constante de la température est essentielle pour maintenir un environnement de travail optimal.

Le capteur de température DHT22, également connu sous le nom de [AM2302](#), est l'un des capteurs les plus utilisés pour la mesure de la température et de l'humidité. Il permet de recueillir des données fiables pour ajuster automatiquement la température d'une salle de classe en fonction des conditions ambiantes et des paramètres définis.

Ce capteur est particulièrement adapté à des applications comme la gestion de la climatisation ou du chauffage dans les bâtiments intelligents. En combinant le DHT22 avec un microcontrôleur comme Arduino, il devient possible d'automatiser le contrôle de la température de manière efficace, en intégrant ce système à une plateforme IoT (Internet des Objets).

Dans cette section, nous allons explorer en détail les caractéristiques techniques du DHT22, pourquoi cette technologie a été choisie, son intégration avec un microcontrôleur, le protocole de communication utilisé, et enfin, comment il peut être relié à un système de climatisation pour réguler la température dans une salle de classe.

## Caractéristiques du Capteur DHT22

Le DHT22 est un capteur numérique de température et d'humidité qui offre une grande précision et une fiabilité dans une large gamme de conditions environnementales. Voici ses principales caractéristiques techniques :

Plage de mesure de la température : De -40°C à 80°C Précision de  $\pm 0,5^{\circ}\text{C}$  dans la plage de mesure de 0°C à 50°C.

Plage de mesure de l'humidité :

De 0 à [100%](#) d'humidité relative.

- Précision de [±2-5%](#) RH (humidité relative). Temps de réponse: Le capteur fournit des données toutes les 2 secondes environ, ce qui est largement suffisant pour des applications de surveillance en temps réel.

Consommation d'énergie: Faible consommation d'énergie, ce qui permet de l'utiliser dans des applications alimentées par batterie.

Sortie numérique :

Le DHT22 fournit des données numériques via un signal de communication One-Wire, ce qui signifie qu'il peut transmettre les données de température et d'humidité sur une seule ligne de données. Interface :

- One-Wire: Le DHT22 utilise un protocole simple de communication numérique appelé One-Wire, qui permet de transmettre les données via une seule ligne de communication entre le capteur et le microcontrôleur.

Dimensions :

- Taille compacte, ce qui le rend facile à intégrer dans différents types d'applications.

Alimentation :

- Le capteur fonctionne avec une tension d'alimentation entre 3,3V et 6V, ce qui le rend compatible avec des microcontrôleurs comme Arduino.

Pourquoi Choisir le DHT22 ?

Le DHT22 a été choisi pour plusieurs raisons liées à sa facilité d'utilisation, sa précision, et ses caractéristiques adaptées à l'application envisagée dans ce projet.

1. Précision et Plage de Mesure:

2. - Le DHT22 offre une large plage de mesure de la température et de l'humidité, avec une précision de  $\pm 0,5^{\circ}\text{C}$  pour la température, ce qui est suffisant pour des applications comme la régulation de la température dans une salle de classe.

- De plus, la plage de mesure de l'humidité ([0-100%](#)) permet de surveiller l'humidité ambiante, ce qui est également important pour le confort des utilisateurs.

## 2. Facilité d'Intégration :

- Le DHT22 est simple à utiliser avec un microcontrôleur Arduino. La communication One-Wire simplifie le câblage et permet de réduire le nombre de fils nécessaires pour connecter le capteur à un microcontrôleur.

- Ce capteur ne nécessite que quelques lignes de code pour récupérer et traiter les données de température et d'humidité.

## 3. Consommation Énergétique Faible :

- Le DHT22 est idéal pour les systèmes alimentés par batterie ou à faible consommation, grâce à sa faible consommation énergétique. Cela permet de l'utiliser dans des applications où l'efficacité énergétique est cruciale.

## 4. Rentabilité :

- Par rapport à d'autres capteurs de température plus complexes et coûteux, le DHT22 offre un excellent rapport qualité-prix pour des applications de surveillance de la température et de l'humidité.

### - 5. Robustesse et Fiabilité :

- Ce capteur est robuste et fiable, avec une performance stable dans un large éventail de conditions environnementales. Cela le rend idéal pour une utilisation dans des environnements comme les salles de classe, où les conditions peuvent varier au fil de la journée.

## Lien avec le Microcontrôleur (Arduino)

Le DHT22 est facilement intégrable avec des microcontrôleurs comme Arduino. Le lien entre le capteur et le microcontrôleur repose sur une connexion simple et un protocole de communication numérique One-Wire.

## 1. Connexion physique :

Le DHT22 a trois broches : VCC (alimentation), GND (masse), et DATA (données).

La broche VCC est connectée à la sortie 5V ou [3.3V](#) du microcontrôleur (en fonction de la tension d'alimentation requise).

- La broche GND est connectée à la masse du microcontrôleur.

- La broche DATA est connectée à l'une des entrées numériques de l'Arduino ou de l'ESP32.

## 2. Communication One-Wire:

- Le DHT22 utilise le protocole One-Wire, ce qui signifie qu'une seule ligne de données est utilisée pour transmettre à la fois la température et l'humidité.

Un résistor de tirage de 10 kΩ est généralement utilisé entre la ligne de données et l'alimentation pour garantir une communication stable.

## 3. Code Arduino pour le DHT22 :

Le code pour lire les données du DHT22 est relativement simple. Voici un exemple de code de base

cpp

```
#include <DHT.h>
#define DHTPIN 2 // Pin de données
#define DHTTYPE DHT22 // Définir le type de capteur (DHT22)
DHT dht(DHTPIN, DHTTYPE); // Initialisation du capteur
void setup() {
    Serial.begin(9600);
    dht.begin();
}
void loop() {
    // Attendre quelques secondes entre chaque lecture
    delay(2000);
    // Lire la température et l'humidité
    float h = dht.readHumidity();
    float t = dht.readTemperature();
```

```

// Vérifier si les lectures échouent et afficher un message d'erreur
if (isnan(h) || isnan(t)) {Serial.println("Échec de la lecture du capteur DHT") return}
// Afficher les valeurs de température et d'humidité
Serial.print("Température: ");
Serial.print(t);
Serial.print(" °C\t");
Serial.print("Humidité: ");
Serial.print(h);
Serial.println(" %");
}

```

#### 4. Rôle du Microcontrôleur :

5. Le microcontrôleur (comme l'Arduino ou l'ESP32) joue le rôle de récepteur des données envoyées par le DHT22. Après avoir récupéré les valeurs de température et d'humidité, le microcontrôleur peut :

Afficher les valeurs sur un écran LCD ou un dashboard en temps réel.

- Comparer la température à un seuil prédéfini et activer un relais pour contrôler un système de climatisation ou chauffage.

Protocole de Communication Utilisé

Le DHT22 utilise le protocole One-Wire pour la transmission des données. Ce protocole permet de transmettre les informations sur la température et l'humidité via une seule ligne de données, ce qui simplifie le câblage et réduit le nombre de pins nécessaires sur le microcontrôleur.

Caractéristiques du protocole One-Wire :

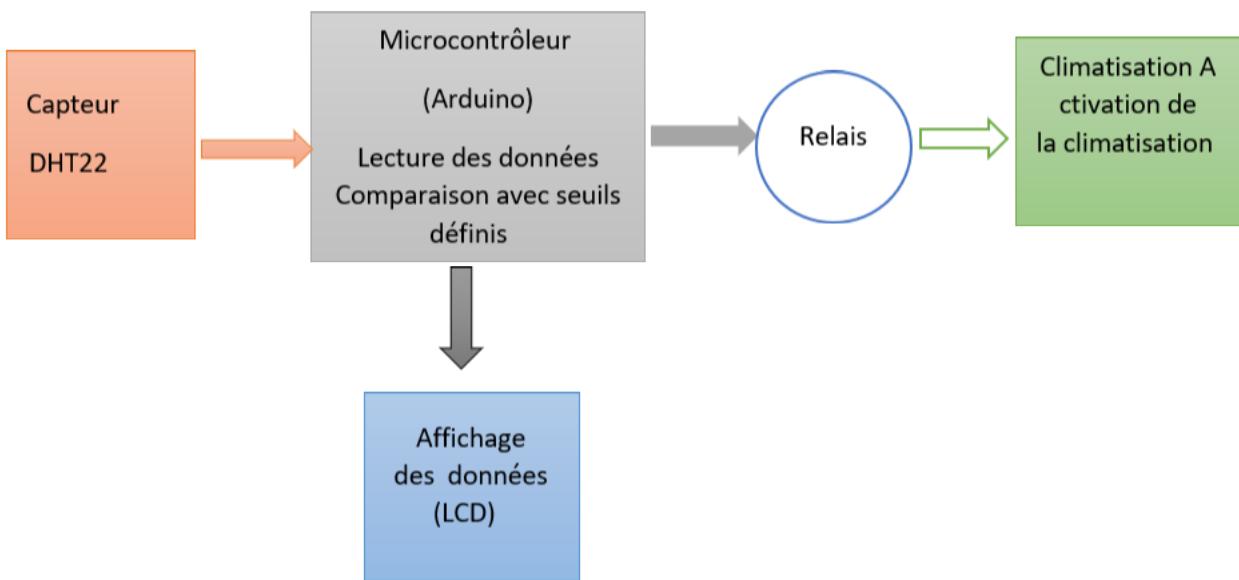
- Un seul fil de données pour communiquer.
- Faible consommation d'énergie.
- Supporte des distances de transmission jusqu'à 20 mètres (dans certaines conditions).

Schéma Synoptique : DHT22 avec Microcontrôleur et Climatisation

Voici un schéma synoptique qui montre l'interconnexion entre le DHT22, le microcontrôleur (Arduino), et un système de climatisation :

**Schéma Synoptique : DHT22 avec Microcontrôleur et Climatisation**

#### 1.Schéma Synoptique



## Schéma synoptique du DHT22

### Explication du Schéma Synoptique

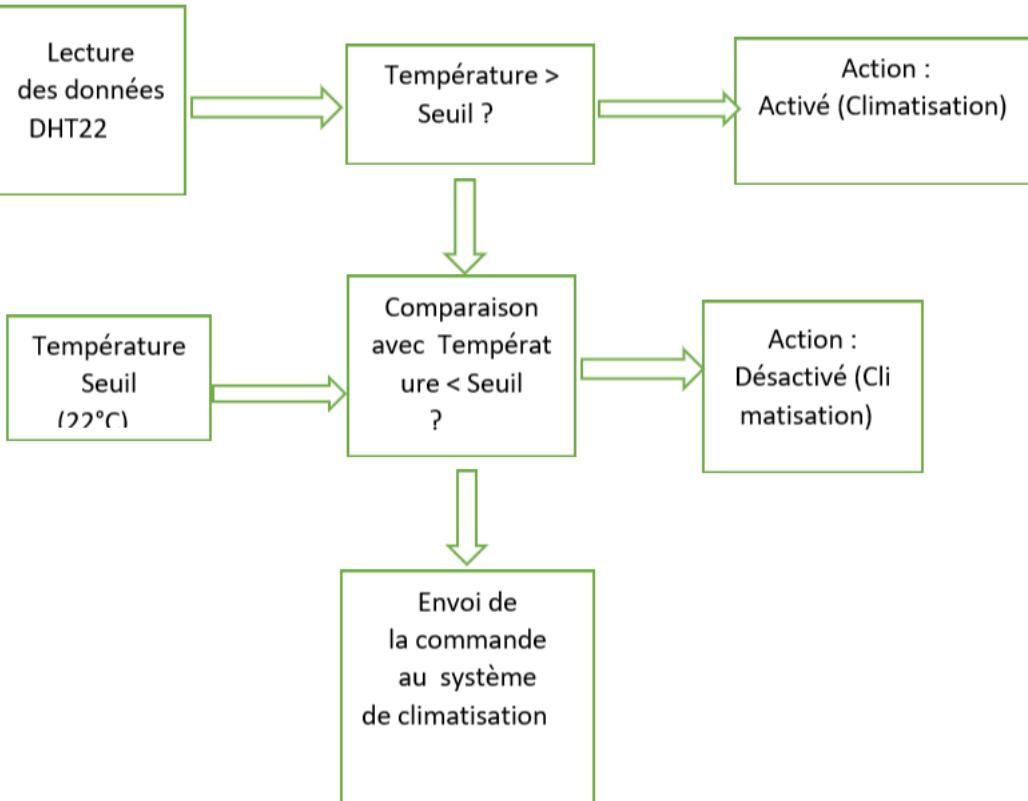
Le schéma synoptique représente l'architecture globale du système de détection de température dans le cadre de la gestion automatique du confort thermique dans une salle de classe. Ce schéma montre les relations entre le capteur DHT22, le microcontrôleur (Arduino) et le système de climatisation.

1. **Capteur DHT22 :** Le capteur de température DHT22 est connecté à un microcontrôleur (comme Arduino) pour fournir des données en temps réel sur la température et l'humidité ambiante dans la salle de classe.
2. **Microcontrôleur :** Le microcontrôleur joue un rôle central en recevant les données du capteur DHT22. Il est programmé pour effectuer un traitement sur ces données :
  - Lire la température et l'humidité envoyées par le DHT22.
  - Comparer ces valeurs avec des seuils définis (par exemple, une température cible de 22°C).
  - Si la température dépasse le seuil (par exemple, 25°C), le microcontrôleur enverra un signal au système de climatisation pour activer le refroidissement.

3. **Système de Climatisation :** Le système de climatisation ou de chauffage est activé par une commande envoyée par le microcontrôleur. Cela permet de réguler automatiquement la température pour maintenir un environnement confortable dans la salle de classe.

