

**Realisé par :**

**Mohamed Mouad Messaoudi**

**Projet algo-microprogrammation : première année**

**Encadré par : Dr. OMAR EL KADMIRI**

**SMARTHOME**

**2020/2021**

## Table of contents

Remerciement.....	3
Introduction .....	4
CHAPITRE 1 .....	5
Cahier de charge.....	5
Diagramme de Gantt .....	7
Etude bibliographique .....	8
CHAPITRE 2 .....	19
Simulation sur Isis Proteus .....	19
I) Contrôle automatique de l'éclairage et la ventilation .....	19
II) Contrôle d'accès.....	26
Montage su fritzing .....	35
a) Contrôle automatique de l'éclairage.....	35
b) Ventilation .....	36
c) Contrôle d'accès .....	37
CHAPITRE 3 .....	38
Caractéristique du matériel utilisée .....	38
Capteur dht11 .....	38
Capteur PIR sensor .....	38
Photorésistance (LDR): .....	39
Photo de la réalisation de la maison intelligente.....	42
Vue générale de la maison.....	42
La ventilation et l'éclairage.....	43
La ventilation.....	44
L'éclairage .....	45
Contrôle d'accès .....	46
Conclusion.....	47

## **Remerciement**

Nous tenons à exprimer nos plus vifs remerciements à Mr Omar El Kadmiri qui fut pour nous un enseignant et un bon encadreur attentif et disponible malgré ses nombreuses charges. Sa compétence, sa rigueur scientifique et sa clairvoyance nous ont beaucoup appris. Ils ont été et resteront des moteurs de notre travail ainsi qu'un bon modèle à suivre.

Nous souhaitons aussi remercier nos parents, ainsi que nos sœurs, et frères, pour l'intérêt qu'ils ont toujours porté à ce que nous réalisons. Merci d'avoir toujours été là.

Nous exprimons tous notre respects et remerciements à l'ensemble des membres du jury.

Nous adressons toute notre gratitude à tous nos ami(e)s et à toutes les personnes qui nous ont aidé dans la réalisation de ce travail.

## **Introduction**

La domotique regroupe les technologies de l'électronique, de l'automatique, de l'informatique et des télécommunications permettant d'améliorer le confort, la sécurité et la gestion d'énergie d'une maison.

En effet, la domotique permet par exemple d'optimiser l'utilisation de l'éclairage, du chauffage afin de réduire notre consommation en énergie. C'est pourquoi notre tuteur nous a proposé de fabriquer un modèle réduit d'une maison dite « intelligente » ou « smart home ».

Cette maquette, à l'échelle d'un Playmobil, permettrait de présenter certaines fonctionnalités de la domotique à travers 3 scénarios : contrôle automatique d'éclairage, ventilation et d'autre fonctionnalité qu'on va choisir par la suite. Ces scénarios seront automatisés via des cartes « Arduino » exécutant des programmes informatiques.

# **CHAPITRE 1**

## **Cahier de charge**

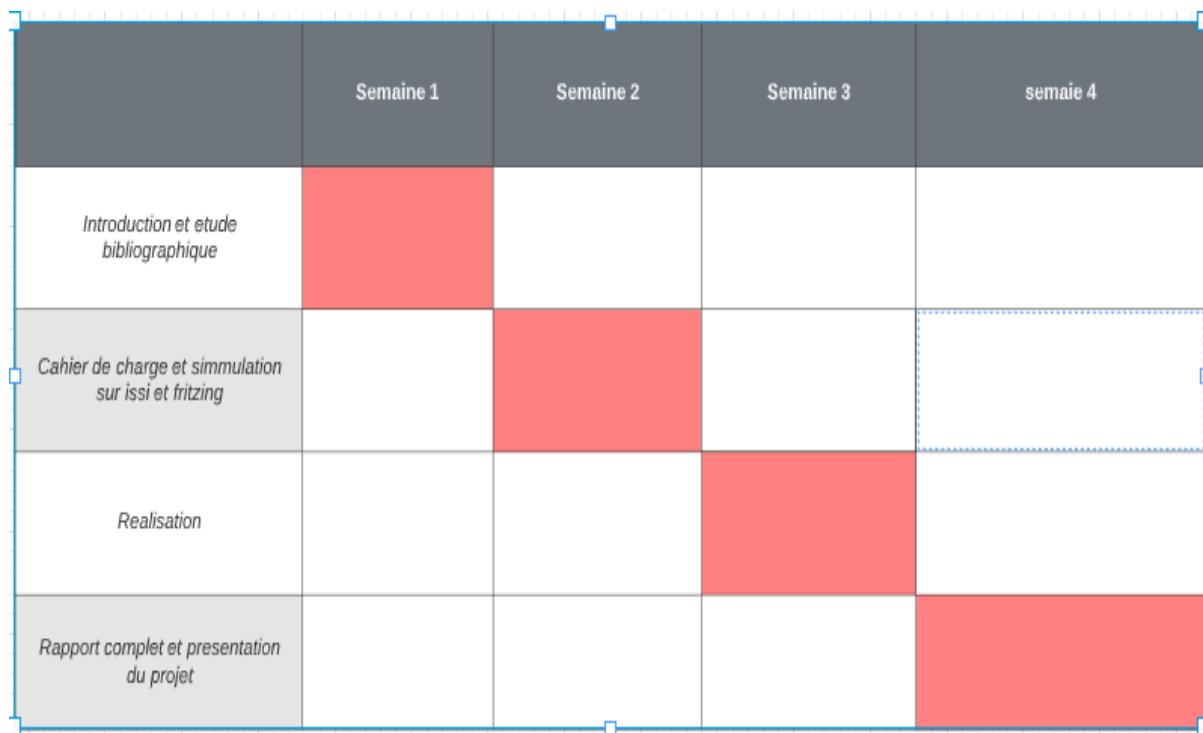
En début , "Smart Home" est le terme communément utilisé pour définir une résidence qui a électroménagers, éclairage, chauffage, climatisation, téléviseurs, ordinateurs, divertissement audio et les systèmes vidéo, de sécurité et des systèmes de caméras qui sont capables de communiquer les uns avec les autres et peut être contrôlée à distance par un calendrier, à partir de n'importe quelle pièce de la maison, ainsi qu'à distance à partir de n'importe quel endroit dans le monde par téléphone ou par Internet. Mais pratiquement notre projet La "Smart Home" (maison intelligente) est une maison qui permet à ses propriétaires le confort de la maison, la sécurité, l'efficacité énergétique (faibles coûts d'exploitation) et commodité à tout moment, indépendamment de savoir si quelqu'un est à la maison.

En effet, La maison intelligente se distingue par les rôles pour lesquels elle est conçue. On peut dire que Notre objectif de ce projet est ses fonctionnalités, dont les plus importantes sont :

1-Améliorer le confort, la sécurité, la communication et la gestion d'énergie.

2-optimiser l'utilisation de l'éclairage, du chauffage afin de réduire notre consommation en énergie.

## Diagramme de Gantt



## **Etude bibliographique**

Garin Lucas, Decelle Lucas, Delise Antoine, Roux Nicolas et Madelon Rémi ont déjà travaillé sur le même projet. Mais ils ont pensé à faire les fonctionnalités suivantes : Motion sensor, Free cooling et Fire Alarm .

### **A) la conception de la maison :**

#### **Struture principale**

La première étape est la plus importante, tout ce qui suivra dépendra de celle-ci. Elle consiste à créer une maison. Pour cela, nous avons dessiné la structure principale, c'est-à-dire les murs extérieurs et le sol. Nous avons alors utilisé le logiciel Makercase dont le lien est ici : <http://www.makercase.com/> . Ce logiciel permet de construire le plan d'un cube ou d'un rectangle dans le but d'être découpé dans un matériau adéquat. Ce logiciel propose 3 façons d'emboîter les murs : Finger, Flat et T-Slot. On utilise le type de fixation « Finger », et de ce fait le cube crée pourra tenir sans colle ni clou. Le



but est de construire une maison adaptée à la taille d'un Playmobil. De ce fait nous avons choisi une dimension de 40cm\*40cm avec des murs fait en MBF d'une épaisseur de 3 mm. Pour une meilleur visualisation et compréhension des différents systèmes présents dans la maison, nous avons effectué une coupe de la maison en ajoutant des traits rouges supplémentaires de biais sur deux murs opposés. Ainsi la hauteur maximum des murs est de 16cm et la hauteur minimum est de 3,5cm.

