

## Project IoT

### – Smart Home & Garden –

#### 1. Conectivitate ( [Tutorial video](#) )

##### 1.1. Dispozitive necesare

Accesam meniul din partea de jos a programului de la **End Devices** < **End Devices** apoi din partea dreapta selectam un **Server-PT**.

Accesam meniul din partea de jos a programului de la **Network Devices** < **Switches** apoi din partea dreapta selectam un **Switch 2950-24**.

Accesam meniul din partea de jos a programului de la **Network Devices** < **Wireless Devices** apoi din partea dreapta selectam un **Wireless Router WRT300N**.

Accesam meniul din partea de jos a programului de la **End Devices** < **End Devices** apoi din partea dreapta selectam un **Laptop-PT**.

Accesam meniul din partea de jos a programului de la **End Devices** < **End Devices** apoi din partea dreapta selectam un **SmartPhone-PT**.

##### 1.2. Conectare dispozitive

Accesam meniul din partea de jos a programului de la **Connections** < **Connections** apoi din partea dreapta selectam **Copper Straight-Through**. Apoi dăm click pe Server și selectam **FastEthernet0** și apoi dăm click pe Switch și apoi pe **FastEthernet0/1**.

Accesam meniul din partea de jos a programului de la **Connections** < **Connections** apoi din partea dreapta selectam **Copper Cross-Over**. Apoi dăm click pe Switch și selectam **FastEthernet0/2** și apoi dăm click pe Router și selectam **Ethernet1**.

Accesam meniul din partea de jos a programului de la **Connections** < **Connections** apoi din partea dreapta selectam **Copper Straight-Through**. Apoi dăm click pe Laptop și selectam **FastEthernet0** și apoi dăm click pe Switch și apoi pe **FastEthernet0/3**.

##### 1.3. Configurare server

- **Server** < IP Configuration < IPv4 Address : 192.168.0.10.
- Click Subnet Mask.
- Default Gateway: 192.168.0.1
- Services < AAA < Client Name: Home | Client IP: 192.168.0.1 | Secret: password | Click Add.
- Service: click On.

- În User Setup adaugă pe rând username-ul și parola fiecărui device, ex: Username: SmartPhone | Password: SmartPhone
- IoT Server < Service: On

#### 1.4. Configurare router

- **Router** < GUI < Wireless < Basic Wireless Settings < Network Name (SSID) : Home
- Click Save Settings
- Wireless Security < Security Mode: WPA2-Enterprise | Radius Server: 192.168.0.10 | Shared Secret: password
- Click Save Settings

#### 1.5. Pentru fiecare device pe care îl vom conecta la rețea vom proceda astfel

- Click the device you want to connect < Config
- Wireless0 < SSID: Home | Authentication: WPA2-Enterprise | Set specific credentials: Username: **SmartPhone** | Password: **SmartPhone**

#### 1.6. Configurare laptop

- **Laptop** < Desktop < IP Configuration < DHCP
- Desktop < Web Browser < URL: 192.168.0.10 (the IoT Server IP address) < Go < Sign Up Now < Registration Server Account Creation < Username: admin | Password: admin < Create

#### 1.7. Pentru a adauga fiecare device la serverul de IOT vom proceda astfel

- Click the device you want to connect < Config
- Settings < IoT Server < Remote Server < Server Address: 192.168.0.10 | Username: admin | Password: admin < Connect

#### 1.8. Pentru a schimba stare unui device vom proceda astfel

- **Laptop** < Desktop < IoT Monitor < IoT Server Address: 192.168.0.10 | Username: admin | Password: admin
- Selectam din lista afișată device-ul a cărui stare dorim ca să o schimbăm

## 2. Protocoale de conectivitate folosite

2.1. WiFi: Folosim WiFi în cea mai mare parte, fiind protocolul de care depinde proiectul nostru, fara de el nu am putea conecta device-urile la rețeaua locală de internet, să le acordăm accesul la internet sau ca device-urile să comunice între ele.

