



## Projet SI

lien de la vidéo:

<https://www.youtube.com/watch?v=iSR2wQu6RJo>: Allumer et éteindre une lampe avec son téléphone !!!

Problème:

Les gestes qui sont pour nous simples comme ouvrir une porte, lire une affiche, tourner les pages d'un livre et allumer une lampe deviennent démesurément compliqués pour les personnes à mobilité réduite.

Dans ce projet, on fait appel à vous pour mettre en place et réaliser un système facilitant aux personnes à mobilité réduite de commander l'éclairage facilement.

Nous avons eu 12 heures en groupe de 4 pour réaliser ce projet :

« Réaliser un système d'éclairage commandé à partir d'un Smartphone »

Pour commencer nous avons réalisé des diagrammes SYSML afin de créer notre cahier des charges et de mieux visualiser les attentes.

Puis, nous avons réfléchi aux composants à utiliser afin de correspondre aux cahiers des charges.

Nous avons ensuite modélisé le montage à reproduire à l'aide du logiciel Proteus pour nous donner une idée et nous permettre de nous projeter dans le montage, la manipulation.

Nous avons créé une application pour smartphone grâce au logiciel MIT APP INVENTOR qui nous a permis de communiquer via la liaison bluetooth entre la carte arduino et notre téléphone.

Après cela nous avons créé un code arduino qui permettait d'allumer une led pour nous assurer que le système fonctionnait. La chaîne d'information était finie on pouvait passer à la chaîne d'énergie. (Voir vidéo)

Nous avons ensuite réalisé la chaîne d'énergie à l'aide du montage Proteus afin d'allumer une lampe 24V (nous avons finalement utilisé une lampe 220V) à la place de la LED.

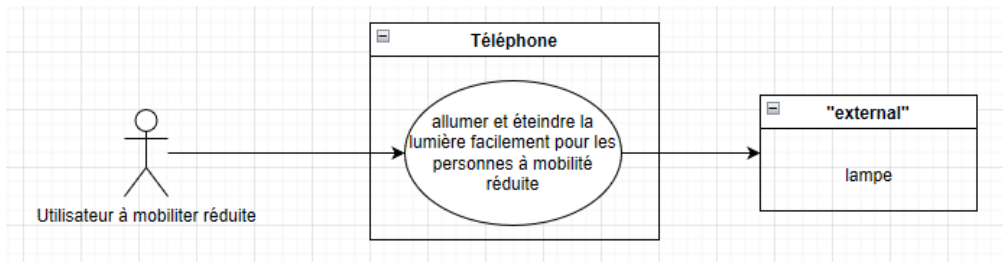
#### Liste des composants nécessaire au projet:

- Carte Arduino
- Téléphone Android
- Câble électrique
- Module Bluetooth compatible avec une carte Arduino: HC-05
- une Lampe
- une Douille
- Câble secteur
- Relais HW-482

Nous avons suivi nos montage fait en amont et nous avons réussi a allumer une lampe 24v.

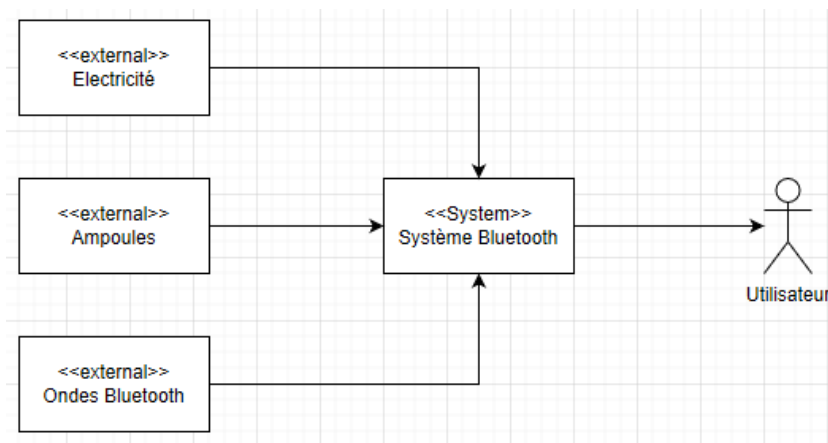
#### Diagrammes (SYSML):