### 2. Schéma Fonctionnel :

Gestion Automatique de la Température. Le schéma fonctionnel illustre le flux de données et les actions entreprises en fonction des entrées et des conditions. Ce schéma montre comment l'information circule depuis la lecture des données du capteur jusqu'à l'action finale prise par le synoptique



## Schéma fonctionnel du DHT22

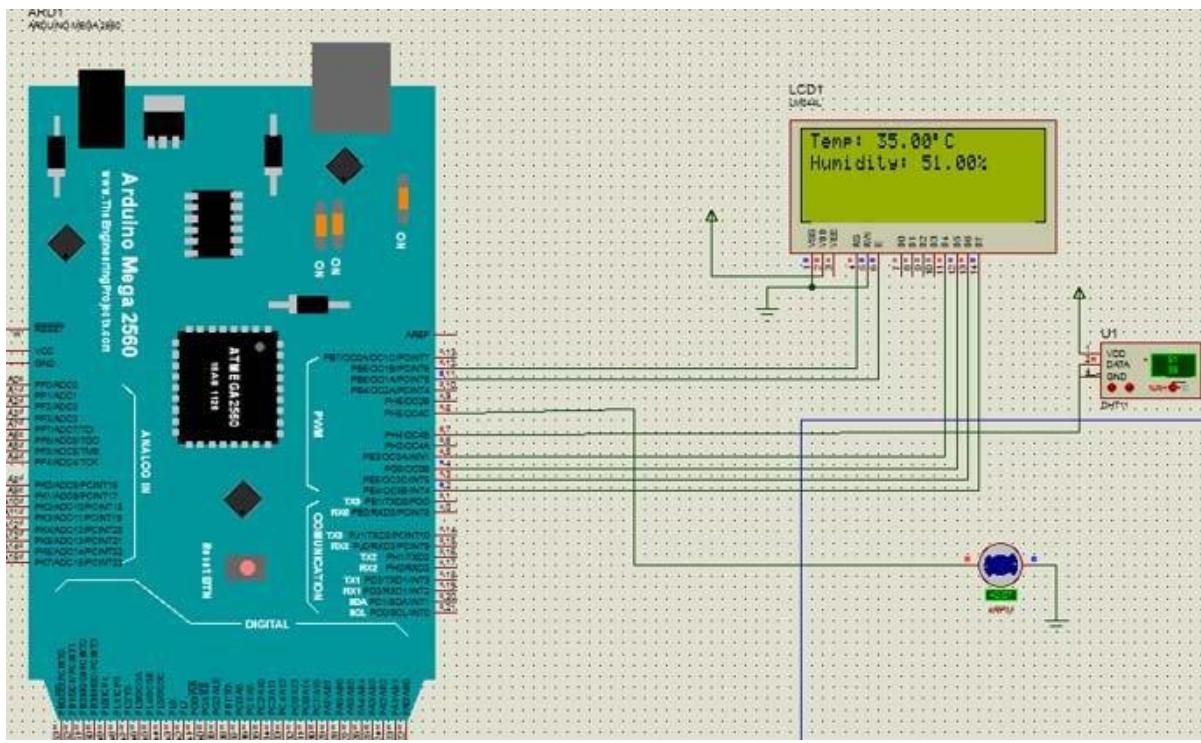
### Explication du Schéma Fonctionnel :

1. Lecture des données du DHT22 : Le microcontrôleur lit périodiquement la température et l'humidité du capteur DHT22. Ces données sont envoyées sous forme numérique au microcontrôleur.
2. Comparaison avec Seuils : Une fois les données récupérées, le microcontrôleur compare la température mesurée avec un seuil défini. Si la température est supérieure à un certain seuil (par exemple, 25°C), le système prendra une action en activant la climatisation.
3. Action : Activation/Désactivation de la Climatisation : En fonction de la comparaison avec les seuils, l'action sera soit d'activer, soit de désactiver le système de climatisation, permettant ainsi un ajustement automatique du climat dans la salle.

### 3. Schéma Électronique :

Connexion du DHT22 avec Arduino et le Système de Climatisation

Le schéma électronique montre comment les composants matériels sont physiquement connectés pour assurer le bon fonctionnement du système de détection et de contrôle climatique. Ce schéma inclut le DHT22, le microcontrôleur (Arduino) et un relais pour contrôler la climatisation.



**Schéma Électronique**

## Explication du Schéma Électronique :

### 1. Capteur DHT22 :

- VCC : Le DHT22 est alimenté par le 5V ou [3.3V](#) provenant du microcontrôleur.

- GND : La broche de masse (GND) du DHT22 est reliée à celle du microcontrôleur.

- DATA : La broche de données du DHT22 est connectée à un pin numérique (par exemple, D2) du microcontrôleur (Arduino ou ESP32). Cette connexion permet la communication entre le capteur et le microcontrôleur via le protocole One-Wire.

### 2. Microcontrôleur (Arduino) :

- Le microcontrôleur reçoit les données de température et d'humidité du DHT22 via la ligne DATA.

- Le microcontrôleur compare la température lue avec un seuil défini et prend une décision : si la température dépasse le seuil, il envoie une commande au relais pour activer la climatisation.

### 3. Relais de Climatisation :

- Le relais est utilisé pour contrôler l'alimentation de la climatisation. Le pin IN du relais reçoit la commande du microcontrôleur pour activer ou désactiver le système de climatisation.

- Le relais fonctionne comme un interrupteur électronique : il permet de fermer ou ouvrir le circuit de la climatisation (en fonction de l'état du relais).

### 4. Système de Climatisation :

- Le système de climatisation (ou de chauffage) est contrôlé par le relais. Lorsqu'un signal est reçu pour activer la climatisation, le relais ferme le circuit et alimente le système de climatisation.

# Conclusion

Le capteur DHT22 est un excellent choix pour surveiller la température et l'humidité dans les salles de classe. Sa précision, sa facilité d'intégration avec des microcontrôleurs comme Arduino et ESP32, et son faible coût en font un capteur idéal pour des applications de régulation climatique dans des environnements intelligents. Grâce à la communication One-Wire, il est simple à intégrer et à utiliser dans des systèmes IoT.

Dans le cadre de ce projet, le DHT22 permet de contrôler en temps réel la température d'une salle de classe, en envoyant des données précises au microcontrôleur, qui peut ensuite agir en fonction des seuils définis, comme l'activation d'un système de climatisation ou de chauffage pour maintenir un environnement optimal. Ce système de détection contribue ainsi à la création d'un environnement d'apprentissage confortable et à une gestion énergétique optimisée.

Le schéma synoptique, fonctionnel et électronique offre une vue complète du système de détection de température et de régulation climatique. Voici un résumé de chaque Schéma Synoptique: Il montre les relations globales entre les différents composants du système, à savoir le capteur DHT22, le microcontrôleur (Arduino) et le système de climatisation. Schéma Fonctionnel : Il illustre le flux de données entre les composants et les actions qui sont prises en fonction des conditions mesurées (température). Il montre comment le système réagit à une température élevée ou basse et comment il ajuste la climatisation en conséquence.

Schéma Électronique : Il détaille les connexions physiques entre le capteur DHT22, le microcontrôleur et le relais, illustrant comment le système est câblé pour contrôler la climatisation en fonction des données collectées par le capteur.

# Capteur de gaz et de fumée MQ2 et MQ7

## Introduction

Les risques liés à l'incendie ou aux fuites de gaz dans des environnements tels que les salles de classe universitaires représentent un danger considérable, tant pour la sécurité des personnes que pour la préservation des biens matériels. Des systèmes de détection efficaces sont donc essentiels pour garantir un environnement sûr, où des actions préventives peuvent être prises avant qu'une situation ne devienne critique.

Le capteur de fumée ou de gaz MQ-2 et le MQ-7 sont des capteurs largement utilisés pour détecter la présence de gaz inflammables, de fumée, ou de monoxyde de carbone (CO), qui sont des indicateurs potentiels d'incendie ou de fuite de gaz. Ces capteurs sont particulièrement utiles dans les systèmes de sécurité automatisés, où ils permettent une détection rapide des risques et une réaction appropriée pour protéger les occupants du bâtiment. Dans cette section, nous allons explorer en détail les caractéristiques techniques des capteurs MQ-2 et MQ-7, les raisons pour lesquelles ces technologies ont été choisies, leur lien avec les microcontrôleurs (Arduino), le protocole de communication utilisé, ainsi que leur intégration dans un système d'alerte précoce en cas de fuite de gaz ou d'incendie.

Notre objectif se résume par : concevoir un système d'alarme incendie à base d'ARDUINO des détecteurs de fumée, flamme et des alarmes sonore et visuel dont l'ensemble est lié à un système d'envoi de messages téléphonique permettant d'informer les personnes concernées et se trouvant à distance, des différentes situations, par des messages

### Caractéristiques du Capteur MQ-2 et MQ-7

Les capteurs MQ-2 et MQ-7 font partie de la série des capteurs de gaz MQ, qui sont conçus pour détecter différents types de gaz, dont les gaz inflammables, le monoxyde de carbone (CO), la fumée, et d'autres gaz potentiellement dangereux. Ces capteurs sont largement utilisés dans les applications de sécurité domestique, industrielle et commerciale

#### Capteur MQ-2 :

- Plage de mesure des gaz détectés :
  - CO (monoxyde de carbone) : 10 à [1000](#) ppm (parties par million).
  - CH4 (méthane) : [300](#) à [10000](#) ppm.
  - C4H10 (butane) : [200](#) à [10000](#) ppm.
  - Smoke (fumée) : [100](#) à [10000](#) ppm.
  - Propane : [300](#) à [10000](#) ppm.
- Temps de réponse :
  - Moins de 10 secondes pour la détection d'un gaz.
- Alimentation :
  - 5V (peut être alimenté avec une tension de 5V provenant de l'Arduino ou ESP32).
- Sortie :
  - Sortie analogique et numérique pour mesurer la concentration du gaz.

- Applications :
  - Détection de fuites de gaz domestiques, fumée d'incendie, ou détection de gaz inflammables.

## Capteur MQ-7 :

- Plage de mesure :
  - Principalement destiné à la détection du monoxyde de carbone (CO) : 20 à [2000](#) ppm.
- Temps de réponse :
  - Environ 30 secondes pour la détection du CO.
- Alimentation :
  - Fonctionne avec une alimentation de 5V.
- Sortie :
  - Sortie analogique pour mesurer la concentration en CO.
- Applications :
  - Surveillance de la qualité de l'air, détection de fuites de monoxyde de carbone dans les espaces fermés.

Pourquoi Choisir le Capteur MQ-2 et MQ-7 ?

Les capteurs MQ-2 et MQ-7 ont été sélectionnés pour leur capacité à détecter une large gamme de gaz, y compris les gaz inflammables, la fumée et le monoxyde de carbone, des menaces courantes dans les bâtiments.

### 1. Polyvalence :

- Le MQ-2 peut détecter plusieurs gaz inflammables (CO, méthane, butane, propane) et de la fumée, ce qui en fait un choix polyvalent pour la détection d'incendie et de fuites de gaz.
- - Le MQ-7 est spécifiquement conçu pour la détection du monoxyde de carbone, un gaz extrêmement toxique souvent produit lors d'incendies ou de fuites dans des espaces mal ventilés.

### 2. Facilité d'intégration :

- Ces capteurs sont faciles à intégrer avec des microcontrôleurs Arduino grâce à leurs sorties analogiques et numériques. Ils sont capables de fournir des données en temps réel pour surveiller la concentration des gaz dans l'air.

### 3. Réactivité rapide :

- Les capteurs MQ réagissent rapidement à la présence de gaz et de fumée, ce qui permet de détecter les incidents potentiels avant qu'ils ne deviennent dangereux.

### 4. Coût abordable :

- Les capteurs MQ-2 et MQ-7 sont peu coûteux par rapport à d'autres capteurs de gaz, ce qui permet de déployer facilement des systèmes de surveillance dans plusieurs zones d'un bâtiment.

### 5. Consommation d'énergie modérée :

- Ces capteurs ont une faible consommation d'énergie, ce qui les rend adaptés à des applications alimentées par batterie ou des dispositifs IoT à faible consommation.

Lien avec le Microcontrôleur (Arduino/ESP32)

Les capteurs MQ-2 et MQ-7 sont facilement connectés à des microcontrôleurs comme l'Arduino. Le lien entre ces capteurs et le microcontrôleur repose sur des connexions simples et un protocole de communication analogique. Voici comment cela fonctionne :

### 1. Connexion physique :

- Le VCC du capteur est relié à la sortie 5V ou [3.3V](#) du microcontrôleur.
- Le GND du capteur est relié à la masse (GND) du microcontrôleur.
- La sortie analogique (A0) du capteur MQ-2 ou MQ-7 est connectée à un pin analogique du microcontrôleur, généralement A0 sur Arduino ou un autre pin analogique sur ESP32.

### 2. Communication analogique :

- Le capteur envoie un signal analogique en fonction de la concentration de gaz détectée. Ce signal est lu par le microcontrôleur via l'entrée analogique.
- Le microcontrôleur convertit cette valeur analogique en une valeur numérique via un convertisseur

analogique-numérique (ADC) intégré.

### 3. Code Arduino pour le MQ-2 ou MQ-7 :

- Un exemple de code simple pour lire les valeurs du capteur de gaz et afficher la concentration de gaz :

```
```cpp
int sensorPin = A0; // Pin analogique pour le capteur MQ
int sensorValue = 0; // Variable pour stocker la lecture du capteur

void setup() {Serial.begin(9600); // Initialisation de la communication série
}

void loop() {
    sensorValue = analogRead(sensorPin); // Lire la valeur du capteur
    Serial.print("Gaz détecté : ");
    Serial.println(sensorValue); // Afficher la valeur sur le moniteur série
    delay(1000); // Attendre 1 seconde avant de lire à nouveau
}
```

```

- Le code lit la valeur analogique du capteur et l'affiche via le port série. Selon les seuils définis, des actions peuvent être entreprises (par exemple, déclencher une alarme ou activer un relais).

### Protocole de Communication Utilisé

Les capteurs MQ-2 et MQ-7 utilisent une sortie analogique pour envoyer les données au microcontrôleur. Le microcontrôleur lit cette sortie analogique via un convertisseur analogique-numérique (ADC).

- Sortie analogique : Le capteur fournit une sortie de tension qui varie en fonction de la concentration du gaz. Cette tension est lue par le microcontrôleur et convertie en une valeur numérique.
- Sortie numérique (optionnelle) : Certains modèles de capteurs MQ disposent également d'une sortie numérique qui peut être utilisée pour activer un seuil de détection. Si la concentration de gaz dépasse un seuil, la sortie numérique passera à HIGH ou LOW, ce qui peut être utilisé pour activer une alarme ou un relais.

Le protocole de communication repose donc sur un signal analogique simple, ce qui le rend facile à implémenter avec des microcontrôleurs comme Arduino.