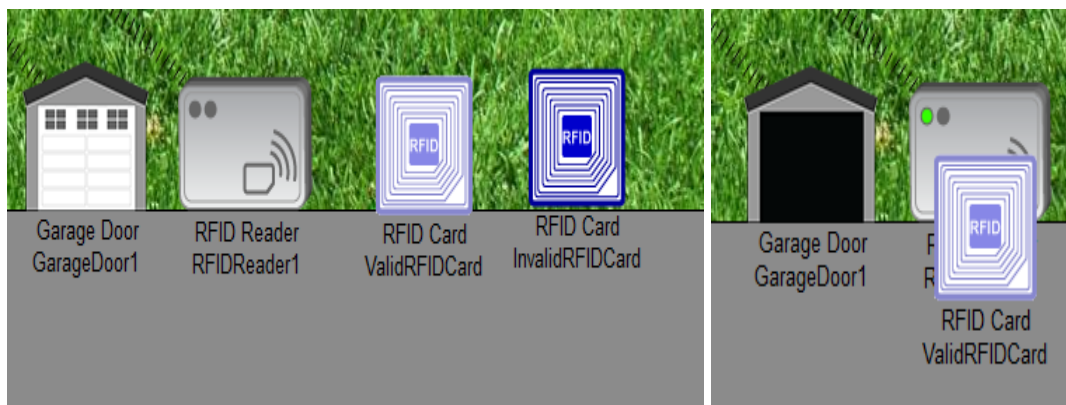
2.2. Bluetooth: Tehnologie de comunicație fără fir care permite dispozitivelor să se conecteze și să comunice între ele la distanțe scurte. În Packet Tracer, utilizăm

acest protocol pentru a stabili conexiuni între diverse dispozitive, cum ar fi smartphone-uri, sisteme audio sau alte dispozitive compatibile cu Bluetooth.

- 2.3. RFID (Radio-Frequency Identification): RFID este o tehnologie de identificare și urmărire bazată pe semnale radio. În Packet Tracer, putem simula utilizarea tehnologiei RFID prin adăugarea cititoarelor RFID și a etichetelor RFID la dispozitive. Aceasta vă permite să monitorizați și să identificați obiecte sau persoane prin intermediul semnalelor radio emise și recepționate între cititor și cardul RFID.

### 3. Funcionalitati (presupunem ca device-urile au fost deja conectate la WiFi)

#### 3.1. Deschidere usa garaj pe bază de cititor RFID



- Vom avea nevoie de un cititor RFID, o usa de garaj și 2 carduri RFID
- Conectam cititorul RFID și usa de garaj prin WiFi la server
- Vom accesa serverul de IOT folosind Laptopul sau Smartphone-ul unde vom implementa regulile din imaginea de mai jos (apasand pe tabul **Conditions** apoi pe butonul **Add**)

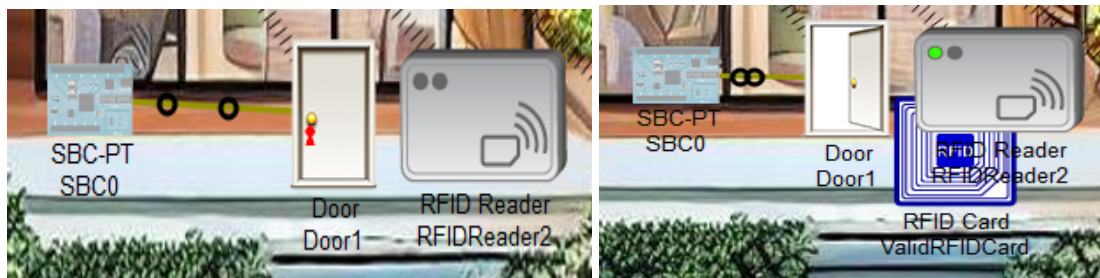
Edit Remove	Yes	RFIDCard1Waiting	RFIDReader1 Card ID = 0	Set RFIDReader1 Status to Waiting
Edit Remove	Yes	RFIDCard2Waiting	RFIDReader2 Card ID = 0	Set RFIDReader2 Status to Waiting
Edit Remove	Yes	RFIDCard1Valid	RFIDReader1 Card ID is between 1000 and 2000	Set RFIDReader1 Status to Valid
Edit Remove	Yes	RFIDCard2Valid	RFIDReader2 Card ID is between 1000 and 2000	Set RFIDReader2 Status to Valid
Edit Remove	Yes	RFIDCard1Invalid	RFIDReader1 Card ID = 2001	Set RFIDReader1 Status to Invalid
Edit Remove	Yes	RFIDCard2Invalid	RFIDReader2 Card ID = 2001	Set RFIDReader2 Status to Invalid