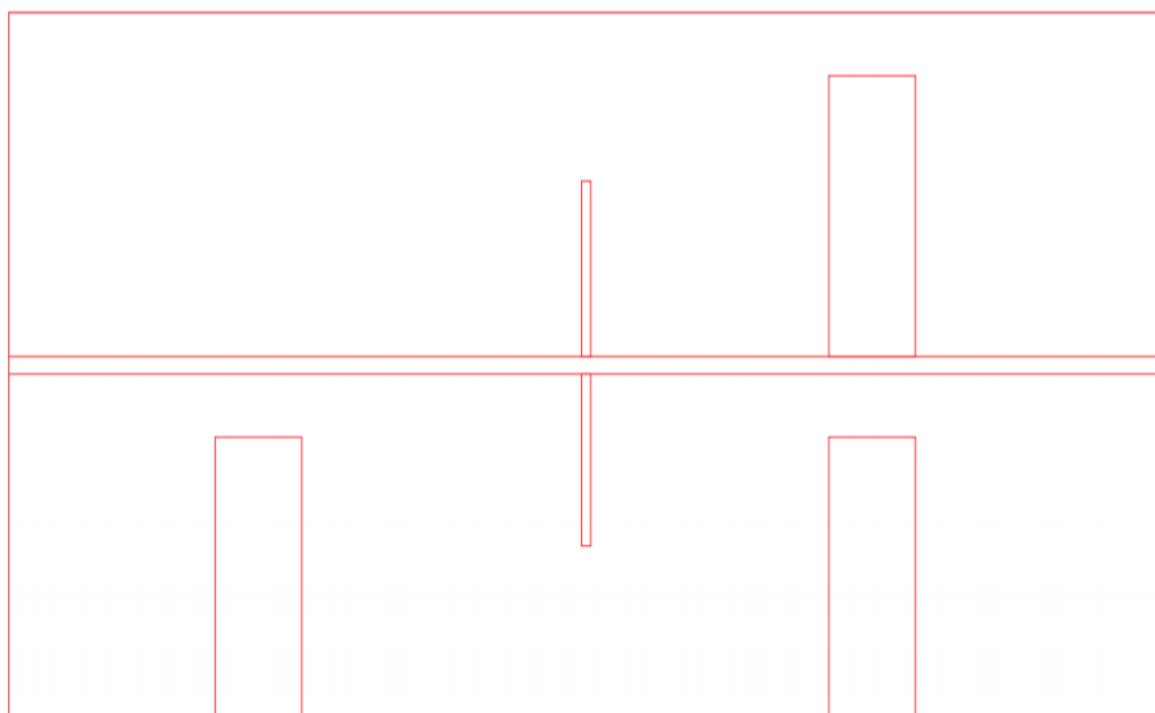
Ensuite, nous avons dessiné une porte d'entrée d'une dimension 8 cm par 3 cm sur un coté perpendiculaire à la coupe. De plus, nous avons incluse deux fenêtres 5cm par 5cm situées sur le mur du fond, celui qui est le plus grand en hauteur et en largeur après la découpe. Elles sont aussi bien nécessaires pour nos scénarios automatisés que d'un point de vue esthétique. Nous avons ensuite enregistré les plans en format .svg pour créer la structure de notre maison.

#### **a) Murs intérieurs**

Le but de cette étape est de définir la disposition des pièces à l'intérieur de la maison. Ainsi il faut dessiner les

plans des murs intérieurs. Il y a 2 logiciels principaux pour concevoir les plans : Inkscape et Adobe Illustrator. Le premier est gratuit mais n'est pas très pratique pour dessiner des plans précis. Le deuxième est payant mais offre de nombreuses fonctionnalités, et est plus adapté à notre besoin. Par chance, Adobe Illustrator met à disposition une version d'essai pendant 30 jours. De ce fait nous avons établi la configuration intérieure de notre maison sur ce logiciel. Nous avons décidé de séparer la maison en quatre pièces de même taille, soit 20cm\*20cm. Ceci nous a permis de mettre en place, et par la suite de présenter, nos quatres scénarios facilement, de manière précise et distincte. Pour cela, nous avons dessiné deux murs perpendiculaires sur Adobe Illustrator. Comme notre maison est censée être utilisée et reconstruite à nouveau, notamment par des collégiens ou des lycéens dans le but de découvrir le potentiel des cartes arduino, nous avons cherché une façon pratique d'ajouter les murs intérieurs dans notre maison. C'est pourquoi, nous avons équipé nos murs de petites fentes, dessinés via Adobe Illustrator, afin de

pouvoir les emboîter en forme de croix. Ceci permet de créer une maison en kit-détachable, on peut enlever les murs intérieurs et détacher les murs extérieurs grâce aux encoches « Finger » afin de mettre le tout dans un sac. Nous avons également incluse 3 portes (8cm\*3cm) sur les murs intérieurs pour permettre aux « Playmobils » de circuler de pièces en pièces et d'actionner chacun des scénarios individuellement.



## **b) Mobiliers**

Pour rendre plus réelle notre maison, nous avons décidé d'inclure plusieurs objets du quotidien, ainsi que du mobilier, propre à une maison. Cette dernière étape de

fabrication est majoritairement esthétique mais s'avère utile suivant les scénarios puisque certains d'entre eux nécessitent un réveil, un lit, etc.. pour que les scénarios s'enclenchent. Nous avons alors récupéré les plans essentiels sur le site

<http://fablab.ensimag.fr/index.php/PILBI-2013-Team2>.

On distingue alors deux types de mobilier, celui en bois, et celui en plastique. Les plans des objets, tel que la table, le bureau etc ... ont été édités sur Adobe

Illustrator et peuvent être fabriqués par la découpeuse laser. Les autres plans, comme les WC, le réveil, etc ...

ont été réalisés grâce à l'imprimante 3D. Nous avons ajusté la taille des objets grâce aux réglages sur

l'imprimante 3D et nous avons lancé leur production. Si quelqu'un veut ajouter des accessoires à notre maison, on trouve énormément de différents objets sur le site

<https://www.thingiverse.com> . L'imprimante 3D permet de matérialiser les plans disponibles sur « Thingiverse », il suffit de modifier la taille avant la production pour rester à l'échelle d'un playmobil. Nous avons ensuite récupéré le plan de notre maison édité sur Makerbase et

nous l'avons ouvert sur Adobe Illustrator afin d'ajouter n'importe quelle touche personnelle, texte ou photo. Effectivement, la découpeuse laser a aussi la possibilité de graver.



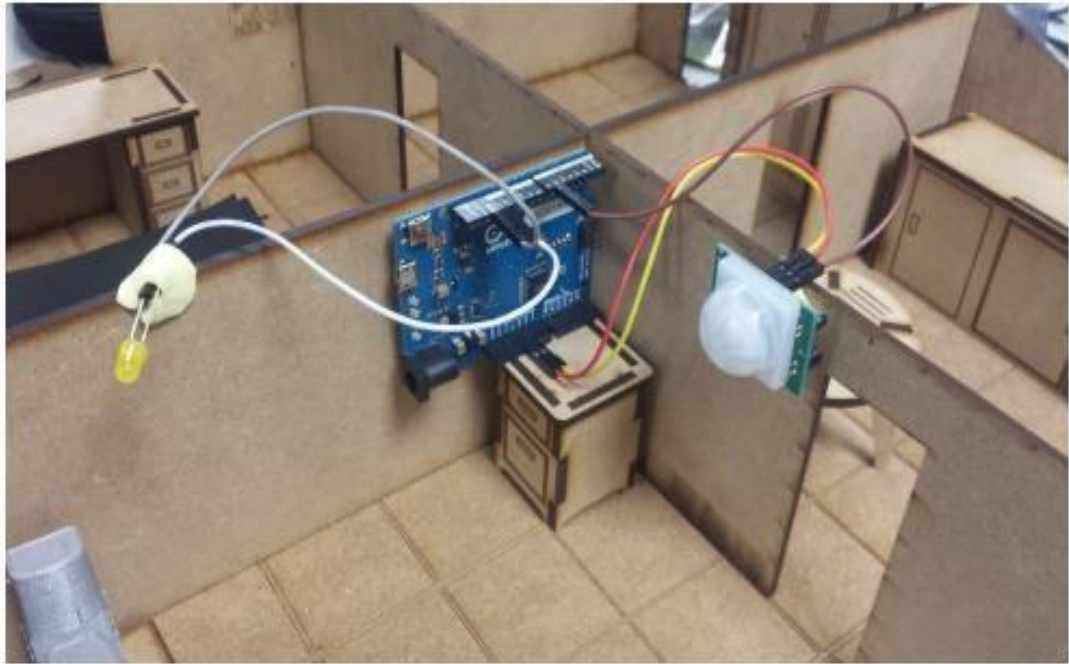
## **B) L'automatisation de la maison**

### **a) Motion Sensor**

La gestion de la consommation d'énergie est une tendance actuelle, c'est pourquoi nous avons modélisé un premier scénario qui respecte cette tendance à une échelle réduite. Notre dispositif permet d'allumer automatiquement la lumière en cas de présence dans une pièce mais permet aussi d'éteindre celle-ci en l'absence de mouvement pendant 10 min.

Le matériel que nous avons utilisé est le suivant :

- Un capteur de mouvement
- Une LED (lampe)
- Une résistance 220 ohm
- Une Carte Arduino