### Schéma Synoptique :

#### MQ-2 / MQ-7 avec Microcontrôleur et Système d'Alerte

Voici un schéma synoptique montrant l'interconnexion entre le capteur MQ-2 / MQ-7, le microcontrôleur et un système d'alerte (par exemple, une alarme ou une notification).

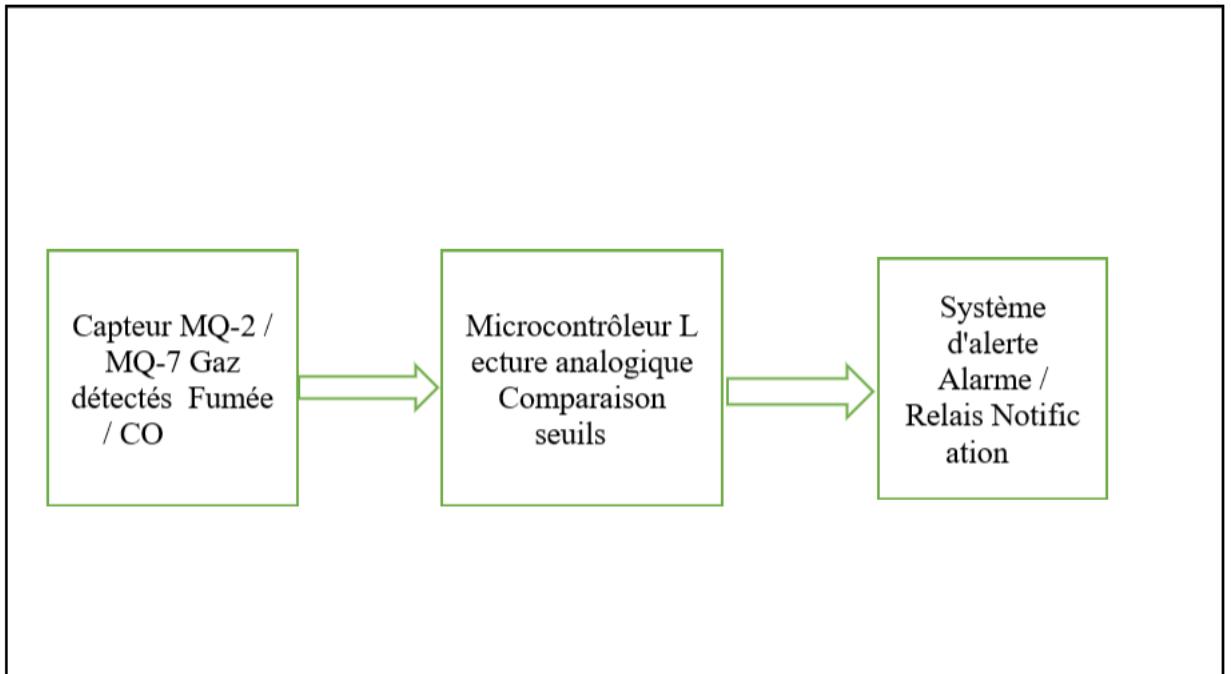
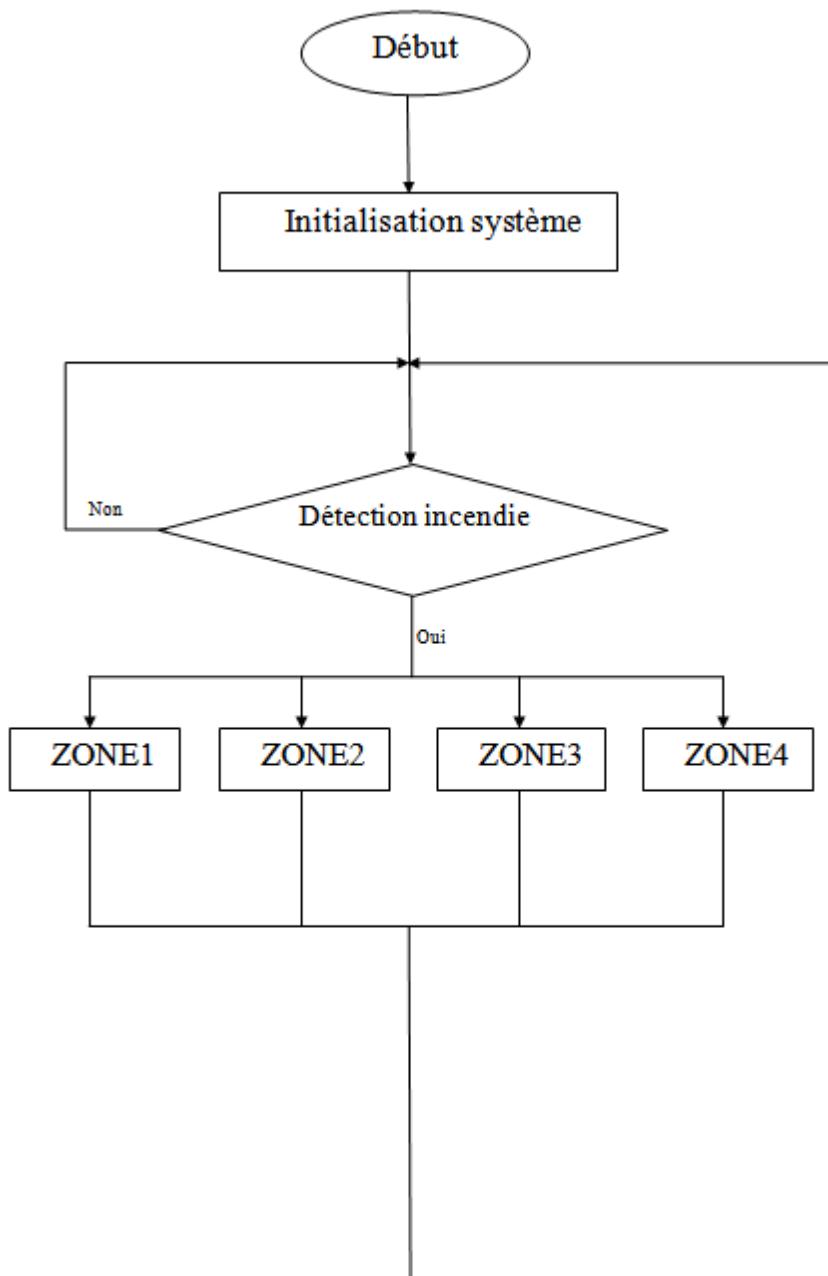


Schéma synoptique du capteur de fumée et de gaz

#### Explication du Schéma Synoptique :

1. Capteur MQ-2 / MQ-7 : Le capteur détecte les gaz (fumée, monoxyde de carbone, gaz inflammables) et génère une sortie analogique en fonction de la concentration du gaz détecté.
2. Microcontrôleur : Le microcontrôleur (Arduino) lit la sortie analogique du capteur, compare la valeur mesurée avec des seuils définis dans le programme, et prend une décision en fonction de ces valeurs.
3. Système d'alerte : Si la concentration de gaz dépasse un seuil critique, le microcontrôleur active un système d'alerte (comme une alarme sonore, un relais pour couper l'alimentation, ou une notification mobile).

#### Schéma Fonctionnel Principal

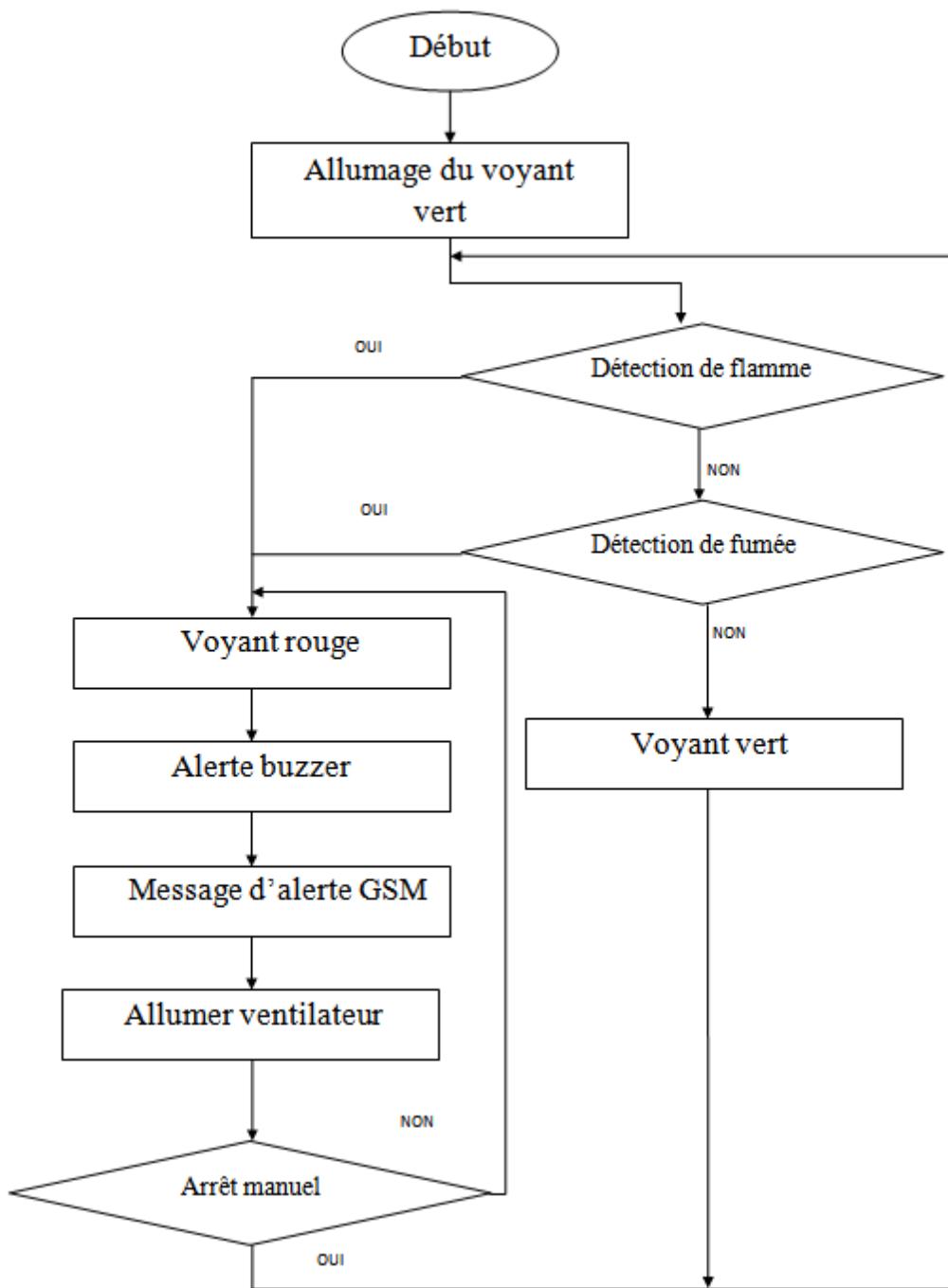


**Schéma Fonctionnel Principal**

L'organigramme principal commence par l'initialisation du système, ensuite il teste s'il y a un état d'un incendie. Dans l'affirmatif il détermine la source, c'est à dire la zone sinistrée, en lisant l'état des détecteurs. A travers cet organigramme on veut exprimer l'idée suivante

Le programme principal scrute en permanence les détecteurs des 4 zones, dès qu'il détecte un état possible de début d'incendie il déclenche l'état d'urgence dans la zone concernée.

### Schéma Fonctionnel d'une zone



### Organigramme de surveillance d'une zone

L'organigramme d'une zone est identique pour les 4 zones, car on a défini la même stratégie de surveillance en équipant les 4 zones par le même matériel. Cet organigramme allume d'abord un allumage vert. Il teste ensuite les

sorties des détecteurs (flamme et fumées), si l'une des conditions du tableau suivant est vraie il déclenche l'état d'urgence décrit précédemment

### Schéma Électronique :

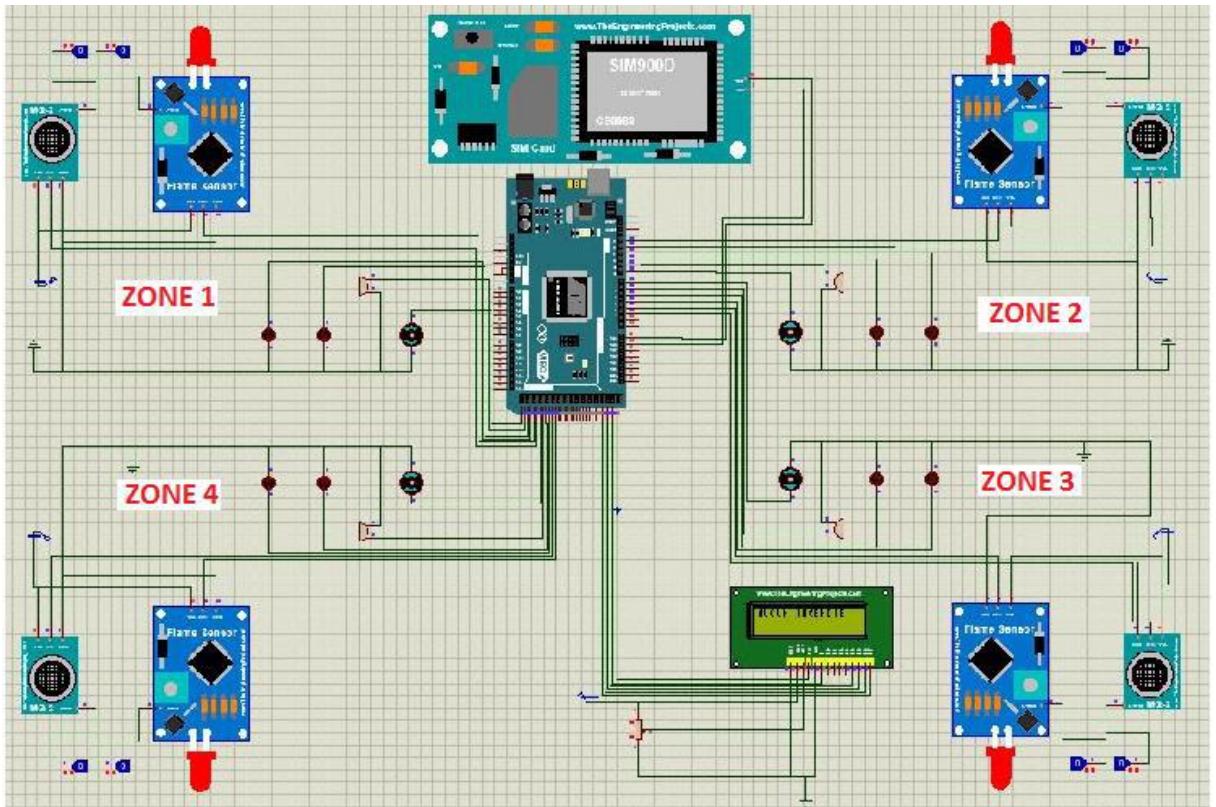


Schéma électrique

### Explication du schéma fonctionnel

Connexion du MQ-2 / MQ-7 avec Arduino et Système d'Alerte

Voici un schéma électronique qui montre comment connecter le capteur MQ-2 / MQ-7 à un microcontrôleur, avec un relais pour contrôler un système d'alerte (par exemple, une alarme).