Diagramme cas d'utilisation:



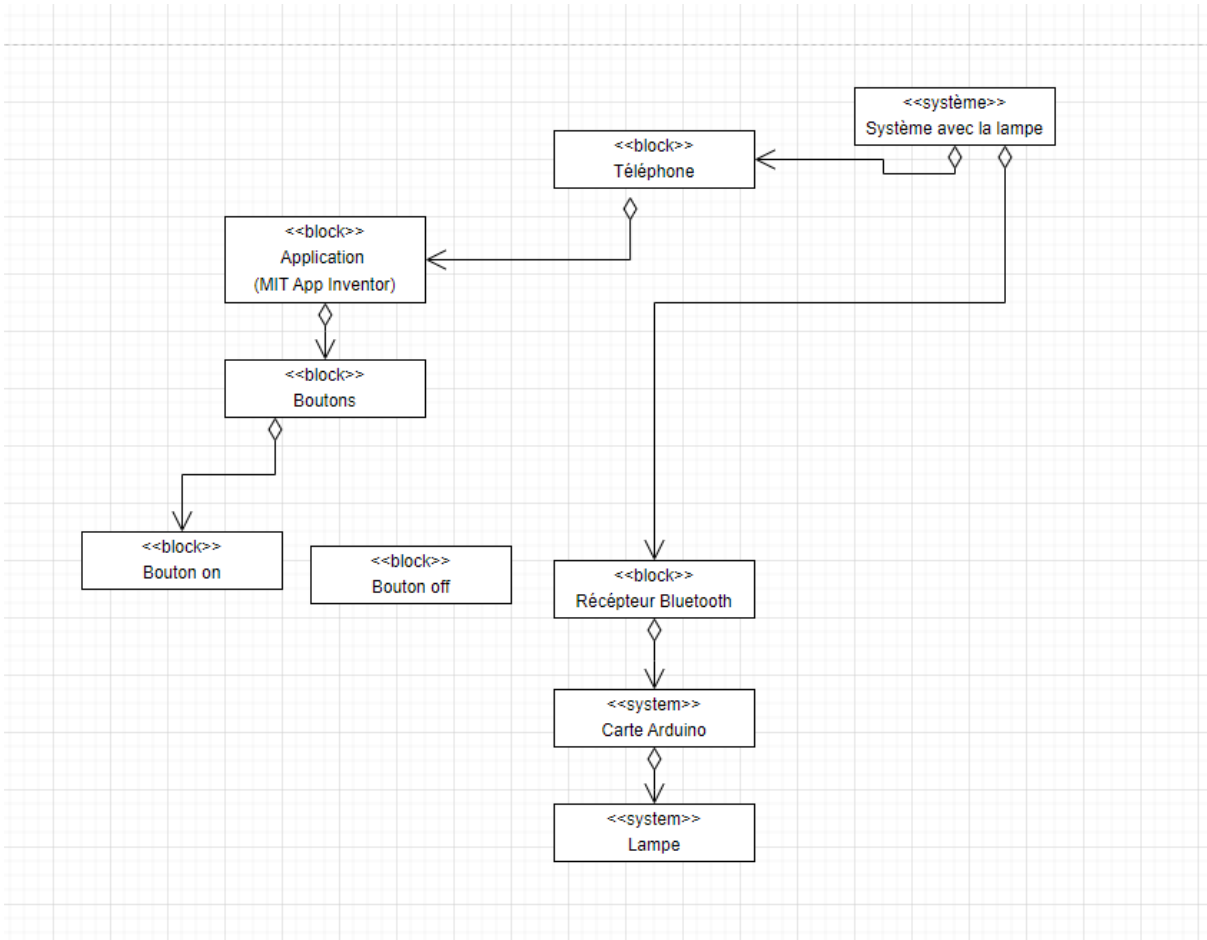
Ce diagramme nous indique les cas d'utilisation du système.

Diagramme contexte :



Ce diagramme sert à définir qui et quoi interagit avec le système.

Diagramme de définition des blocs:



Ce diagramme sert à indiquer de quoi est composé le système .