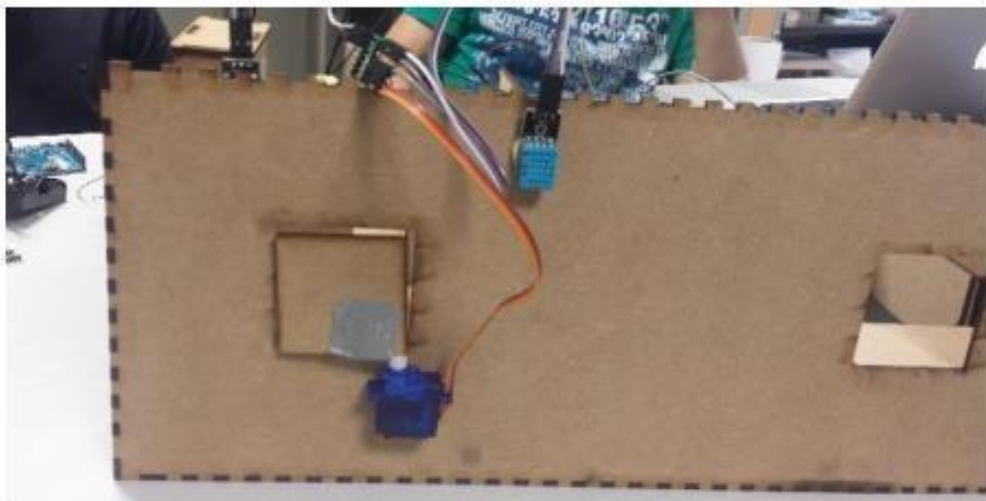


### **b) Free Cooling**

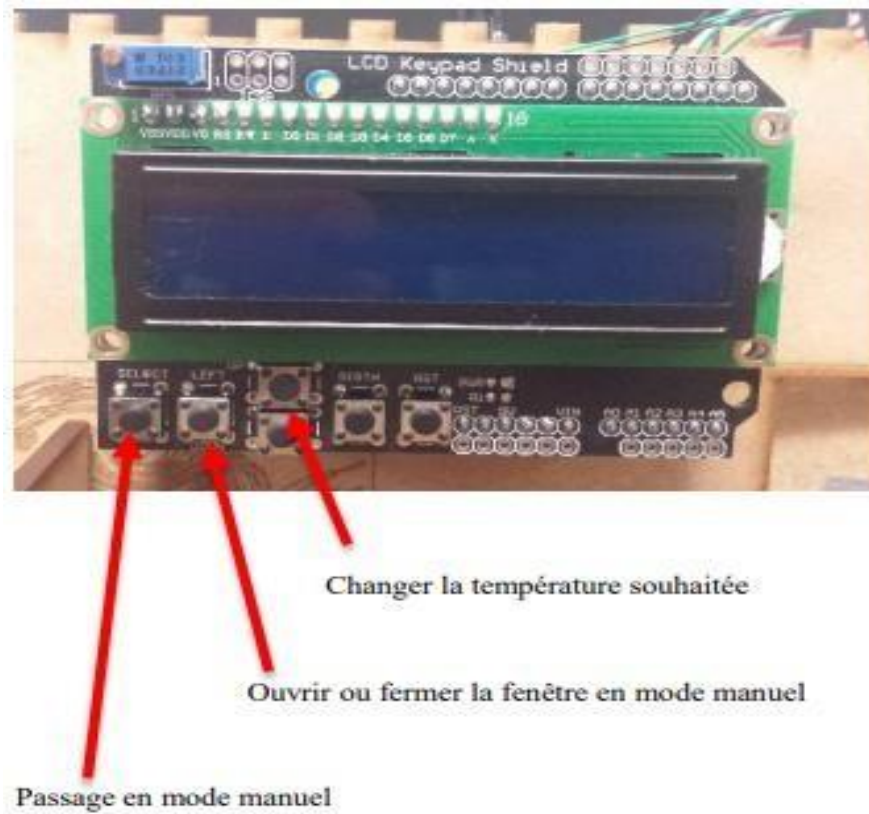
Ce dispositif permet d'améliorer la gestion d'énergie de la maison. En effet la dépense de chauffage ou de climatisation est une part importante du budget énergétique. Ainsi, afin de réduire ce coût, nous avons créé un dispositif prenant en charge l'ouverture et la fermeture des fenêtres en fonction d'une température réglée par l'utilisateur. Il compare la température intérieur et la température extérieure et décide d'ouvrir ou non la fenêtre. Nous avons utilisé comme matériel :

- Deux capteurs DHT 11
- Une shield LSD avec 4 boutons selectors

- Un cerveau moteur
- Des piles ou une batterie
- Une carte arduino



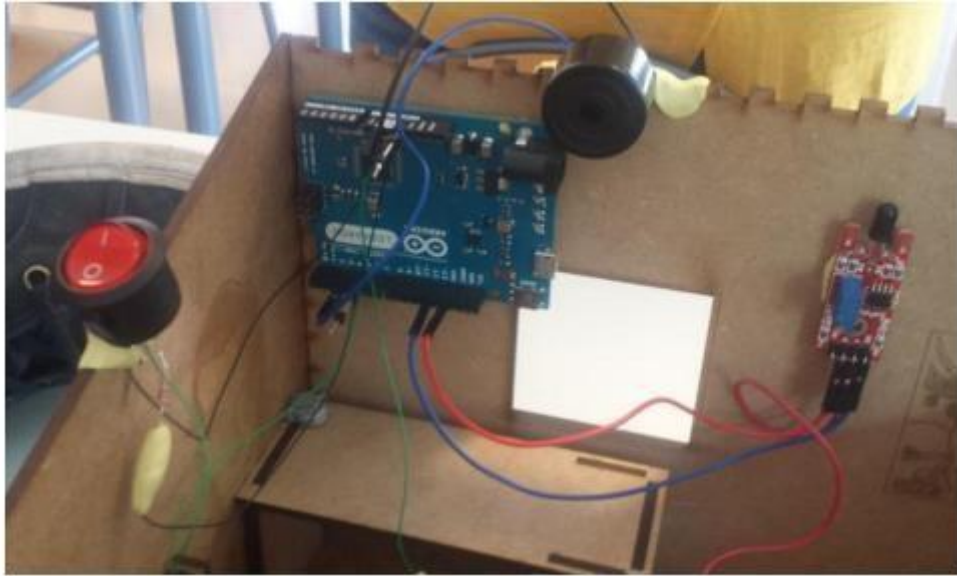




### c) Fire Alarm

La sécurité est devenue un élément primordial dans le choix d'une maison. Et l'une des plus grande crainte d'accident reste l'incendie. Ainsi nous avons associé différents composants afin de créer un détecteur de flamme. Ce détecteur de flamme déclenche une alarme. Le matériel utilisé est le suivant :

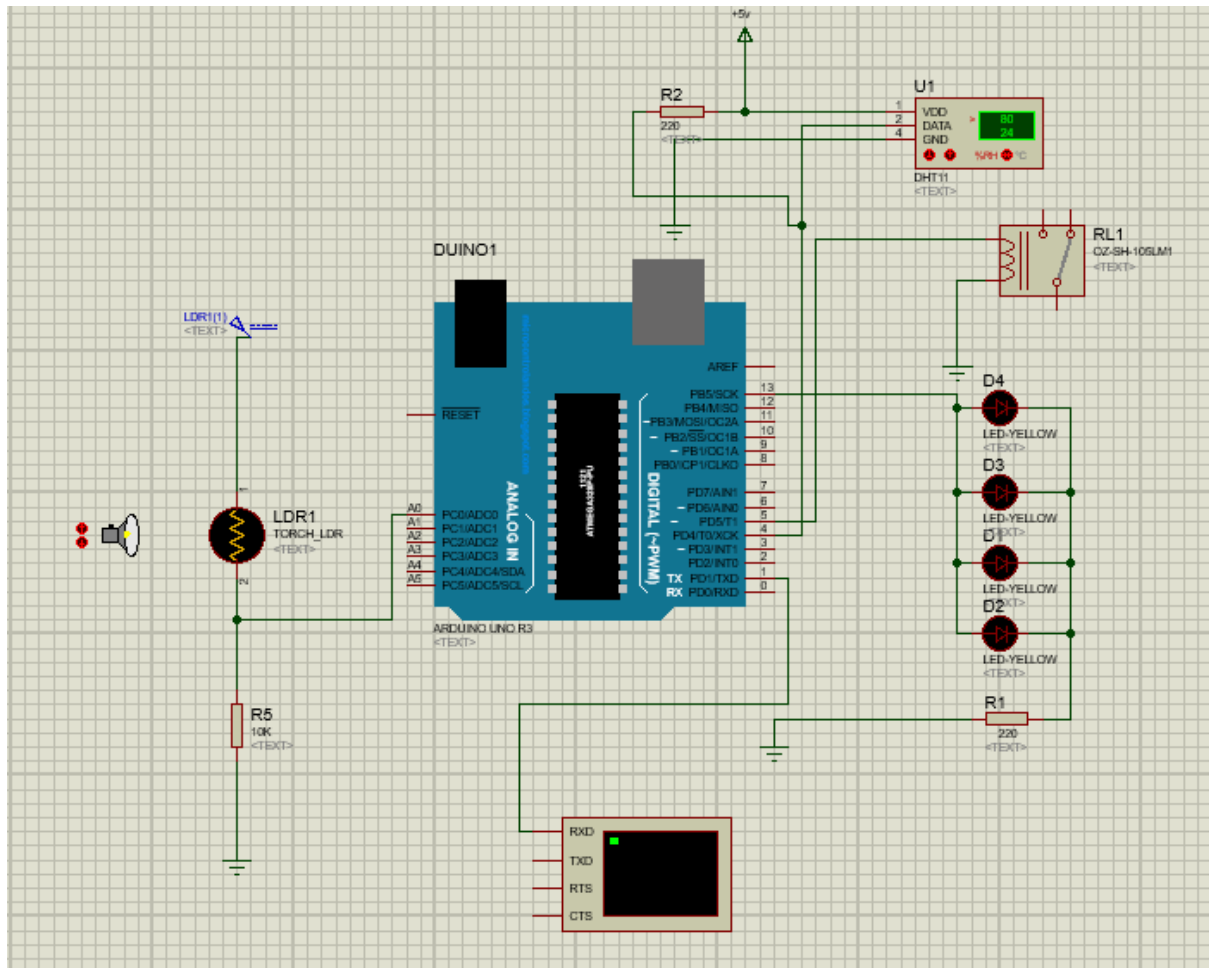
- Un détecteur de flammes
- Un buzzer
- Une carte arduino
- Un bouton poussoir