Le schéma électrique

Matériels installés dans chaque zone :

- détecteur de fumée
- buzzer
- ventilateur

- leds de signalisation (pour malentendants)

Au niveau de la commande centrale on trouve la carte arduino mega avec l'afficheur LCD et le module GSM de communication.

### **Conclusion**

Le choix des capteurs de gaz et de fumée comme éléments clés du système de détection et d'alerte précoce est judicieux pour garantir une couverture complète des risques. Leur utilisation dans un environnement comme un campus universitaire permet non seulement de sécuriser les lieux, mais aussi de répondre de manière proactive aux risques avant qu'ils ne deviennent des menaces réelles. L'intégration de ces capteurs dans un système intelligent permet une gestion optimale de la sécurité, avec des actions automatiques qui assurent la protection des personnes et des biens.

En résumé, le capteur de gaz et le capteur de fumée sont des composants essentiels pour tout système de sécurité moderne. Leur capacité à détecter des anomalies de manière rapide et précise en fait des outils incontournables pour assurer un environnement sécurisé et réactif.

# Capteur de flamme KY-026

Le capteur de flamme KY-026 est un module utilisé pour détecter les flammes en utilisant un récepteur infrarouge qui capte les émissions lumineuses provenant de sources de chaleur. Le module KY-026 mesure l'intensité de la lumière infrarouge émise par le feu dans une plage de longueurs d'onde allant de 760 à 1100 nm, ce qui en fait un choix courant pour les systèmes de détection d'incendie.[22]

## Caractéristique du KY-026

Les caractéristiques du KY-026 sont citées dans le tableau ci-dessous :

|                   |                        |
|-------------------|------------------------|
| Sortie analogique | 0V-5 V                 |
| Sortie numérique  | logique 1 ou logique 0 |

|                         |                |
|-------------------------|----------------|
| Consommation de courant | 10mA           |
| Tension d'alimentation  | 3.3V-5V        |
| Plage de mesure         | 10 à 10000 ppm |
| Dimensions              | 42 mm x 15 mm  |

**Tableau II.13 – Caractéristiques du KY-026**

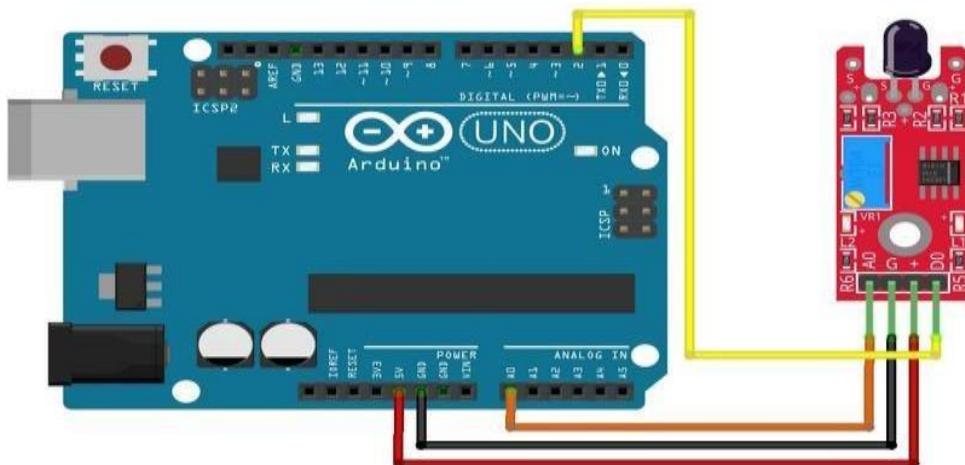
### **Brochage du KY-026**

Le capteur possède quatre broches :

La broche marquée d'un symbole + doit être reliée à une broche d'alimentation 5V du Arduino.

La broche marquée d'un G doit être reliée à une broche GND du Arduino.

La broche A0 est une broche analogique qui renvoie la température mesurée de la flamme La la broche D0, quant à elle, renvoie HIGH si du feu est détecté, et LOW sinon. [23]



**Figure II.21 – KY-026**

### **Alimentation**

Généralement il existe deux sortes d'alimentation :

1) Alimentation filaire :

- Câble USB: lorsqu'on relie la carte à notre PC (à n'importe quel port USB), notre Arduino s'allume. La plupart des modèles requiert un câble USB standard A-B, mais quelques uns (comme le nano) nécessite un câble USB A – mini B.
- Adaptateur: la plupart des adaptateurs type téléphone ou autre, sortant avec une tension continue comprise entre 7 et 12V et ayant un connecteur Jack.

2) Alimentation autonome :

- Pile 9V: Cette pile est idéale car commune, avec un faible encombrement et se trouvant dans la plage de tension recommandée (entre 7 et 12V) via le connecteur jack [28].
- Autres piles: Une solution moins pratique car plus encombrante est de mettre en série des piles de type AA (ou AAA).[26]

## **Langage de programmation**

Un langage de programmation est utilisé pour écrire des instructions qui seront converties en langage machine par un compilateur. Dans le cas d'un programme Arduino, l'exécution des instructions se fait de manière séquentielle. La structure d'un programme Arduino est présentée dans la Figure II.20.[27]

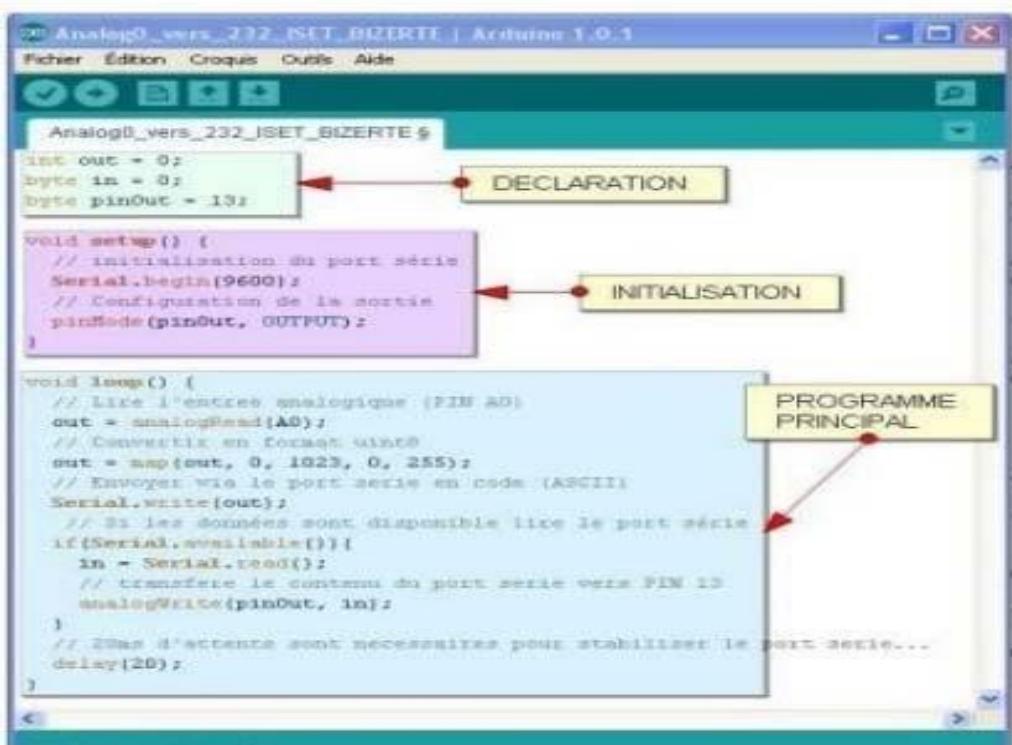


Figure II.23- Structure d'un programme

- La partie déclaration des variables (optionnelle)
- La partie initialisation et configuration des entrées/sorties : la fonction `setup()`
- La partie du programme principale qui s'exécute en boucle : la fonction `loop()`

## Logiciel Arduino

Dans chaque partie d'un programme nous utilisons différentes instructions issues de la syntaxe du langage Arduino

Dans un langage de programmation comme C ou C++, nous utiliserons des instructions pour déclarer et initialiser des variables, définir des fonctions, gérer la mémoire, etc. Dans tous les cas, chaque instruction doit être écrite selon la syntaxe spécifique du langage de programmation utilisé, afin d'être interprétée correctement par l'ordinateur et de réaliser la tâche souhaitée.

Les fonctions principales du logiciel Arduino sont :

- écrire et compiler des programmes pour la carte Arduino.
- se connecter avec la carte Arduino pour y transférer les programmes.
- communiquer avec la carte Arduino.

Cet espace de développement intégré (EDI) dédié au langage Arduino et à la programmation des cartes Arduino comporte (Figure II.21) :

- une BARRE DE MENUS.
- une BARRE DE BOUTONS qui donne un accès direct aux fonctions, essentielles du logiciel et fait toute sa simplicité d'utilisation.

Un EDITEUR (à coloration syntaxique) pour écrire le code de programme, avec onglets de navigation. • une ZONE DE MESSAGES indiquant l'état des actions en cours.

- une CONSOLE TEXTE qui affiche les messages concernant le résultat de la compilation du programme

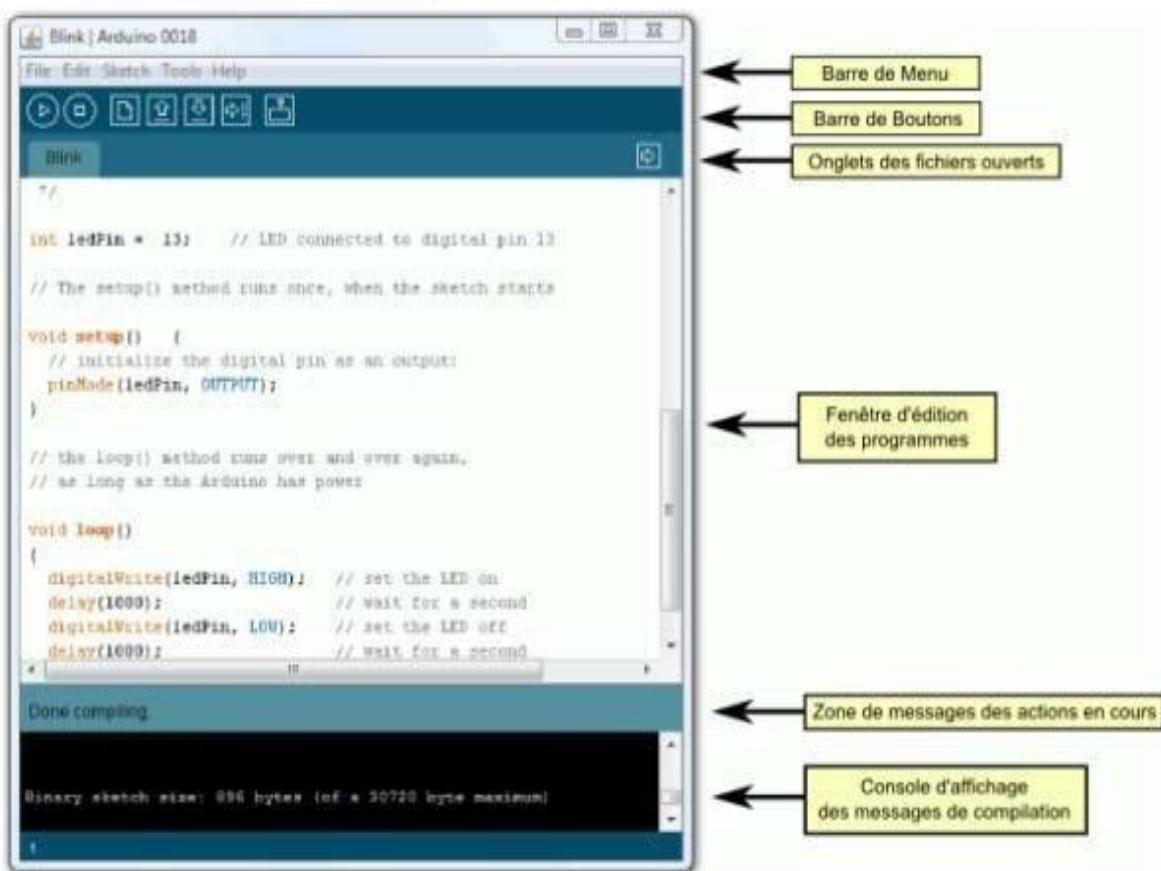


Figure II.24-Espace de développement Intégré (EDI). [27]

### Principe général d'utilisation

Le code écrit avec le logiciel Arduino est appelé un programme (ou une séquence - sketch en anglais)

:

- Ces programmes sont écrits dans l'éditeur de texte. Celui-ci a les fonctionnalités usuelles de copier/coller et de rechercher/remplacer le texte.
- la zone de messages donne l'état de l'opération en cours lors des sauvegardes, des exportations et affiche également les erreurs.
- La console texte affiche les messages produits par le logiciel Arduino incluant des messages d'erreur détaillés et autres informations utiles.
- La barre de boutons permet de vérifier la syntaxe et de transférer les programmes, créer, ouvrir et sauver votre code, et ouvrir le moniteur série.
- La barre des menus permet d'accéder à toutes les fonctionnalités du logiciel Arduino

## Description des menus

Il est courant dans les logiciels de voir des menus contextuels qui s'adaptent au travail en cours. Dans ce cas, cela signifie que les commandes complémentaires disponibles dans les cinq menus sont sensibles au contexte. Cela signifie que seuls les items qui correspondent au travail en cours sont disponibles dans le menu.