Diagramme d'état:

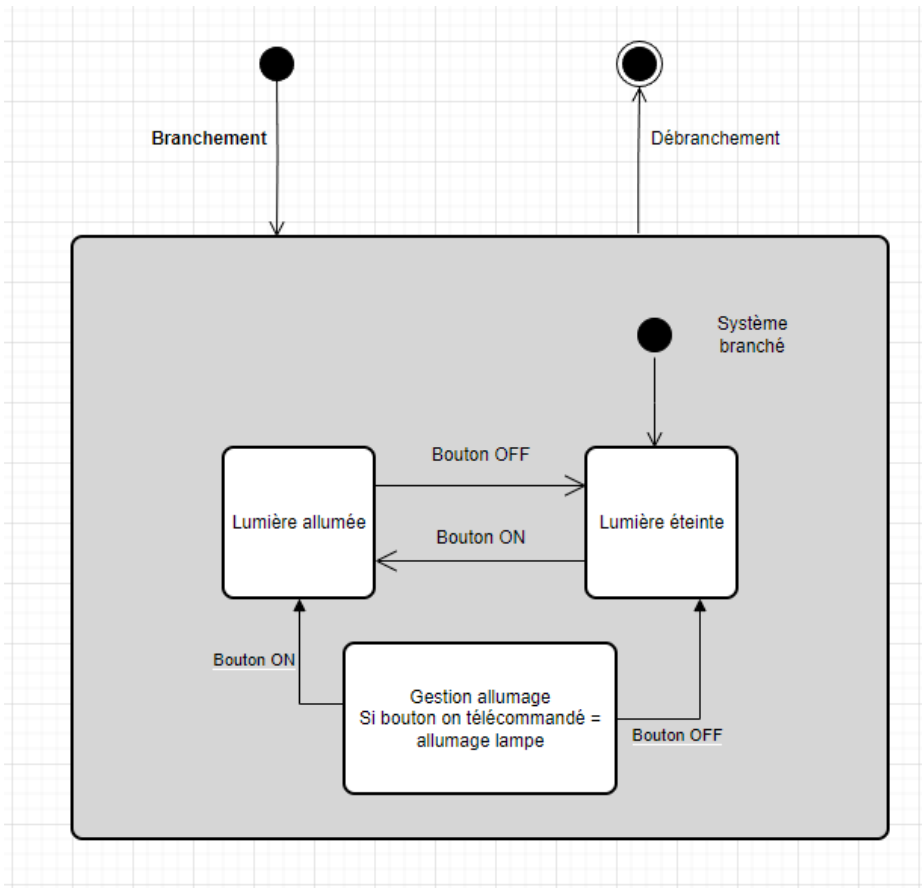
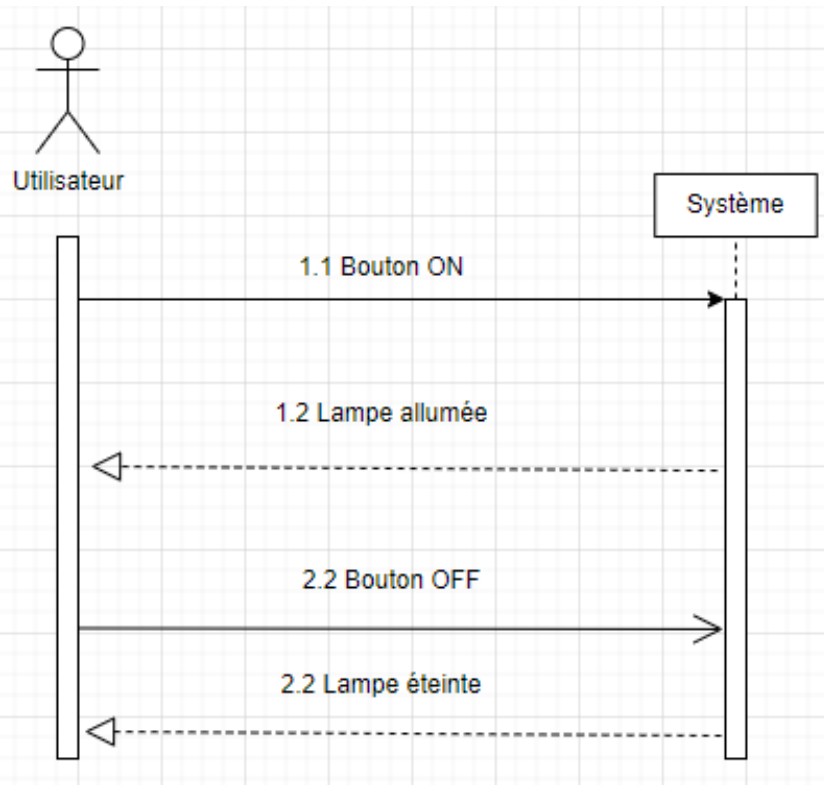