## CHAPITRE 2

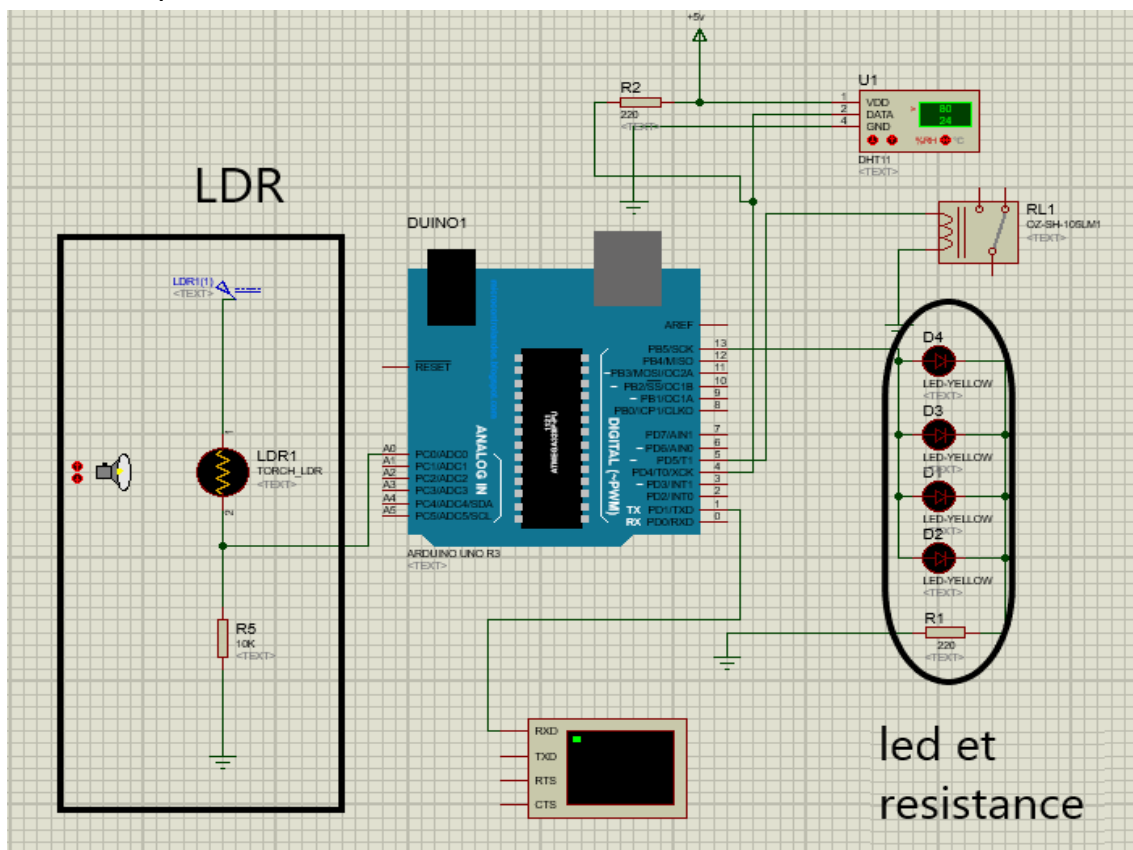
### Simulation sur Isis Proteus

#### I) Contrôle automatique de l'éclairage et la ventilation



## a) Contrôle automatique de l'éclairage

La gestion de la consommation d'énergie est une tendance actuelle, c'est pourquoi nous avons modélisé un premier scénario qui respecte cette tendance à une échelle réduite. Notre dispositif permet d'allumer automatiquement la lumière en fonction de la luminosité ambiante.



Le matériel que nous avons utilisé est le suivant :

- Une carte Arduino Uno
- Deux LED (lampe)
- Une résistance 220 ohm
- Une résistance 10k ohm
- Une photorésistance (LDR)

La partie du code qui concerne cette fonctionnalité :

```

int LUM=13;
int sun=A0;
int valeursun;
void setup() {
  pinMode(LUM,OUTPUT);
  pinMode(sun,INPUT);
}
void loop() {
  valeursun=analogRead(sun);
  Serial.println(valeursun);
  if(valeursun<160)
  {
    digitalWrite(LUM,HIGH);
  }else{
    digitalWrite(LUM,LOW);
  }
}

```

Les instructions du code qui concerne chaque matériel :**Photoresistance ldr :**

```

int sun=A0;
int valeursun;
pinMode(sun,INPUT);
valeursun=analogRead(sun);
Serial.println(valeursun);
if(valeursun<160)

```

**LED :**

```

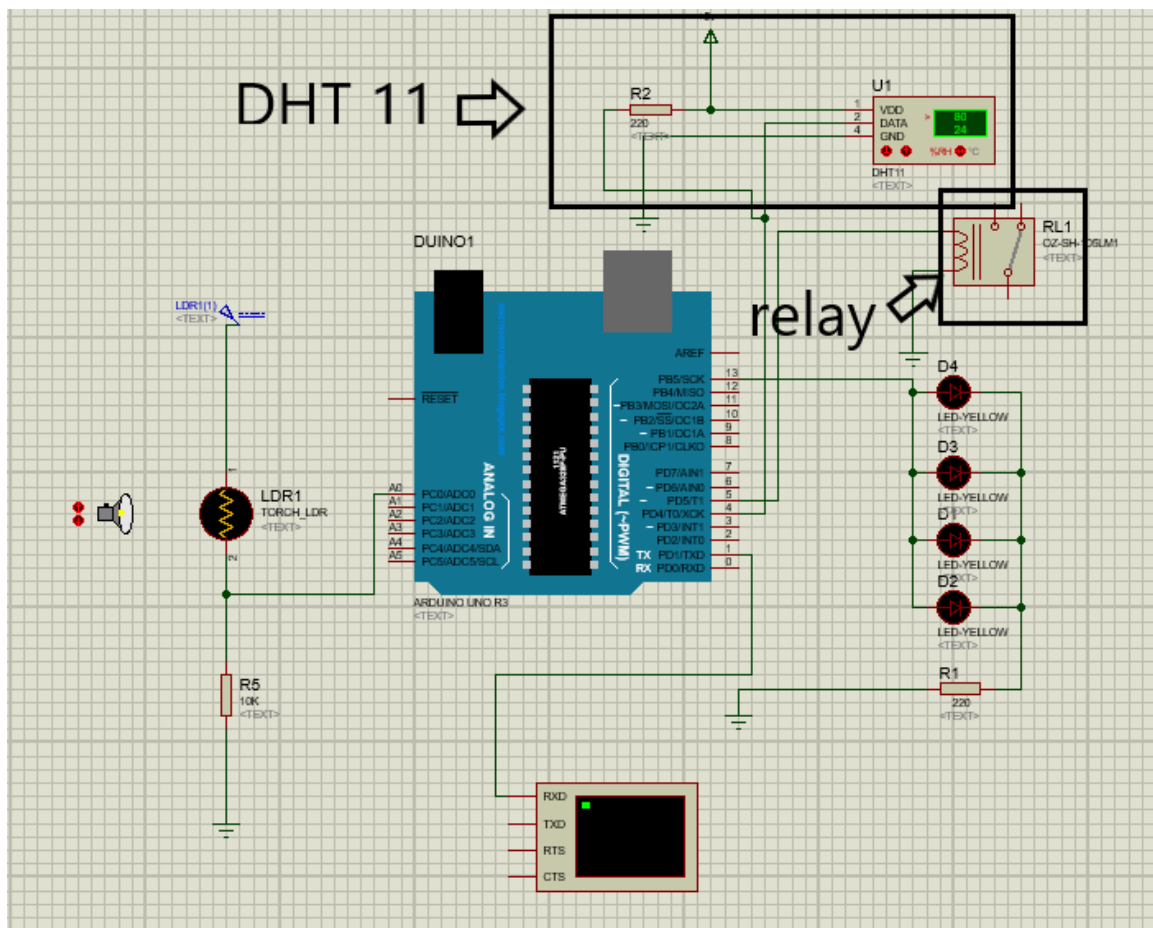
int LUM=13;

```

```
pinMode(LUM,OUTPUT)
{
  digitalWrite(LUM,HIGH);
}else{
  digitalWrite(LUM,LOW);
}
```

## b) Ventilation

Ce dispositif permet d'améliorer la gestion d'énergie de la maison. En effet la dépense de chauffage ou de climatisation est une part importante du budget énergétique. Ainsi, afin de réduire ce coût, nous avons créé un dispositif prenant en charge d'allumer ou éteindre le climatiseur en fonction d'une température réglée.



Le matériel que nous avons utilisé est le suivant :

- Une carte Arduino Uno
- Un capteur DHT 11
- Un Relay

La partie du code qui concerne cette fonctionnalité :

```
#include <SimpleDHT.h>

int clim=5;
int pinDHT11 = 2;
SimpleDHT11 dht11(pinDHT11);
void setup() {
  pinMode(clim,OUTPUT);
  Serial.begin(9600);

}

void loop() {
  byte temperature = 0;
  byte humidity = 0;
  dht11.read(&temperature, &humidity, NULL);
  Serial.println(temperature);
  if(temperature>20){
    digitalWrite(clim,HIGH);
  }else{
    digitalWrite(clim,LOW);
  }
  delay(2000);
}
```