- **File** (Fichier) voir figure II.22 le menu propose toutes les fonctionnalités usuelles pour gérer les fichiers de programme Sketchbook. Parmi ces fonctionnalités, il y a apparemment une fonctionnalité qui permet d'avoir un accès direct à tous les programmes dans le répertoire de travail.

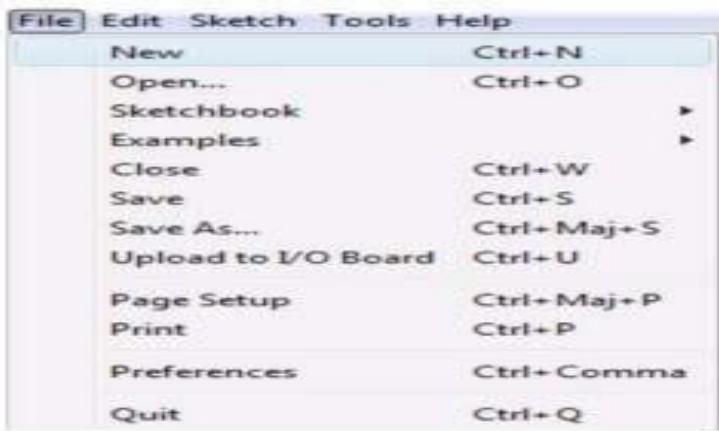
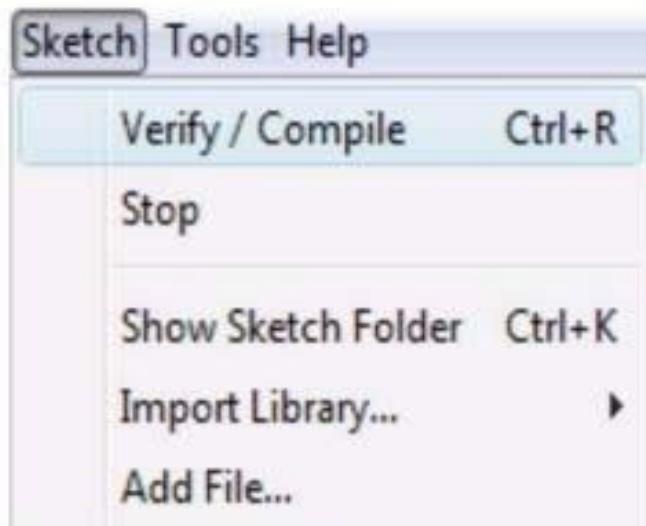


Figure II.25-Menu file.

- **Edit** (Editer),
- **Sketch** (Programme ou Séquence) voir figure II.23, il contient un menu de fonctions
  - Compiler et vérifier le programme (Verify/Compile).
  - Ajouter une librairie au programme en insérant l'instruction # include dans le code. (Import Library).
  - Ouvrir le répertoire courant du programme (Show Sketch Folder).

-Ajouter un fichier source au programme (Add File...). Le nouveau fichier apparaît dans un nouvel onglet dans la fenêtre d'édition. Les fichiers peuvent être retirés du programme en utilisant le menu "tab"



**Figure II.26 –Menu de Sketch.**

□ **Tools** (Outils) voir figure II.24, ce menu permet :

- La mise en forme automatique (Auto Format).
- La sélection de la carte Arduino utilisée (Board).
- Serial Port (Port Série), Ce menu contient tous les ports séries (réels ou virtuels) présents sur l'ordinateur. Il est automatiquement mis à jour à chaque fois que le niveau supérieur du menu outils soit ouvert.

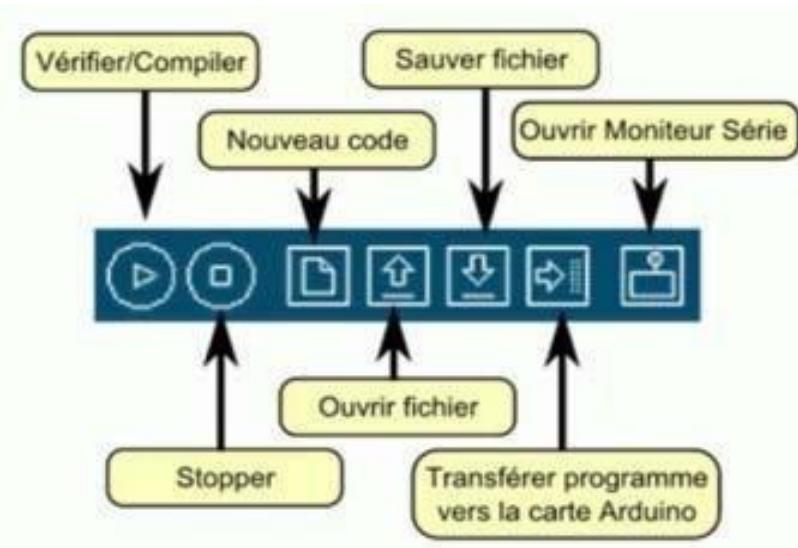


**Figure II. 27-Menu de Tools.**

### Description de la barre des boutons

Elle contient plusieurs options voire figure II.25 :

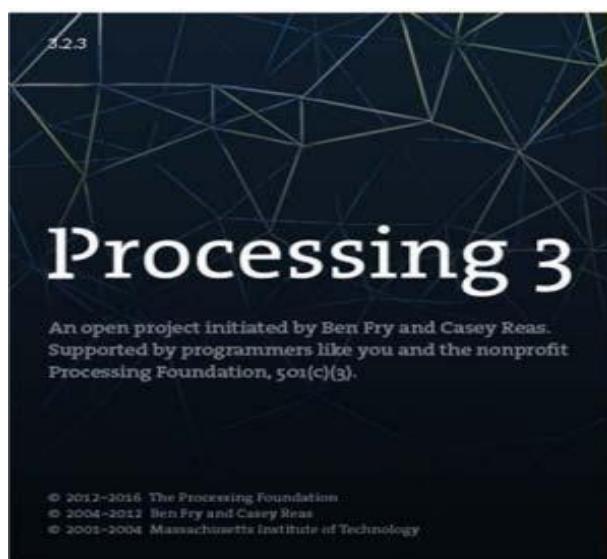
- Vérifier/compiler : vérifie le code en recherchant l'erreur.
- Stop : arrête les boutons activés.
- Nouveau : ouvre une nouvelle fenêtre d'édition vierge.
- Ouvrir : ouvre la liste de tous les programmes déjà sauvegardé au paravent.
- Sauver fichier : permet la sauvegarde du programme.
- Transférer vers la carte : compile et transfert le code vers la carte Arduino.
- Moniteur série : ouvre la fenêtre (TERMINAL SERIE).



**Figure II.28-boutons du logiciel Arduino**

## Processing

Processing est un environnement de programmation open-source basé sur Java, spécialement conçu pour la création d'œuvres graphiques et interactives. Il offre une syntaxe simplifiée et des bibliothèques graphiques prêtes à l'emploi, permettant aux utilisateurs de créer des animations, des jeux, des visualisations et des applications interactives.



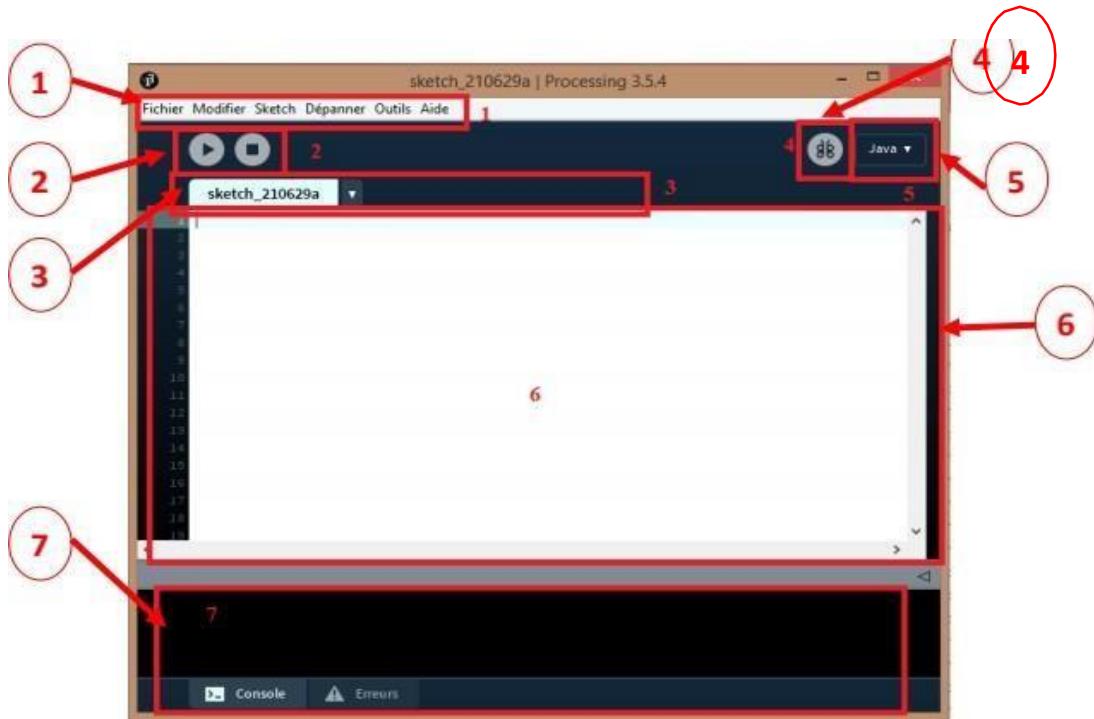
**Figure II. 29-Logo du logiciel processing**

## Interface de Processing

L'interface d'utilisation de processing est constituée de deux fenêtres distinctes dont la première est la fenêtre principale dédiée à la création des projets et la deuxième représente la fenêtre de visualisation destiné à la création de : dessins, animations et vidéos qui vont apparaître dans l'interface.

On trouve plus précisément les éléments suivants dans l'interface :

1. Barre de menu
2. Barre d'actions
3. Barre d'onglets
4. Bouton pour activer le mode debug
5. Liste déroulante pour les modes
6. Zone d'édition (Pour y saisir le programme)
7. Console comprenant un onglet dédié aux messages propre au programme exécuté comme elle comprend aussi un onglet qui affiche les erreurs. Cette console indique aussi si des mises à jour (« Updates ») sont disponibles.



30. L'interface de processing

Figure II.

## **Conclusion**

Dans ce chapitre nous avons pu tirer des concepts importants sur les outils de développements en détaillant les notions de base en matière de fonctionnalité de la carte

Arduino, les différents modules nécessaires pour la mise en place de notre prototype comme on a abordé les différents capteurs utilisé et les différents logiciel utilisé.

Ces notions colletées dans ce chapitre nous seront d'une grande utilité afin de poursuivre le chapitre suivant qui sera dédié à la partie pratique de notre travail.

# **CHAPITRE III**

---

## **Réalisation pratique**

---

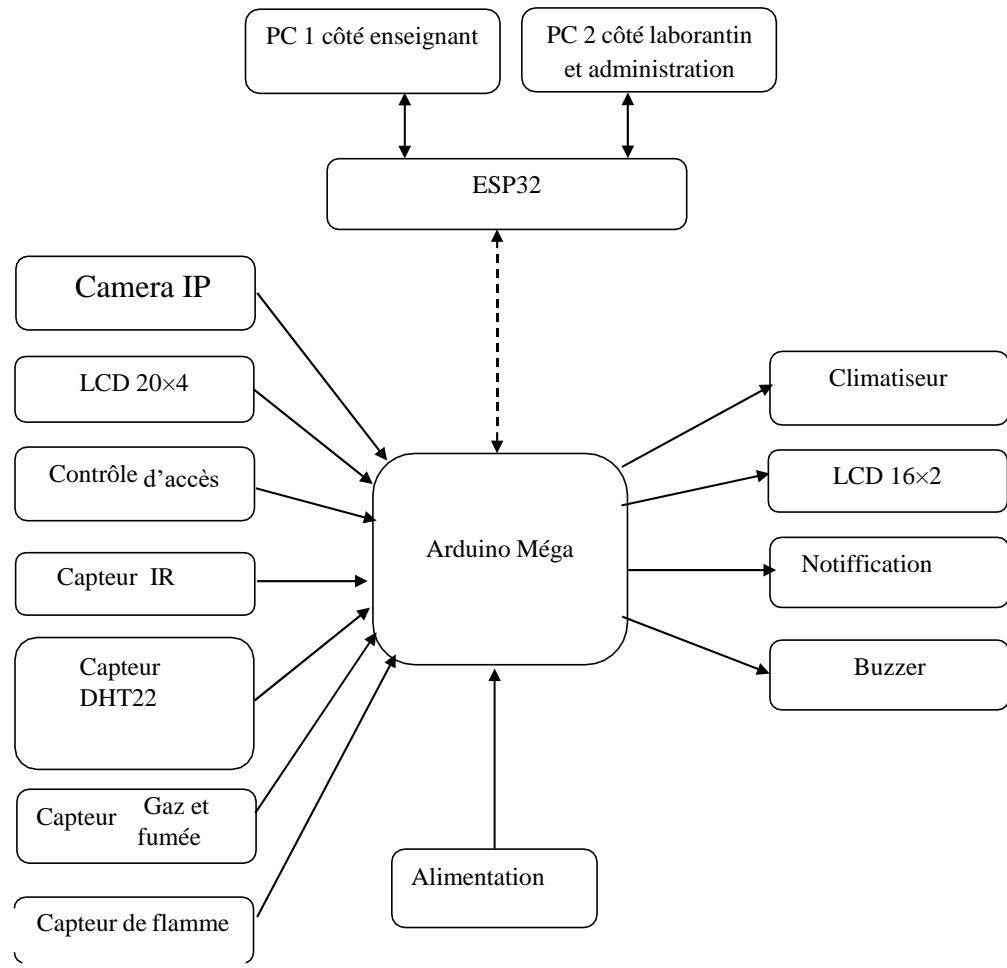
### **III-1. INTRODUCTION**

Les recherches et les informations recueillies pendant la réalisation de notre projet de fin d'études nous ont permis de relier la théorie à la pratique, en nous permettant ainsi de concrétiser la réalisation de notre salle de cours ou de TP intelligente. Pour ce faire ; plusieurs outils de développement matérielles et logicielles sont indispensable en particulier la carte

"ARDUINO". Le présent chapitre est consacré à la mise en œuvre de notre prototype tout en effectuant des tests sur les différents systèmes de la salle de cours ou de TP qui offrent à l'utilisateur le contrôle totale d'une manière fiable et automatique, ainsi que le dialogue à distance entre l'occupant de la salle de TP et les laborantins ou une interface reliant l'enseignant et l'administration pour la salle de cours afin d'établir le suivi en temps réel de cette dernière.