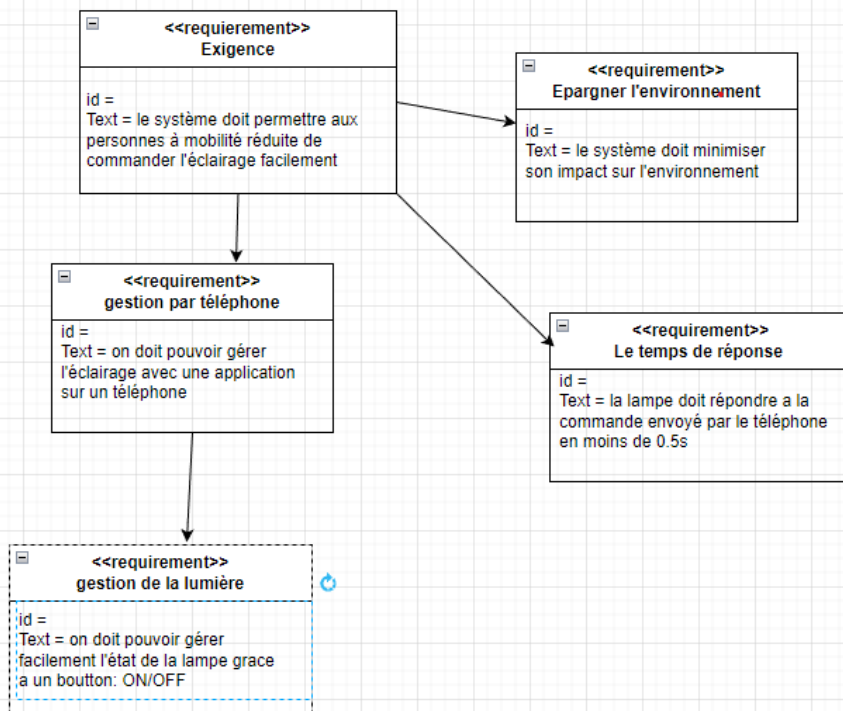
diagramme qui permet de savoir comment le système fonctionne de manière interne

Diagramme séquence:



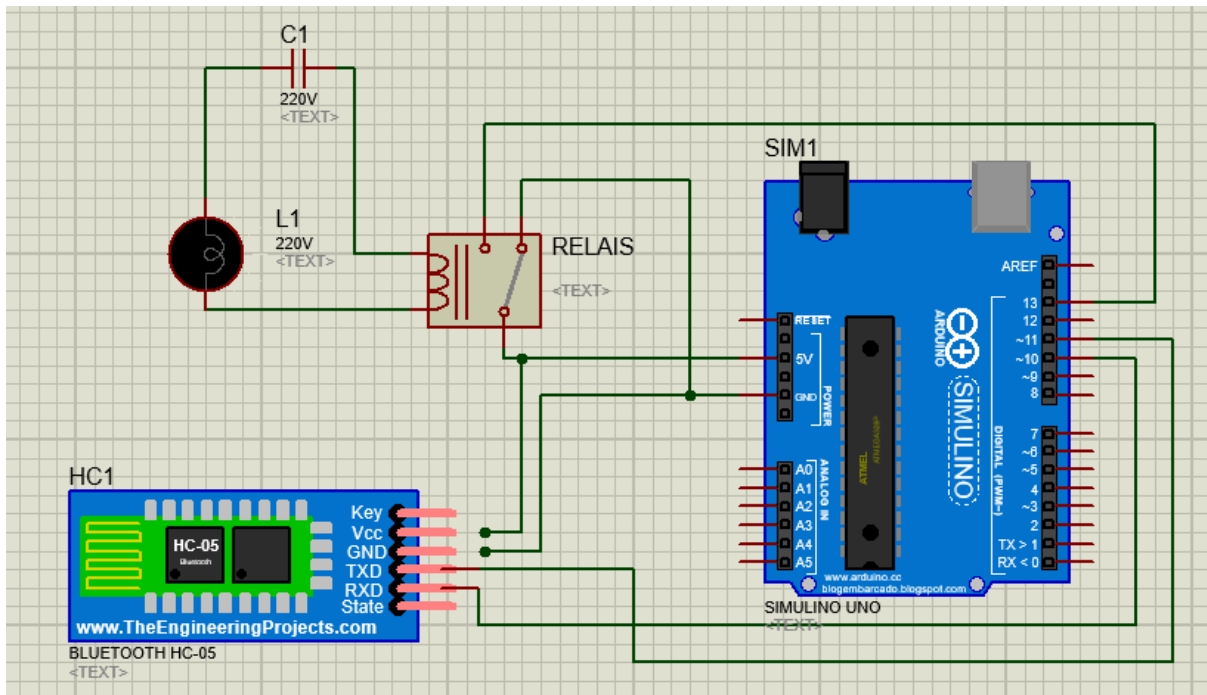
ce diagramme indique comment le système interagit avec les utilisateurs et les systèmes externes au cours du temps

Diagramme exigences:



ce diagramme donne les performances que doit atteindre le système en fin de conception (elles peuvent être esthétique, économique, environnementale, technique)

Circuit électrique:



L'information est reçue par le récepteur bluetooth qui est envoyé par le Txd au pin 11. En effet il y a un flux d'information du Txd au pin 11 qui revient du pin 10 qui finit au Rxd.

Ensuite, l'information est traitée par la carte arduino qui envoie l'information du pin 13 au relais (allumer/éteindre).

Puis, la lampe est reliée en secteur au relais (série). Le relais agit comme un interrupteur.

Enfin, l'information du pin 13 envoyé au relais est codée en binaire, c'est à dire qu'elle prend seulement 2 valeurs : soit 0 soit 1 (allumer ou éteindre)

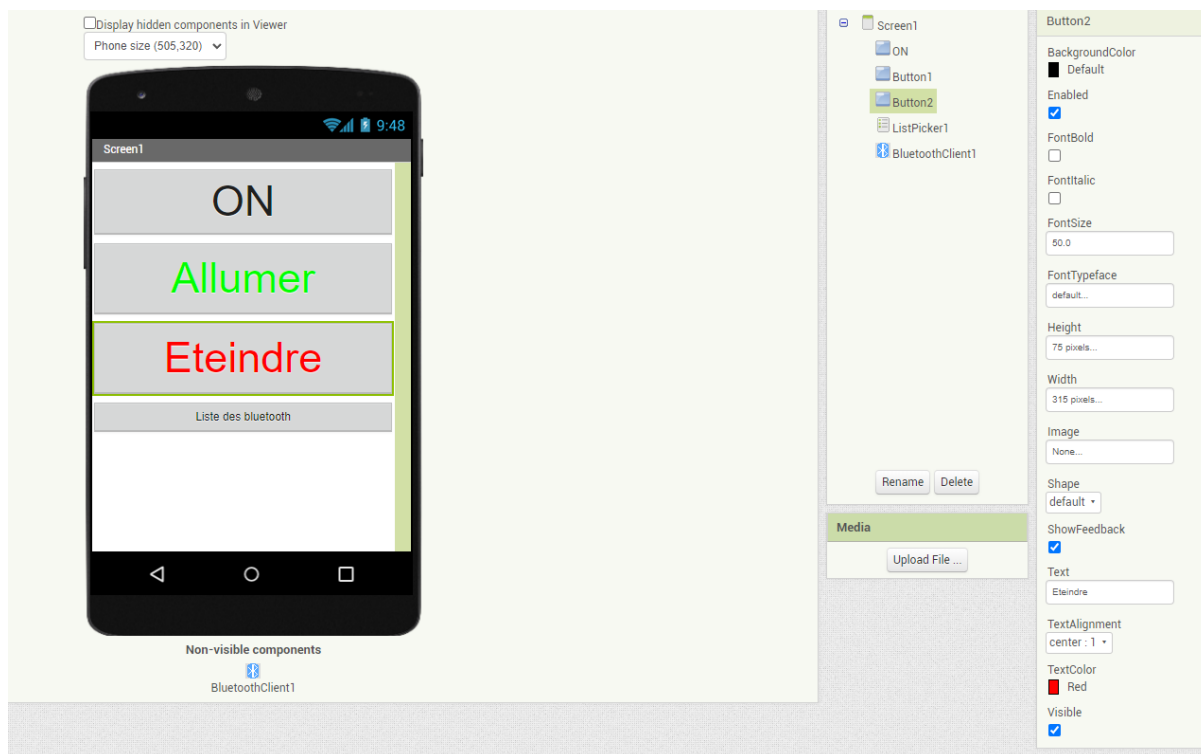
De plus, le relais est relié a la carte arduino et au récepteur bluetooth avec un courant en 5V en dérivation.