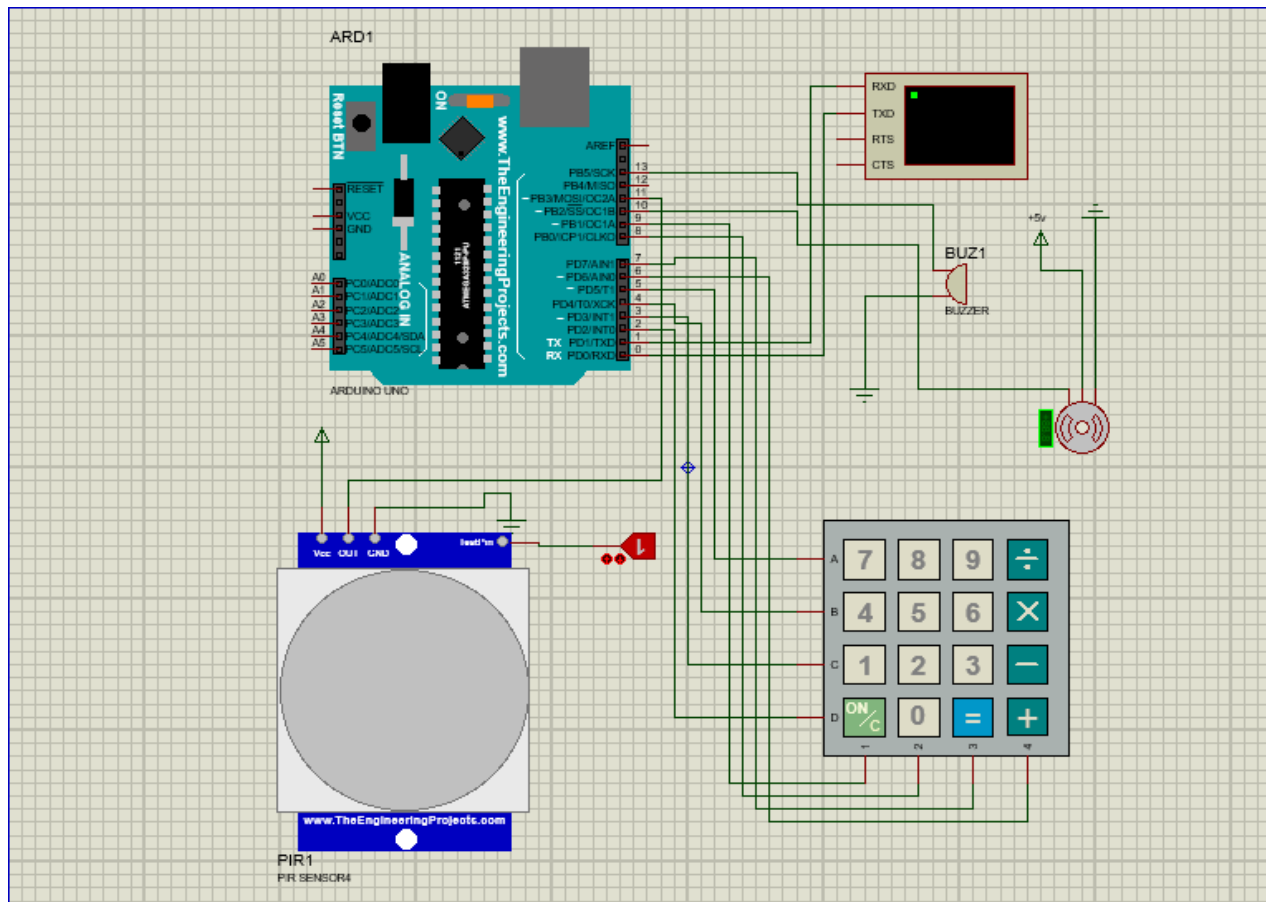
Les instructions du code qui concerne chaque matériel :**Capteur DHT 11:**

```
#include <SimpleDHT.h>
int pinDHT11 = 2;
SimpleDHT11 dht11(pinDHT11);
byte temperature = 0;
  byte humidity = 0;
```



```
dht11.read(&temperature, &humidity, NULL);  
Serial.println(temperature);  
if(temperature>20){  
Relay :  
int clim=5;  
pinMode(clim,OUTPUT);  
if(temperature>20){  
    digitalWrite(clim,HIGH);  
}else{  
    digitalWrite(clim,LOW);  
}  
delay(2000);  
}
```

## II) Contrôle d'accès



La sécurité est devenue un élément primordial dans le choix d'une maison. Ce dispositif permet d'améliorer la sécurité de la maison intelligente, en cas de présence de mouvement il faut saisir le code valide pour que la porte s'ouvre automatiquement pour 10 sec puis elle se ferme.

Si le code est invalide ou aucun code n'a été saisi pendant 20 sec, l'alarme sonne.

Pour arrêter l'alarme il faut entrer le code valide.

Le matériel que nous avons utilisé est le suivant :

- Une carte Arduino Uno
- Un détecteur de mouvement (SR505)
- Une keypad 4\*4
- Un servo moteur
- Active buzzer

La partie du code qui concerne cette fonctionnalité :

```

#include <Servo.h>

#include <Keypad.h>

const byte ROWS = 4;

const byte COLS = 4;

char keys[ROWS][COLS] = {

    {'A','3','2','1'},

    {'B','6','5','4' },

    {'C','9','8','7'},

    {'D','#','0','*'}

};

byte rowPins[ROWS] = {5, 4, 3, 2};

byte colPins[COLS] = {9, 8, 7, 6};

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins,
ROWS, COLS );

char pin[4]={'8','8','8','8'};

char code[4];

char key;

int i=0;

int a=0;

int b;

long movement;

```

```
int alarm=13;

Servo myservo;

int pirsensor=11;

void setup() {

    Serial.begin(9600);

    myservo.attach(10); //myservo.write(val); pour controller le servo
    val=angle en degree 0 180

    pinMode(pirsensor,INPUT);
    pinMode(alarm,OUTPUT);
}

void loop() {

    if(digitalRead(pirsensor)==HIGH){

        i=0;

        digitalWrite(alarm,HIGH);

        delay(50);

        digitalWrite(alarm,LOW);

        delay(35);

        digitalWrite(alarm,HIGH);

        delay(50);
```

```
digitalWrite(alarm,LOW);  
movement=millis();  
Serial.println("movement detected");  
Serial.println("entrez le code");  
while(i<4&&millis()-movement<20000){  
    key = keypad.getKey();  
    if (key!= NO_KEY){  
        code[i]=key;  
        digitalWrite(alarm,HIGH);  
        delay(50);  
        digitalWrite(alarm,LOW);  
        Serial.print(code[i]);  
        i++;  
    }  
}  
if(i==4)  
{  
    Serial.println();  
    for(a=0;a<4;a++){  
        Serial.print(code[a]);  
    }  
}
```

```
Serial.println();

}

b=0;

if(code[0]==pin[0]&&code[1]==pin[1]&&code[2]==pin[2]&&code[3]=
=pin[3]){

    b=1;

}

while(b==0){

    digitalWrite(alarm,HIGH);

    Serial.println("invalide code");

    i=0;

    while(i<4){

        key = keypad.getKey();

        if (key!= NO_KEY){

            code[i]=key;

            Serial.print(code[i]);

            i++;

        }

    }

    if(code[0]==pin[0]&&code[1]==pin[1]&&code[2]==pin[2]&&code[3]=
=pin[3]){
```

```
b=1;

}

}

while(code[0]==pin[0]&&code[1]==pin[1]&&code[2]==pin[2]&&code[3]==pin[3]){

    digitalWrite(alarm,LOW);

    Serial.println("valide code");

    myservo.write(10);

    delay(10000);

    myservo.write(90);

    delay(2000);

    code[0]='x';

    code[1]='x';

    code[2]='x';

    code[3]='x';

}

}else{

    Serial.println("no movement");

}

delay(10000);

}
```

## Les instructions du code qui concerne chaque matériel :

### **Détecteur de mouvement (SR505) :**

```
int pirsensor=11;

pinMode(pirsensor,INPUT);

if(digitalRead(pirsensor)==HIGH){
```

### **Keypad 4\*4 :**

```
#include <Keypad.h>

const byte ROWS = 4;

const byte COLS = 4;

char keys[ROWS][COLS] = {

    {'A','3','2','1'},

    {'B','6','5','4' },

    {'C','9','8','7'},

    {'D','#','0','*'}

};

byte rowPins[ROWS] = {5, 4, 3, 2};

byte colPins[COLS] = {9, 8, 7, 6};

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

while(i<4&&millis()-movement<20000){

    key = keypad.getKey();

    if (key!= NO_KEY){

        code[i]=key;

        digitalWrite(alarm,HIGH);
```



```

delay(50);

digitalWrite(alarm,LOW);

Serial.print(code[i]);

i++;

}

}

while(i<4){

    key = keypad.getKey();

    if (key!= NO_KEY){

        code[i]=key;

        Serial.print(code[i]);

        i++;

    }

```

### **Servo moteur :**

```

#include <Servo.h>

Servo myservo;

myservo.attach(10);

myservo.write(10);

delay(10000);

myservo.write(90);

delay(2000);

```

### **Active buzzer :**

```

digitalWrite(alarm,HIGH);

    delay(50);

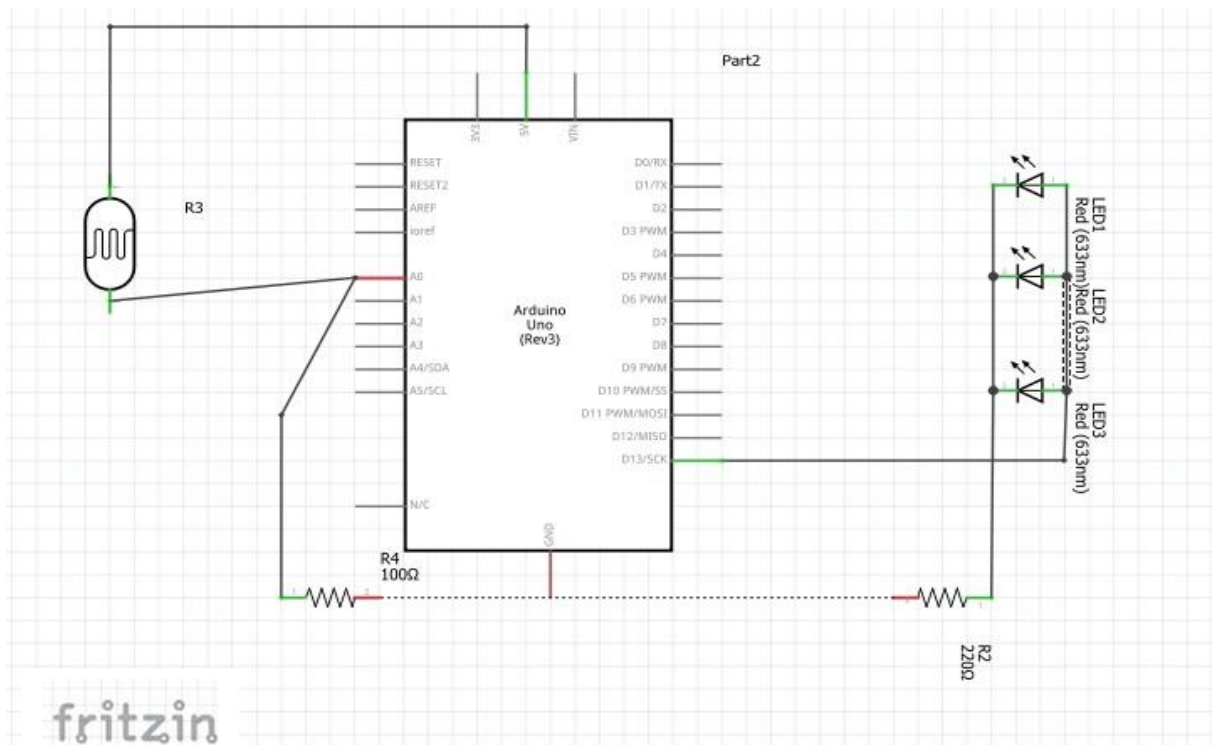
```

```
digitalWrite(alarm,LOW);  
delay(35);  
digitalWrite(alarm,HIGH);  
delay(50);  
digitalWrite(alarm,LOW);  
if (key!= NO_KEY){  
code[i]=key;  
digitalWrite(alarm,HIGH);  
delay(50);  
digitalWrite(alarm,LOW);  
Serial.print(code[i]);  
i++;  
}  
digitalWrite(alarm,HIGH);
```