### **III-2. Schéma synoptique générale**

La figure ci-dessous représente un schéma synoptique de notre prototype



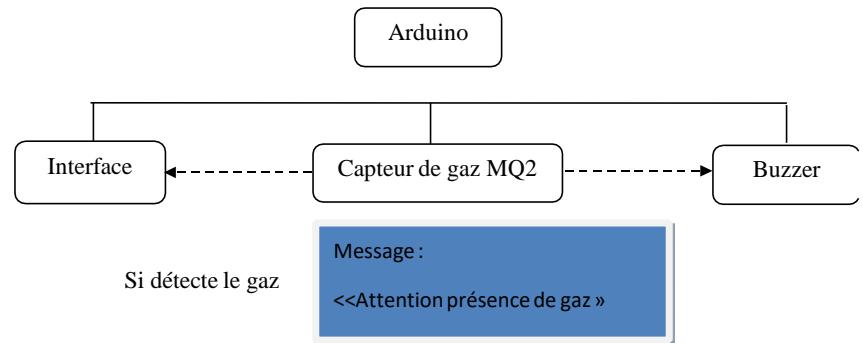
**Figure III.1- Schéma synoptique générale**

## **Etude pratique et fonctionnement de la salle intelligente**

Le fonctionnement de notre salle intelligente se devise en plusieurs sous système pour assurer un fonctionnement optimal du système dans son ensemble.

### **Système de détection de gaz**

Lorsque le système détecte une fuite de gaz, l'alarme se déclenche et un voyant dans l'interface dédié aux personnels de l'administration s'allume en indiquant la présence de gaz. Le schéma suivant résume les différentes actions lors de la détection d'une fuite de gaz (Figure III.2).



**Figure III.2-Schémas de fonctionnement Système de détection de gaz**

### **Simulation sous proteus et réalisation pratique du système de détection de gaz**

L'illustration suivante illustre la simulation sous proteus de système de détection de gaz (Figure III.3).

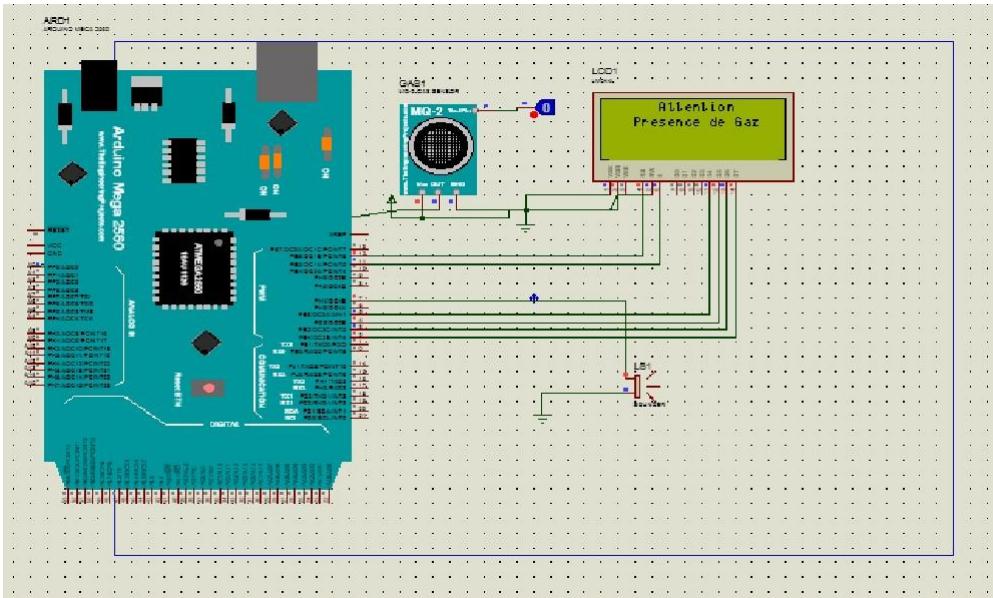


Figure III.3- Simulation sous proteus de système de gaz

Lorsqu' il ya une détection de fuite de gaz (Figure III.4), l'alarme (Buzzer) se déclenche, le LCD affichera l'état de système et un voyant indiquant la présence de fuite de gaz sera allumé dans l'interface graphique implanté sous l'environnement de programmation Processing, témoignent la présence de gaz par le voyant rouge et dans le cas contraire par le voyant vert.



Figure III.4- information de présence de gaz



Figure III.5 –Etat OFF de système de gaz



Figure III.6 –Etat ON de système de gaz

### Système de détection de flamme

Lorsque le système détecte la flamme, l'alarme se déclenche et un voyant dans l'interface dédié aux personnels de l'administration s'allume en indiquant la présence de la flamme. Le schéma suivant résume les différentes actions lors de la détection d'une flamme (Figure III.7).

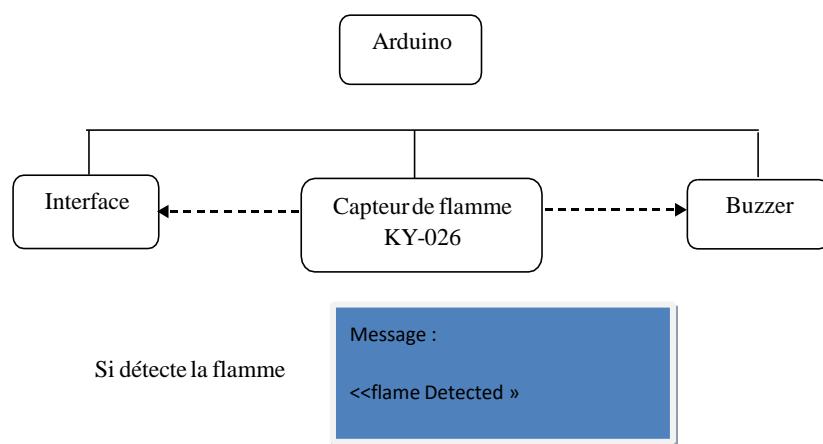


Figure III.7-Schémas de fonctionnement Système de détection de flamme

## Simulation sous proteus et réalisation pratique du système de détection de flamme

L'illustration suivante illustre la simulation sous proteus de système de détection de flamme (Figure III.8).

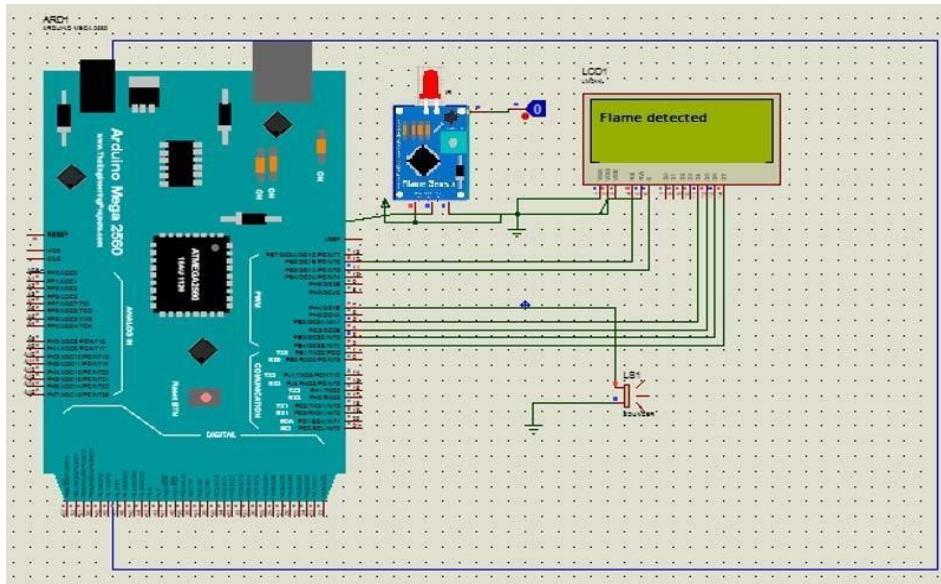


Figure III.8- Simulation sous proteus de système de flamme

Lorsqu'il ya une détection de flamme (Figure III.8), l'alarme (Buzzer) se déclenche, le LCD affichera l'état de système et un voyant indiquant la présence de flamme sera allumé dans l'interface graphique implémenté sous l'environnement de programmation Processing, la (figure III.9) et la (figure III.10) témoignent la présence de flamme par le voyant rouge et dans le cas contraire par le voyant vert

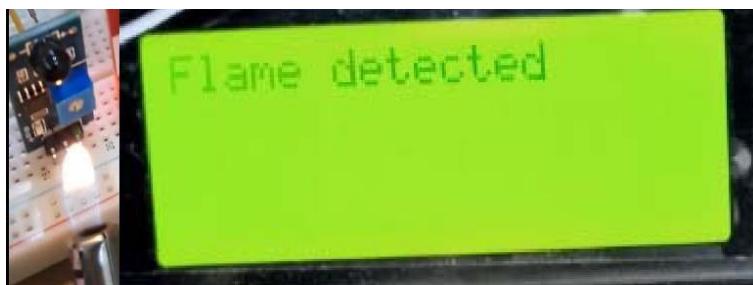


Figure III.9- information de présence de flamme

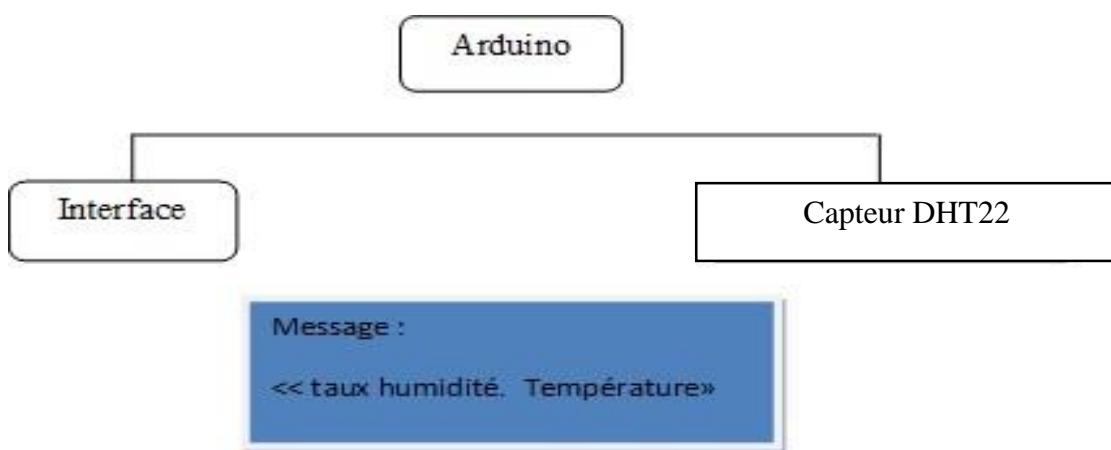


**Figure III.10 –Etat ON de système de gaz et de flamme**

### Système de température et d'humidité

Ce système nous permet de consulter la température et l'humidité dans la salle.

La (Figure III.11) représente le fonctionnement du système de température/humidité en fonction des conditions déclarées.



**Figure III.11-Schémas de fonctionnement système de température/humidité**

**Simulation sous proteus et réalisation pratique de température et d'humidité**

L'illustration suivante représente la simulation sous proteus de système de température et d'humidité (Figure III.12).

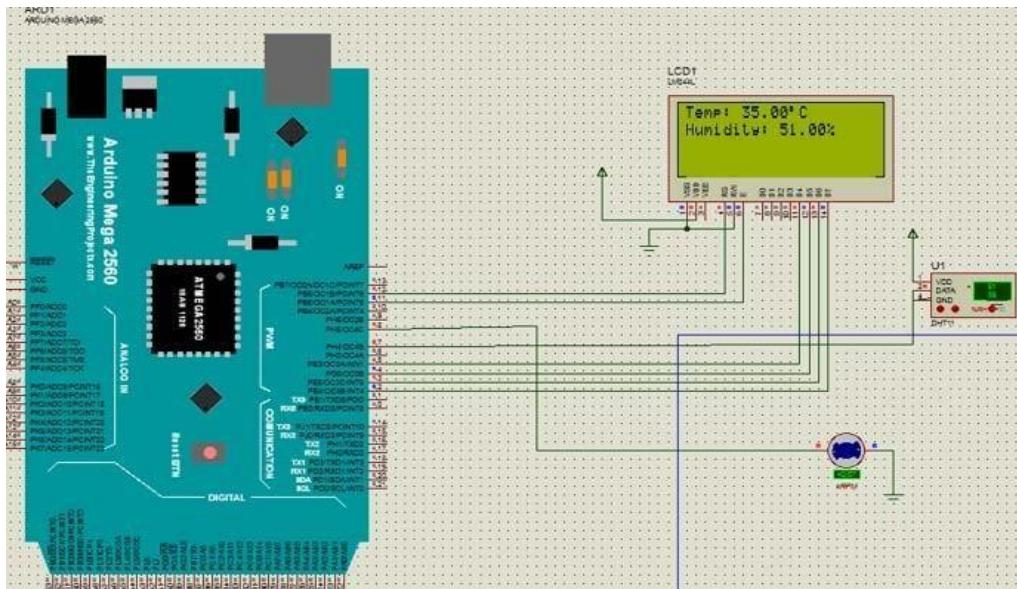


Figure III.12—Simulation sous Proteus de système de température et d'humidité

-La figure ci-dessous montre les paramètres : humidité et température affichés aussi sur l'interface graphique implémentée sous l'environnement de programmation Processing.