photo application :

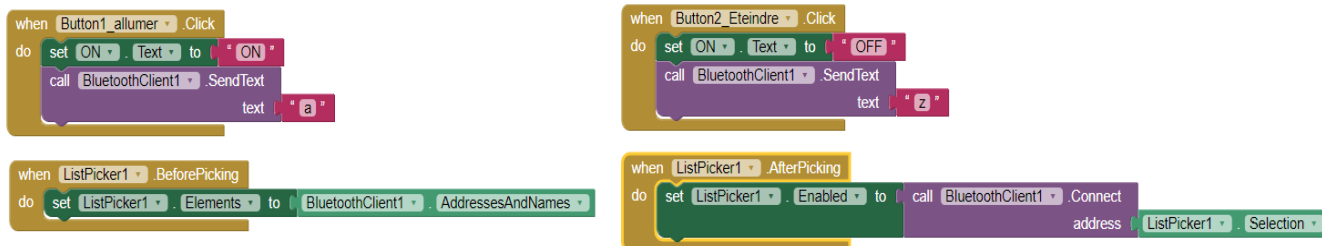


<http://ai2.appinventor.mit.edu/#5201755506606080>

## - photo interface



## - photo blocs



les 4 blocs sont des blocs de condition, le premier bloc dit quand le bouton 1 (allumer) est cliquer,

+

Le code :

```
#include <SoftwareSerial.h> //librairie qui permet d'ajouter une liaison série sur la carte(
ici c'est la liaison bluetooth )
```

SoftwareSerial BluetoothSerial(11,10); // permet de créer une liaison bluetooth pour communiquer avec le bluetooth

```
void setup() {
  pinMode(11, INPUT); // pin permettant de transmettre l'information pour que l'arduino puisse lire ce qui arrive sur ce pin
  pinMode(10, OUTPUT); // pin permettant d'écrire de l'information sur la liaison bluetooth
  pinMode(13, OUTPUT); //
  Serial.begin(9600); // communication avec le moniteur série
  BluetoothSerial.begin(9600); //communication avec le Bluetooth
}

void loop() {
  char var = 'z'; //création d'une variable à qui on attribut la valeur "z"
  if(BluetoothSerial.available()){ // on lit la liaison bluetooth
    var = (char) BluetoothSerial.read(); //caste en char
    trait_trame(var);
  }
}

void trait_trame(char car){
  if (car == 'a'){
    digitalWrite(13,HIGH); //allumer la LED
  }
  if (car == 'z'){
    digitalWrite(13,LOW); //eteindre la LED
  }
}
```

Nous avons mis et commenté le code qui nous a permis de résoudre le problème ci-dessus. Les "//" ne font pas partie du code, ce sont des commentaires que nous avons rajouté et qui permettent de comprendre le code.