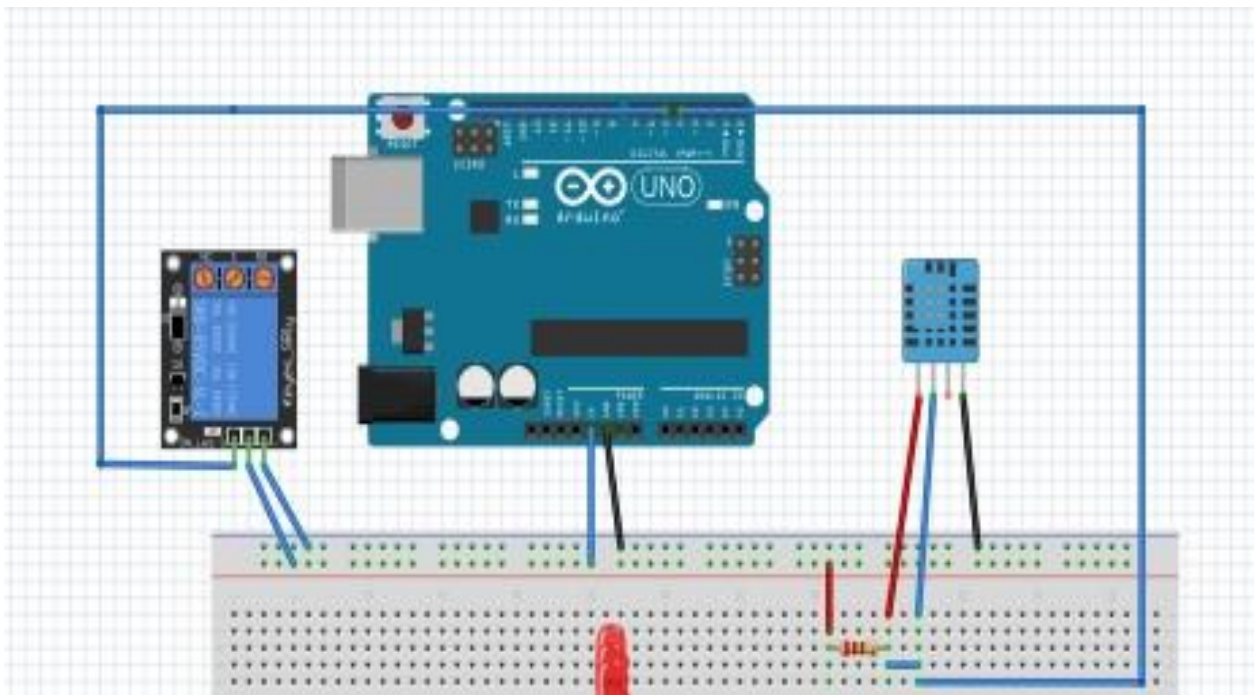
## Montage su fritzing

### a) Contrôle automatique de l'éclairage

Montage électronique :

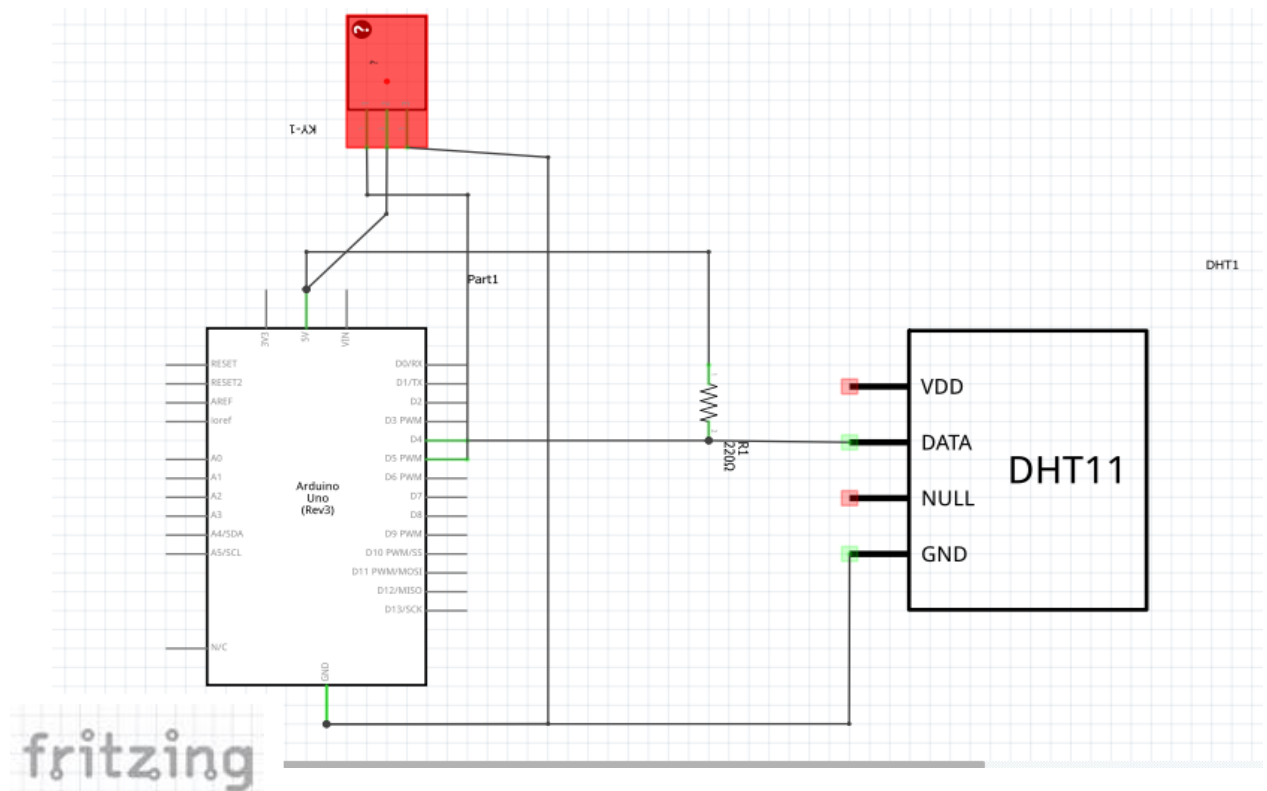


Breadboard:

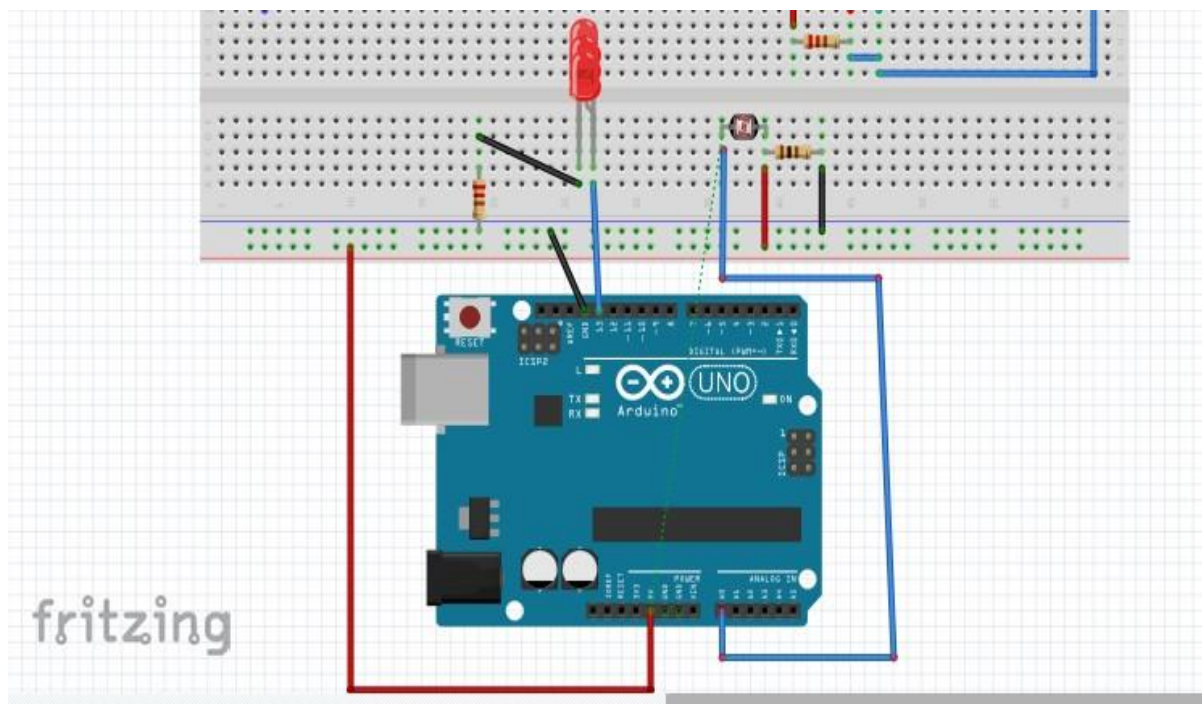


## b) Ventilation

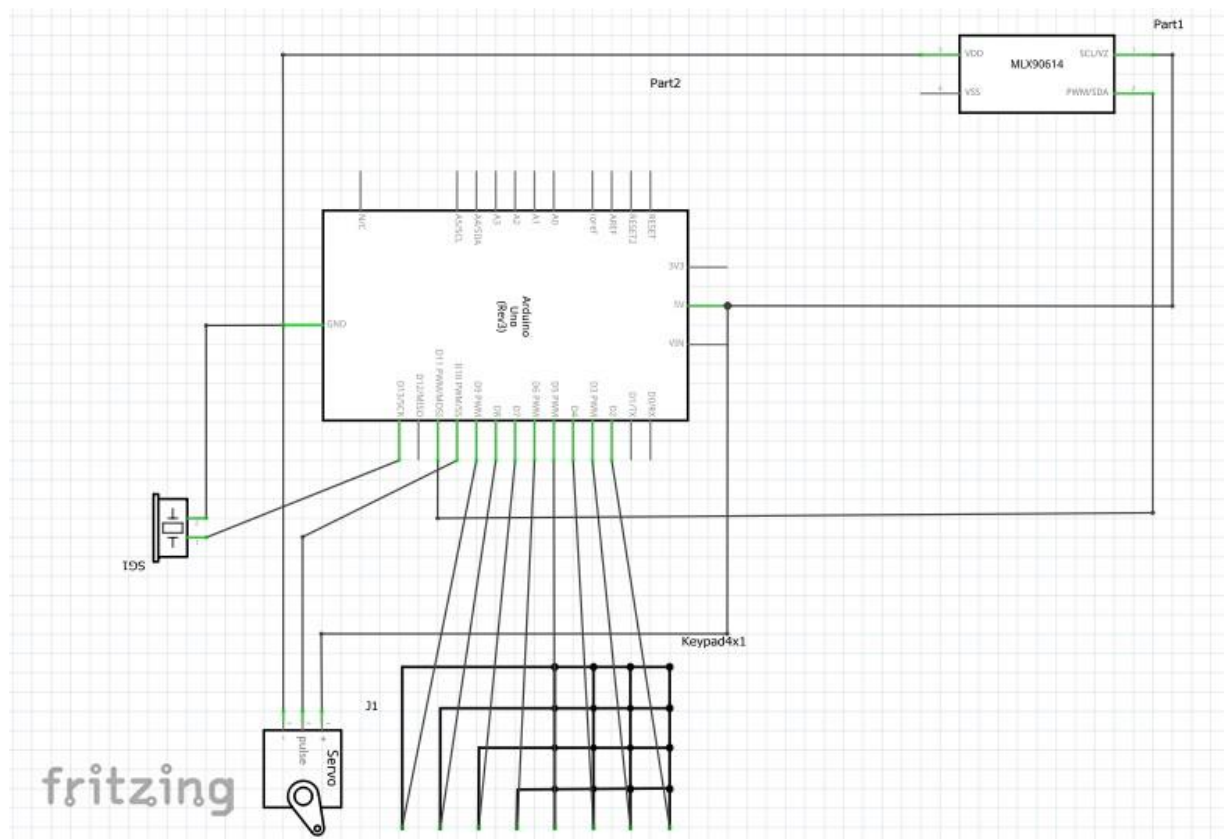
Montage électronique :



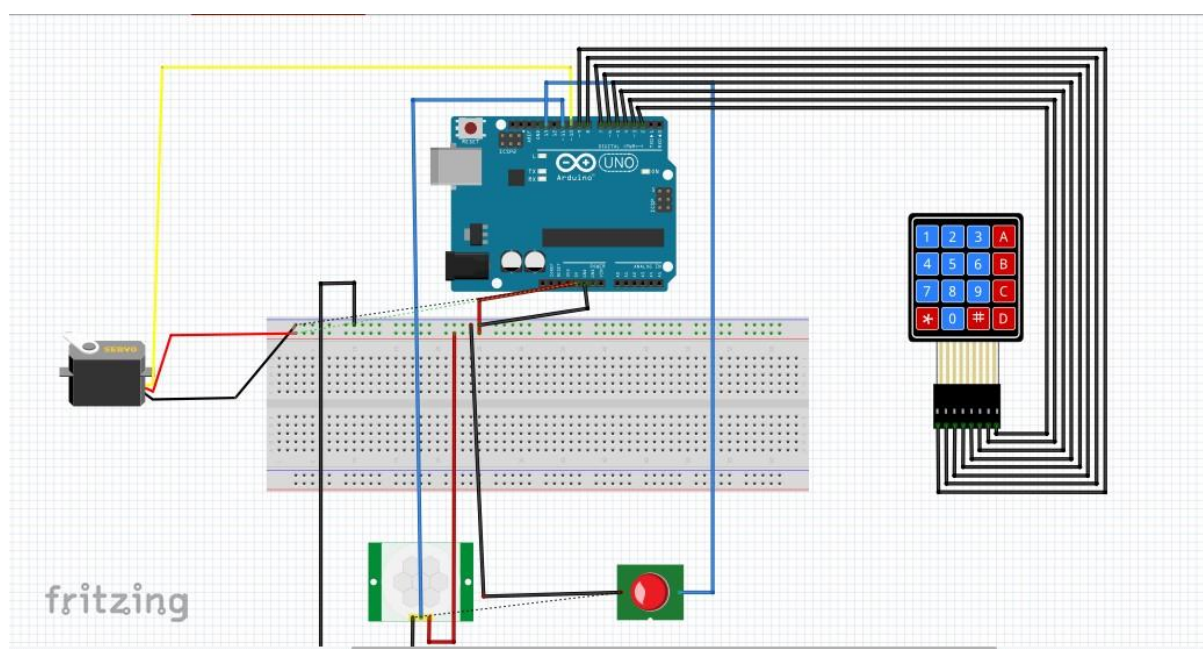
Breadboard:



c) **Contrôle d'accès :**  
montage électronique :



Breadboard:



## **CHAPITRE 3**

### **Caractéristique du matériel utilisée**

#### **Capteur dht11**

Le DHT11 est un capteur de Température et d'Humidité très apprécié pour sa simplicité de mise en œuvre et son coût peu élevé. Il ne requiert qu'une résistance de tirage et une alimentation 3V ou 5V pour fonctionner. Sa programmation est facile à l'aide des librairies Arduino.



Figure : Capteur de température et d'humidité DHT11

#### **Caractéristiques**

- Alimentation : 3 à 5 Vcc
- Consommation maximale : 2.5 mA.
- Plage de mesure : - température : 0 à +50 °C. - humidité : 20 à 100 % HR.
- Précision : - température :  $\pm 2$  °C. - humidité :  $\pm 5$  % HR.

#### **Capteur PIR sensor**

Le capteur de mouvement PIR (Passive Infrared Sensor) est un capteur électronique qui mesure la lumière infrarouge (IR) rayonnant à partir d'objets dans son champ de vision. Ils sont très souvent utilisés dans les systèmes d'alarmes ou de détection de présence pour leur faible coût et leur efficacité.



Figure : Pir sensor HC-SR505

### Caractéristiques

- Voltage : 5-12VDC.
- Output : 3.3V TTL.
- Détection Distance : 3-7mt (approx, ajustable).
- Delay Time: 5-200s (adjustable).
- Trigger: L: non repeatable trigger - H: repeatable trigger.

### Photorésistance (LDR):

LDR (résistance variant à la lumière, ou photorésistance) agit comme une résistance variable, dont la résistance varie en fonction de la lumière perçue. Le module capteur LDR génère une sortie 5V dans l'obscurité et 0V en pleine lumière, La sensibilité de la LDR peut être ajustée via un potentiomètre ajustable placé sur le module.



Figure : LDR



## Servo Moteur :

Un servomoteur contient un moteur à courant continu associé à une série d'engrenages qui va lui permettre de gagner une puissance. Mais comme rien n'est gratuit, ce gain en puissance réduit sa vitesse de rotation. Le servomoteur est souvent limité dans sa rotation. En effet, il ne peut tourner que d'un demitour, soit 180°. Il peut donc prendre 180 positions et les tenir.



Figure : Servomoteur 5V et son symbole.

Un servomoteur a trois fils :

- Un fil rouge se connecte à l'alimentation.
- Un fil noir (parfois marron) se connecte à la masse.
- Un fil jaune (parfois orange ou blanc) se connecte à n'importe quelle sortie numérique de l'Arduino qui sert à son contrôle.

## Relay :

Le relais est un composant électromécanique, il joue le rôle d'un interrupteur, mais la commande de ce genre d'interrupteur n'est pas manuel, la commande se fait par l'application d'un effort approprié sur le relais, cet effort soit 5 volts ou 9 volts, soit 12 volts ou 220 volts.

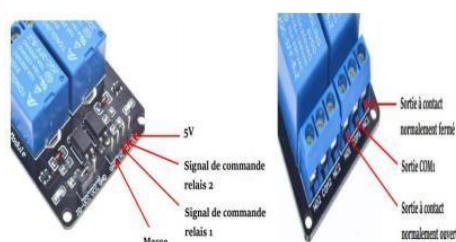


Figure : Broches de sortie et d'entrée de module relais.



## Keypad 4\*4:

Keypad 4\*4 ou le clavier 4\*4 est un outil qui permet de saisir des chiffres et des symboles et de contrôler de nombreuses applications, de telle que les calculs, le code, etc. ils sont pratiquement partout. Le clavier comprend seize touches disposées en quatre colonnes, l'appui sur une touche fait communiquer une ligne avec une colonne.