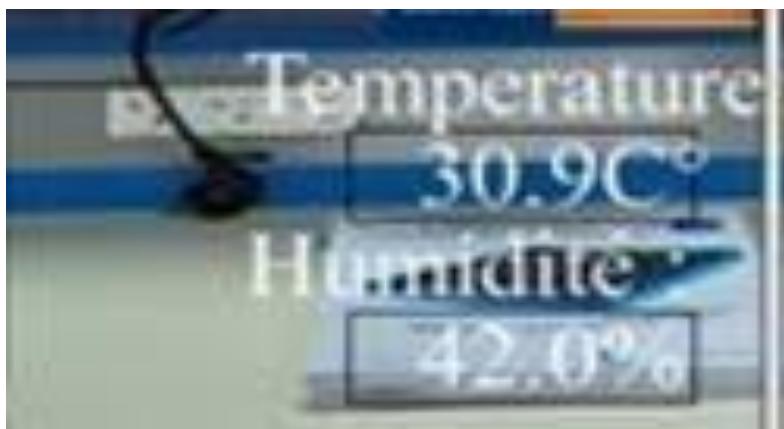
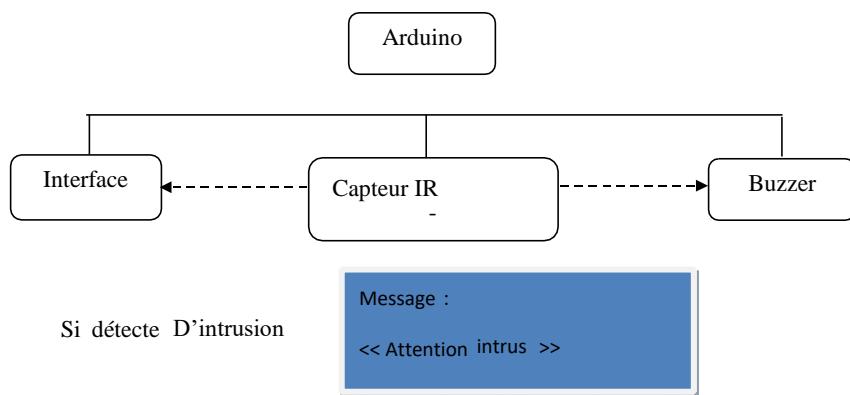


Figure III.13—système d'affichage de température et d'humidité dans l'interface graphique implémenté sous environnement Processing

## Empreinte digitale

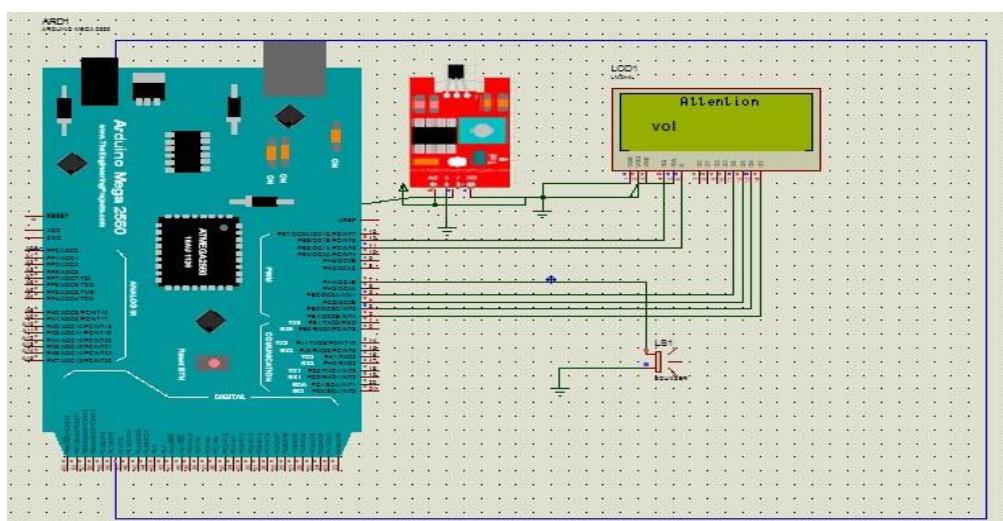
Lorsqu' une personne tente d'introduire dans une salle de cours ou de TP le système signal cette tentation où une alarme se déclenche et un voyant dans l'interface dédié aux personnels de l'administration s'allume en indiquant la tentation de d'intrusion.

Le schéma suivant résume les différentes actions lors de la détection d'une tentation d'intrusion (Figure III.14).



**Figure III.14-Schémas de fonctionnement système antivol  
Simulation sous proteus et réalisation pratique de système d'antivol**

L'illustration suivante représente la simulation sous proteus de système antivol (Figure III.15).



**Figure III.15–Simulation sous proteus de système d'antivol**

Lorsqu'il ya une détection de vol (Figure III.15), l'alarme (Buzzer) se déclenche, le LCD affichera l'état de système et un voyant indiquant la tentation de vol sera allumé dans l'interface graphique implémenté sous l'environnement de programmation Processing, la (figure III.16) et la (figure III.17) témoignent la tentation de vol par le voyant rouge et dans le cas contraire par le voyant vert.



Figure III.16- information de tentation de vol



Figure III.17- système d'affichage de tentation de vol dans l'interface graphique  
implémenté sous environnement Processing

### Système de messagerie

Ce système permet aux enseignants d'envoyer des messages aux laborantins sans se déplacer afin pour demander des équipements ou des composants qui en ont besoin. La (FigureIII.18) montre ce système implémenté sous environnement Processing.



**Figure III.18– système de messagerie implémenté sous environnement Processing**

### Système d'accès

Ce système permet aux enseignants et aux étudiants d'accéder à la salle en insérant le code d'entrée préalablement configuré par l'administration via le clavier ou en passant leur empreinte d'identification. Si le code est correct, un message s'affiche sur l'écran LCD indiquant le nom du propriétaire du code, et la porte s'ouvre automatiquement. En revanche, si l'accès est refusé, un message d'erreur s'affiche sur le même écran LCD. On peut aussi ouvrir la porte directement par le biais de l'interface graphique sans insertion de code la (Figure III.23) montre cette option qui se trouve dans notre interface graphique



Figure III.23– système de contrôle de porte dans l’interface graphique implémenté sous environnement Processing

Le schéma suivant illustre le fonctionnement de ce système (Figure III.24).

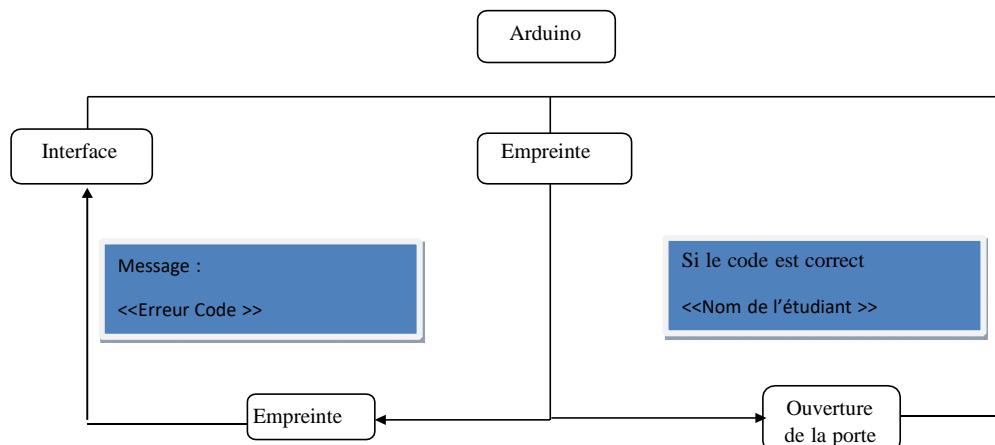


Figure III.24-Schémas de fonctionnement système d'accès

### Système de communication

Le passage de l’information concernant l’état de la salle de cours ou de TP entre notre prototype et l’interface graphique que nous avons implémenté est assuré par une communication WIFI en se servant du module WIFI ESP32. (Figure III.31).

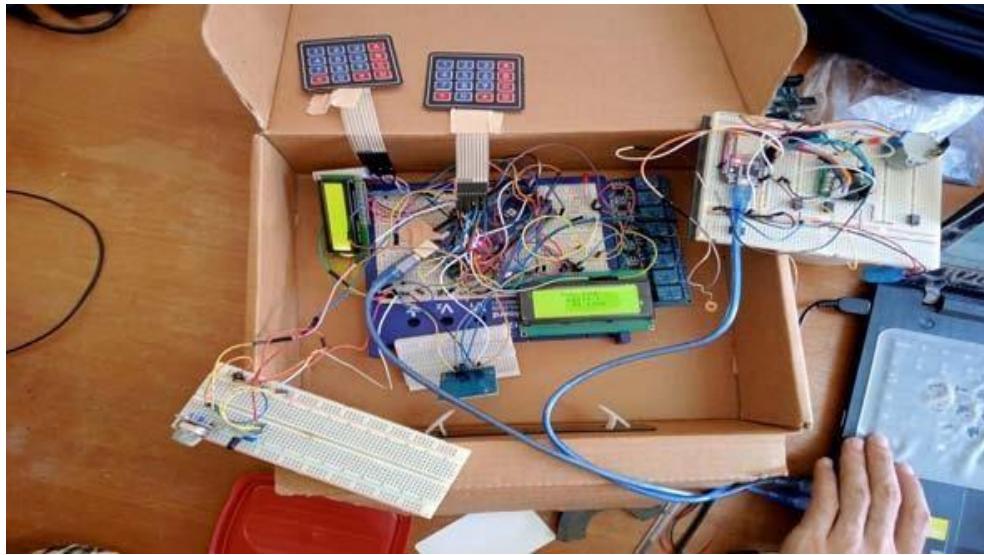


Figure III.31. Exemple de communication non filaire

Le module WIFI ESP32 transmettra un ensemble de données qui comprendra les informations relatives à la salle de cours ou de TP, afin de les afficher dans l'interface sur un PC (Figure III.32).

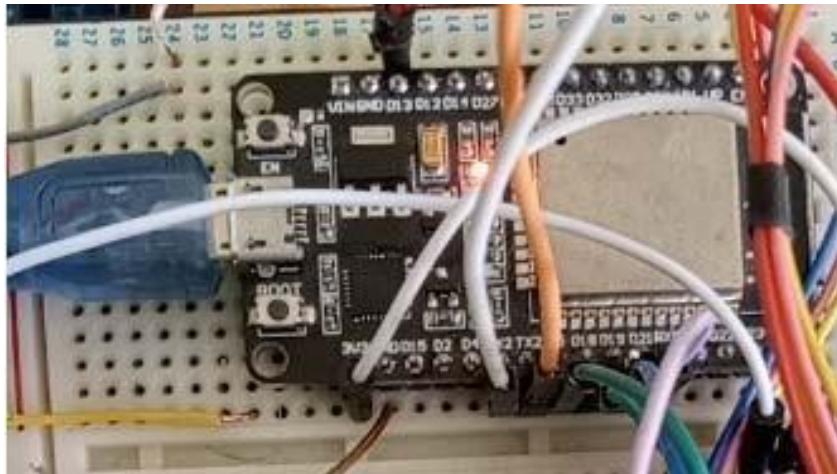


Figure III.32-Module wifi ESP32

### Interface graphique de contrôle de la salle de cours ou de TP

Interface graphique c'est la plateforme (Figure III.33) utilisée pour afficher les diverses données et informations concernant la salle de cours ou de TP sur un PC via une connexion WIFI. L'interface a été développée avec le logiciel Processing, qui utilise le langage de programmation Java. Cette interface joue le rôle de station finale pour recevoir et afficher les différentes données et paramètres

transmis, permettant ainsi de surveiller et de contrôler les différents aspects de la salle de cours ou de TP

## Présentation d'interface

Notre interface portant le nom BK Classroom (Figure III.33) regroupe :

- 1- Secteur de contrôle manuel      3- Secteur d'état (ON/OFF)
- 2- Secteur de monitoring    4- Secteur de messagerie

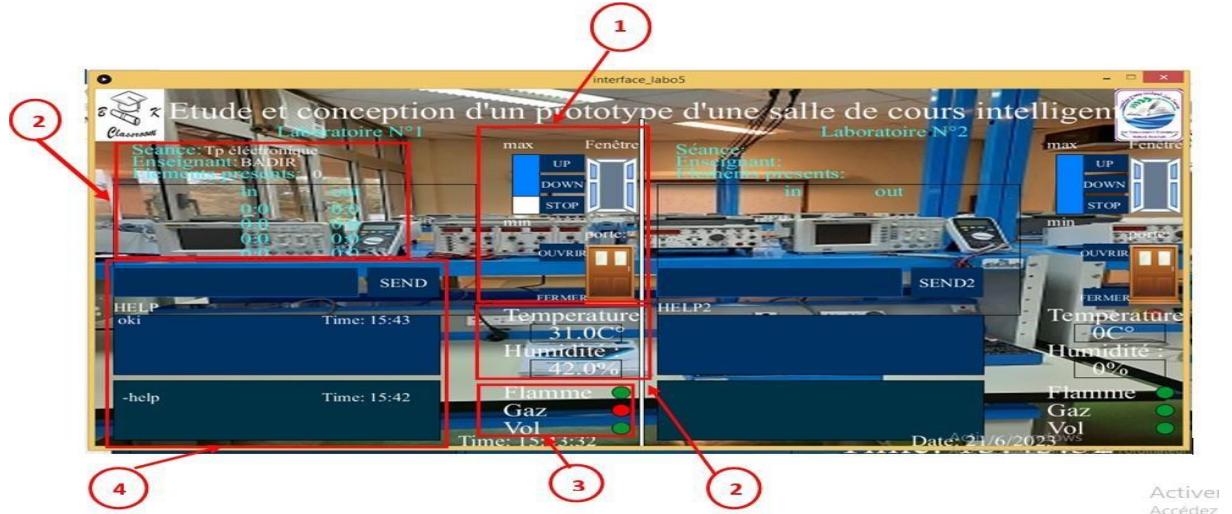


Figure III.33-L'interface graphique

## Schéma global du prototype de la salle de cours ou de TP domotique

La figure ci-dessous représente un schéma global de notre prototype

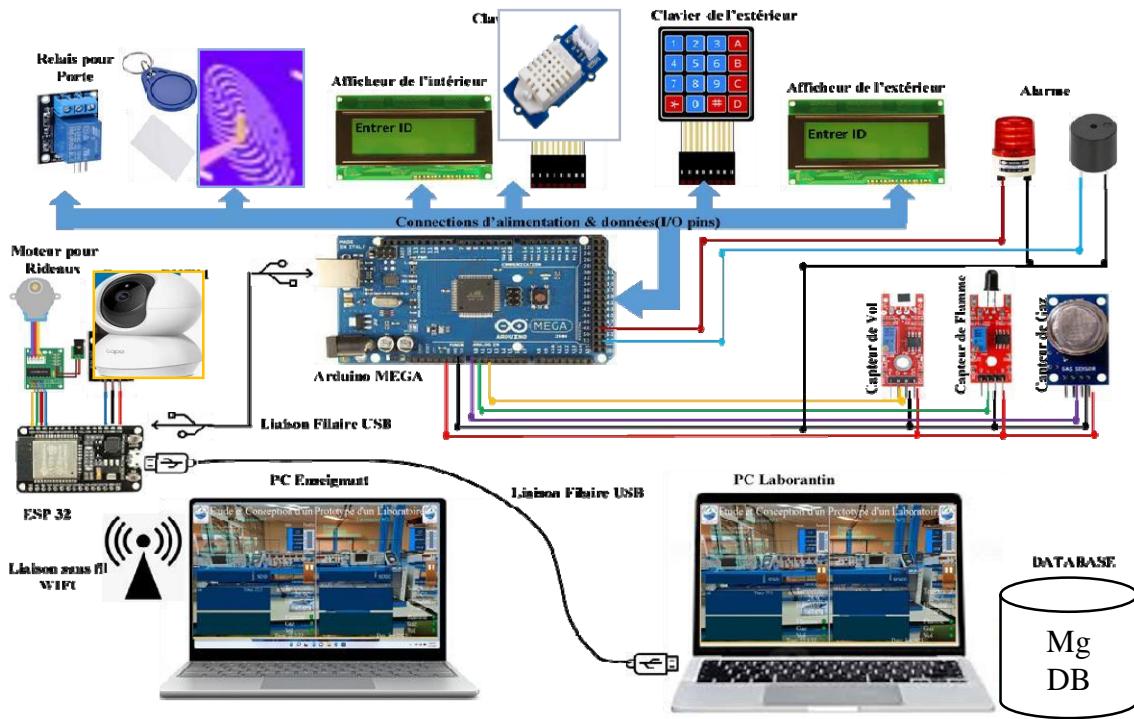


Figure III.34. Schéma global du prototype

## Conclusion

Le chapitre présente une évaluation pratique de notre salle de cours ou de TP domotique, dans le but de confirmer le bon fonctionnement opérationnel de notre prototype. Les aspects abordés tout au long de ce chapitre sont mis en pratique pour concrétiser notre projet

---

## **Conclusion générale**

---

Conclusion générale

Notre projet de fin d'étude est dédié à la conception et la réalisation d'un plateau technique hardware et software aboutissant à un prototype de salle de cours ou de TP domotique conçu pour assurer :

- La conservation et la protection des équipements pédagogiques dans les salles de cours et de TP.
- La sécurité de salle de cours et de TP.
- Le contrôle d'accès des étudiants ainsi que les enseignants au sein de la salle de cours ou de TP.
- Le contrôle de présence et absence des étudiants ainsi que les enseignants.
- Le contrôle de chevauchement des occupations de salles de cours et de TP.
- La coordination efficace entre laborantins et enseignants de TP via une interface graphique conçue sous environnement de programmation Processing.

Nous sommes ravis et fière de dire que ce projet nous a permis d'approfondir nos connaissance en matière d'électronique tout en manipulant avec fluidité les matériel et les outils de développement de notre prototype. Bien évidemment, l'ensemble de ce travail s'est déroulé dans des conditions favorables, où un effort collectif remarquable et une entente harmonieuse ont conduit à la réalisation d'un travail concluant et gratifiant.

Ce modeste projet nous a offert l'opportunité d'explorer de nouveaux horizons dans le domaine des informations techniques. En approfondissant nos recherches, nous avons pu repousser nos limites et progresser dans un domaine vaste, développé et sophistiqué.

Un aspect à prendre en compte serait la disponibilité limitée de certains composants électroniques, ainsi que le manque de temps nécessaire pour approfondir davantage ce projet. En effet ce domaine est vaste et en constante évolution, et il reste encore de nombreuses possibilités d'intégration et d'amélioration à explorer.

En effet, les amateurs de domotique ont de nombreuses possibilités à explorer. Rien ne nous empêche de poursuivre notre chemin et de concrétiser nos idées.

En effet, il existe de nombreuses possibilités pour les passionnés de la domotique en particulier et de l'électronique en général. Malgré les limitations, nous avons fait de notre mieux pour atteindre notre objectif.

Effectivement, il existe de nombreuses opportunités pour les passionnés de la domotique en particulier et de l'électronique en général. En dépit des ressources limitées dont nous disposions, nous avons fait de notre mieux pour atteindre notre objectif.

Ce projet était pour nous une expérience riche et très intéressante

## Références bibliographique

### Bibliographie

- [1] Astalaseven ,Eskimon et olyte , (Arduino pour bien commencer en électronique et en Programmation).
- [2] LEHSAINI Ilyes et BENDIMERAD Abderrahman,(Etude et réalisation d'une plateforme d'acquisition micro contrôlée et de transmission Bluetooth du signal ECG sur Smartphone), mémoire de projet de fin d'études, Tlemcen 2015.

[3] SIDI ALI CHERIF ABD EL GHANI ; BENTRARI OUM EL KHEIR HAYET (Etude et conception d'un système dédié à la mesure de l'activité électrique myocardiqueECG).

[4] SIDI ALI CHERIF ABD EL GHANI ; BENTRARI OUM EL KHEIR HAYET (Etude et conception d'un système dédié à la mesure de l'activité électrique myocardiqueECG).

[5] Tutoriel pdf Arduino DHT11 [Eng]

[6] StambouliEchaima et BerbaraRokia (contrôle de maison a distance), mémoire de master spécialité : signaux en ingénierie des systèmes et informatique industrielle (SISII) édition

2016-2017

[7] Simon Landrault (Eskimon) et Hippolyte Weisslinger (olyte), (Arduino : Premiers pas en informatique embarquée), Édition du 19 juin 2014.

## Webographie

[W1] <https://www.lemagdeladomotique.com/dossier-1-domotique-definition-applications.html>

[W2] <https://www.journaldelagence.com/wpcontent/uploads/2018/09/la-domotique.pdf>

[W3] <https://www.quelleenergie.fr/economies-energie/domotique/>

[W4] <https://www.aranacorp.com/fr/gerez-un-ecran-lcd-16x2-avec-arduino/>

[W5] <http://electromaroc.com/12-module-arduino-pic-fpga/84-clavier-matriciel.html>

[W6] <http://wiki.jelectronique.com/doku.php?id=esp32>

[W7] <https://pecquery.wixsite.com/arduino-passion/le-buzzer>

[W8] <https://www.circuitbasics.com/what-is-a-relay/>

[W9] <https://www.lextronic.fr/module-relais-5v-40436.html>

[W10] <https://www.lextronic.fr/module-relais-5v-40436.html>

[W11] <https://www.moussasoft.com/product/moteur-pas-a-pas-avec-driver-uln2003>

[W12] <https://volta.ma/produit/module-driver-uln2003-moteur-pas-a-pas/>

[17]<https://www.futura-sciences.com/tech/definitions/tech-rfid-4187/>

[W13] <http://wiki.jelectronique.com/doku.php?id=esp32>

[W14] <https://www.indiamart.com/proddetail/mq2-gas-sensor-module-21147694873.html>

- Page

## Références bibliographique

[W15]<https://sensorkit.oy-it.net/fr/sensors/ky-026>

[W16]<https://arduino-france.site/capteur-flamme/>

[W17]<https://fr.rs-online.com/web/p/capteurs-a-effet-hall/2166231>

[W18]<https://manuals.plus/fr/honeywell/linear-hall-effect-sensor-ics-manual#axzz85jZmMCM3>

[W19] <http://automacile.fr/definition-arduino-quest-ce-quun-arduino/>

---

## **Annexes**

---

Ces dernières décennies le monde a connu un développement indéniable dans le secteur de technologie en générale et dans le bien être du personnel en particulier, où l'apparition de la domotique a rendu la vie quotidienne beaucoup plus simple notamment dans les habitats ainsi que dans nos lieux de travail y compris les milieux universitaires. Nous nous sommes focalisés sur la conception d'une salle de cours ou de TP domotique afin de répondre à des problèmes rencontrés au sein de l'université où nous citons :

#### Sécurité de salle de cours et TP.

- Perte d'équipements pédagogiques.
  - Mauvaise coordination entre laborantins et enseignants de TP.
  - Manque de contrôle de présence des étudiants et enseignants et chevauchement des occupations de salles de cours.
- 
- Permet de visualiser et de contrôler les paramètres fait par nos spécialistes dans la salle de cours ou de TP en temps réel
  - Utilisation des capteurs répandant à la sécurité intérieure de la salle de cours ou de TP
  - Installation de capteur dédié aux antivols
  - Fournir un service de sécurité pour la protection des équipements pédagogiques
  - Respect des emplois du temps en matière d'occupation des salles ainsi que la présence des étudiants et enseignant via un capteur et une conception d'interface graphique pour le contrôle et la commande.
  - Développement d'une interface de messagerie entre enseignant et laborantin

- Améliorer et augmenté la qualité de l'enseignement
- Fabrication et installation selon les besoins du client
  
- Augmentation de l'efficacité et gain de temps
- Automatisation totale du système sans l'intervention humaine
- Maintenance préventive par notre équipe spécialiste
- Ajout/suppression d'un service selon les besoins des clients
- Facilité le travail de l'administration
- Mode manuel / automatique
- Adaptation automatique des paramètres de notre prototype

Les projets ciblant le même problème sont :

- Au niveau national : il n'y a aucun projet comme notre prototype.
- Au niveau international :

Il existe des salles de cours universitaires domotiques dans différents pays à travers le monde. Les universités et les établissements d'enseignement supérieur reconnaissent l'importance de la technologie et de la domotique dans l'éducation et l'apprentissage. Par conséquent, de nombreux établissements universitaires ont mis en place des salles de cours équipées de technologies domotiques pour offrir un environnement d'apprentissage plus interactif, moderne et adapté aux besoins des étudiants.

Certains pays, tels que les États-Unis, le Japon, l'Allemagne, le Royaume-Uni et la Corée du Sud, sont généralement à la pointe de l'adoption de ces technologies dans les universités et dans l'Afrique du sud.

Voici les liens vers les sites web officiels des universités mentionnées précédemment :

1. Massachusetts Institute of Technology (MIT) - États-Unis:  
[<https://www.mit.edu/>](https://www.mit.edu/)
2. Stanford University - États-Unis: [<https://www.stanford.edu/>](https://www.stanford.edu/)
3. University of California, Berkeley - États-Unis:  
[<https://www.berkeley.edu/>](https://www.berkeley.edu/)
4. Tokyo Institute of Technology - Japon:  
[<https://www.titech.ac.jp/>](https://www.titech.ac.jp/)
5. Seoul National University - Corée du Sud: [<https://www.snu.ac.kr/>](https://www.snu.ac.kr/)
6. Technical University of Munich - Allemagne: [<https://www.tum.de/>](https://www.tum.de/)
7. University of Cambridge - Royaume-Uni:  
[<https://www.cam.ac.uk/>](https://www.cam.ac.uk/)

University of Oxford - Royaume-Uni: [<https://www.ox.ac.uk/>](<https://www.ox.ac.uk/>)

8. University of Cape Town - Afrique du Sud:  
[<https://www.uct.ac.za/>](<https://www.uct.ac.za/>)

Les types de domotique offerts par les universités peuvent varier en fonction des ressources, des priorités et des domaines d'expertise spécifiques de chaque institution. Voici quelques exemples des types de domotique que certaines universités pourraient proposer dans leurs salles de cours :

1. Contrôle de l'éclairage intelligent : Les salles de cours peuvent être équipées de systèmes d'éclairage intelligents qui permettent de régler l'intensité lumineuse, la couleur et les scénarios d'éclairage en fonction des besoins. Cela peut favoriser un environnement d'apprentissage confortable et productif.

2. Gestion de l'énergie : Les systèmes de gestion de l'énergie peuvent être utilisés pour surveiller et contrôler la consommation d'énergie dans les salles de cours, en optimisant l'utilisation des ressources et en réduisant les coûts énergétiques.

3. Contrôle de la climatisation : Les salles de cours peuvent être équipées de systèmes de climatisation intelligents qui permettent de réguler la température et le confort thermique en fonction des besoins, tout en optimisant l'efficacité énergétique.

4. Intégration des technologies audiovisuelles : Les universités peuvent utiliser la domotique pour intégrer des technologies audiovisuelles avancées dans leurs salles de cours, telles que des écrans interactifs, des vidéoprojecteurs, des systèmes de sonorisation et des équipements de vidéoconférence.

5. Automatisation des équipements : Les équipements de la salle de cours, tels que les stores, les rideaux, les écrans et les tableaux, peuvent être automatisés pour s'ajuster en fonction des besoins des enseignants et des étudiants.

- Destiné aux :

- les universités

- les lycées et secondaire

- voire même dans toute entreprise ou société étatique ou privé désirant domotisé leur établissement

○ Fournir un service de livraison gratuit

○ Guide d'installation offert

- Formation gratuite
- Garantie d'usage
- Service d'installation et de configuration selon le type d'offres
- Maintenance préventive
- Assistance téléphonique et en ligne pour répondre aux questions des clients
- Assistance téléphonique et en ligne pour répondre aux questions des clients

Les matériels pour la faisabilité du projet :

| Nom matériel                         | Prix unitaire |
|--------------------------------------|---------------|
| Kit complet arduino uno              | 520 000 FG    |
| Empreinte digitale (optique)         | 250 000 FG    |
| ESP32                                | 180 000 FG    |
| Caméra IP (par capteur de mouvement) | 50 000 FG     |
| Lecteur luminosité                   |               |
| Capteur de fumée et de Gaz           | 80 000 FG     |
| Total                                | 1080 000 FG   |