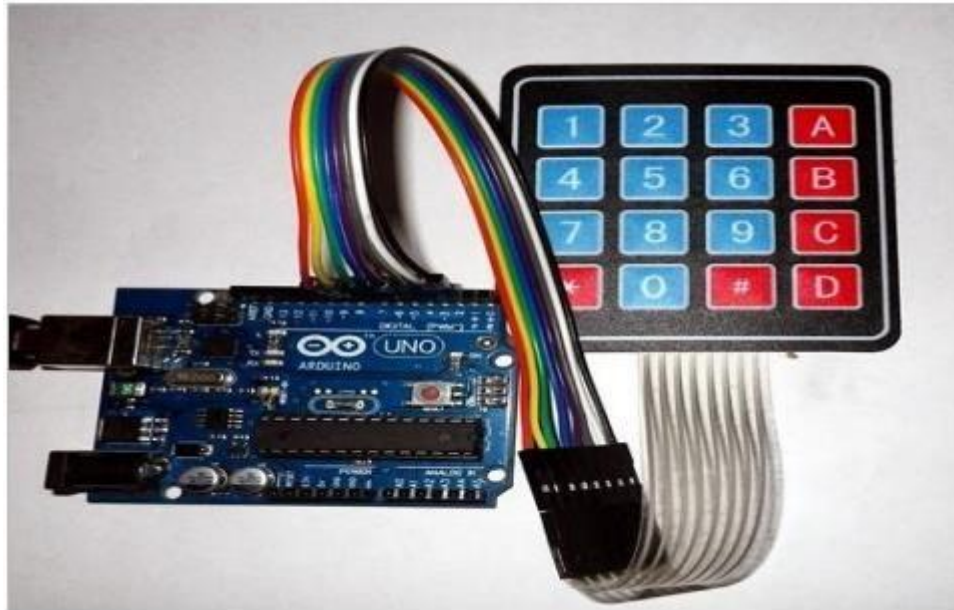


Figure : keypad 4\*4

## Active buzzer :

C'est un composant électromécanique ou piézoélectrique, qui produit un son distinct lorsqu'on lui applique une tension, la fréquence de ce signal sonore est alors relative au signal appliqué. Il y a deux types de buzzer, actif et passif, la différence entre eux, c'est que les buzzer actif fonctionne en tension continu, alors que les buzzer passif fonctionne seulement en tension alternatif.



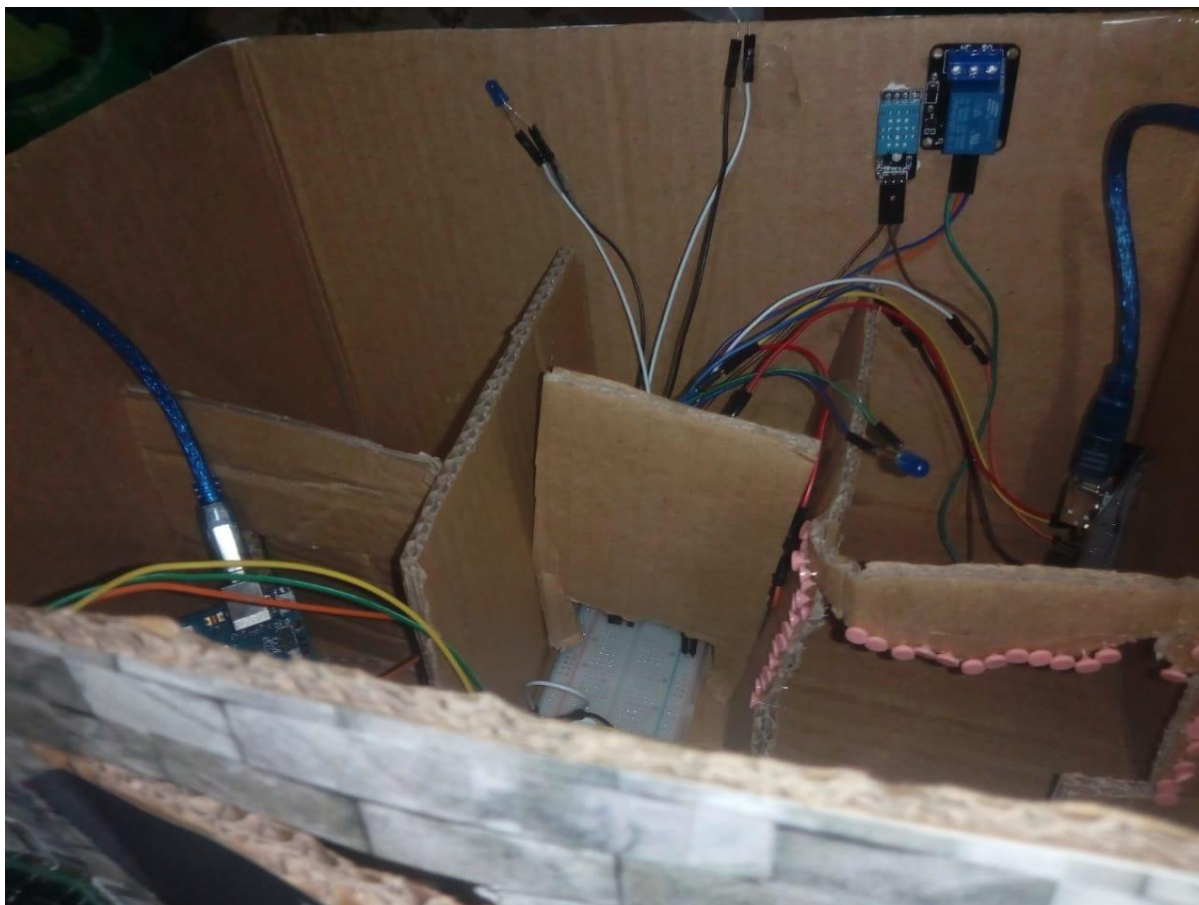
Figure : active buzzer

## Photo de la réalisation de la maison intelligente

### Vue générale de la maison



## La ventilation et l'éclairage :

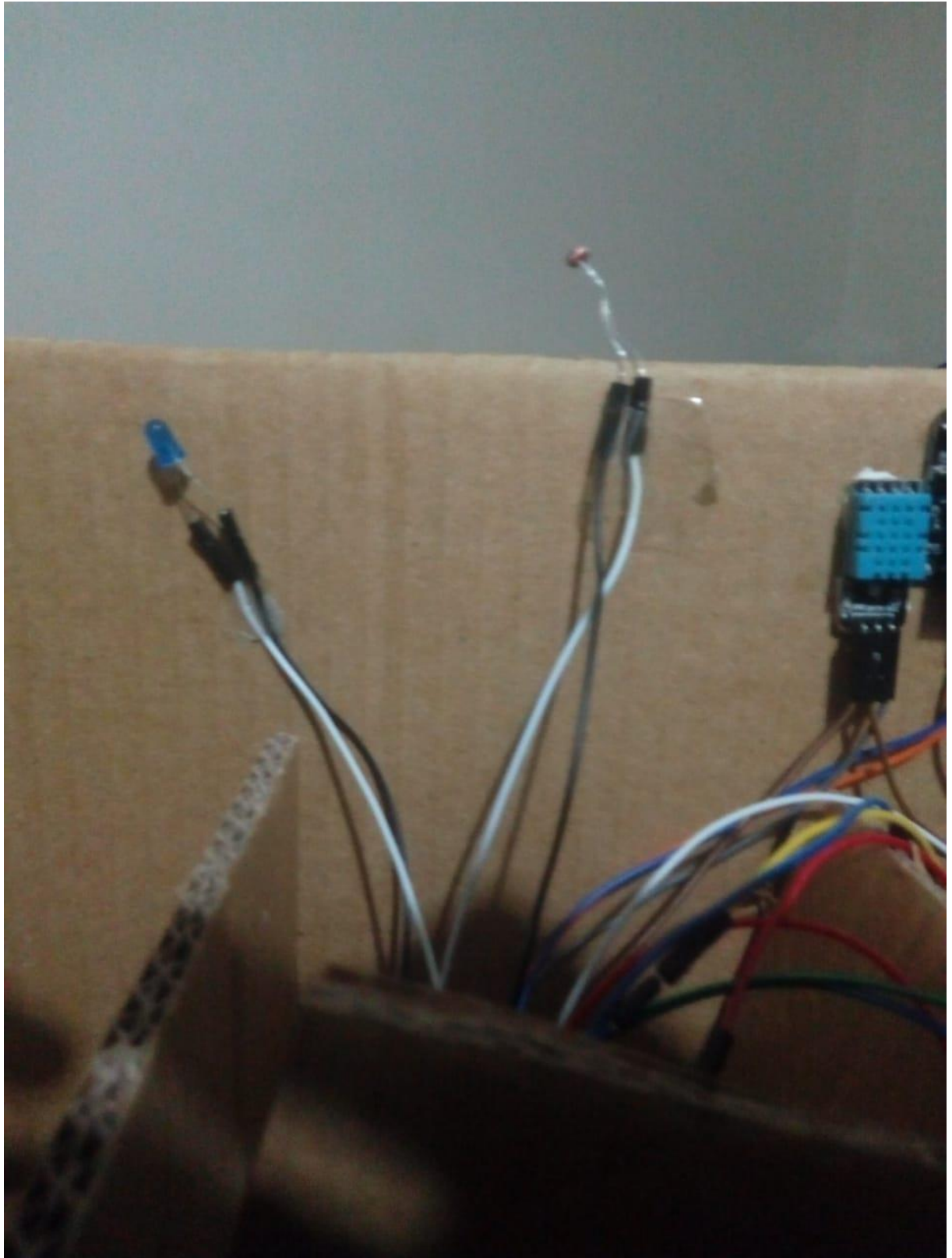




## La ventilation :



## L'éclairage :





## Contrôle d'accès :



## **Conclusion :**

La Smart Home ne se limite pas à l'espace de ses quatre murs. Elle est poussée par un contexte énergétique en pleine évolution et par la révolution des objets connectés, la Smart Home est en train de devenir une brique incontournable du système électrique.

Le but étant de développer un mode de vie basé sur l'anticipation des besoins, l'optimisation des consommations d'énergie et plus encore.