- Apoi vom lua doua carduri RFID pe care le vom numi ValidRFIDCard and InvalidRFIDCard, cărora le vom asigna ID-urile 1001, respectiv 2001 astfel: Click RFIDCard < Attributes < Introducem în casuta CardID valoarea pentru cardul nostru RFID < Refresh.

- Pentru ca usa de garaj sa se deschidă și să se inchida cand cititorul RFID nu mai detectează niciun card valid vom impune următoarea regulă:

Edit Remove	Yes	GarageDoorOn	RFIDReader1 Status is Valid	Set GarageDoor1 On to true
Edit Remove	Yes	GarageDoorOff	RFIDReader1 Status is Waiting	Set GarageDoor1 On to false

### 3.2. Descuriere a ușii de la intrare a casei pe bază de cititor RFID



- Vom avea nevoie de un cititor RFID, o usa și o placuta SBC (Single Board Computer)
- Conectam cititorul RFID și ușa de la casa prin WiFi la server
- Conectăm placuta SBC (slot D0) la ușa de la intrare (slot D0) printr-un cablu IOT
- Pentru ca usa sa se descurie cand atingem cititorul RFID cu cardul RFID vom implementa următoarea regulă:

Edit Remove	Yes	DoorOpen	RFIDReader2 Status is Valid	Set Door1 Lock to Unlock Set Light1 Status to Dim
----------------	-----	----------	-----------------------------	--

- Deoarece ușa are 2 stadii locked/unlocked și open/closed, pentru a face ca usa sa se si deschida automat odată cu atingerea cardului RFID la cititor, vom proceda astfel: Click SBC < Programming < New < Python Project și vom adăuga cod Python de mai jos < Run.

```
from gpio import *
from time import *
```

```
def main():
```

```
    while True:
```

```
        # Read the state of the contact sensor
        contact_sensor_state = customRead(0)
```

```
        # If the contact sensor is closed (door is locked), OPEN
        the DOOR
```

```
        if contact_sensor_state == "0,0":
```

```

customWrite(0, "1,0");
delay(5000);
customWrite(0, "0,1");
# delay(1000);

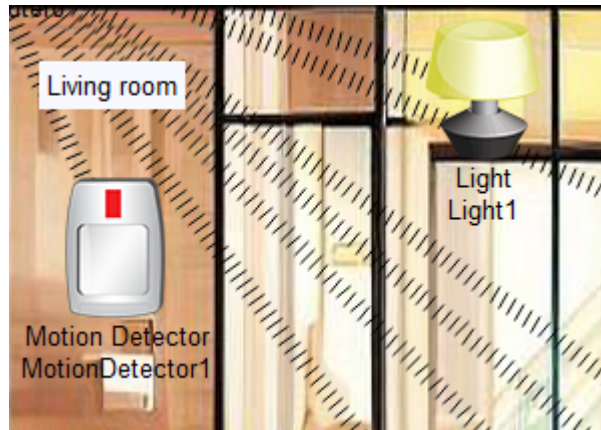
```

```

if __name__ == "__main__":
    main()

```

### 3.3. Aprinderea becului din living room/dormitor pe baza de mișcare



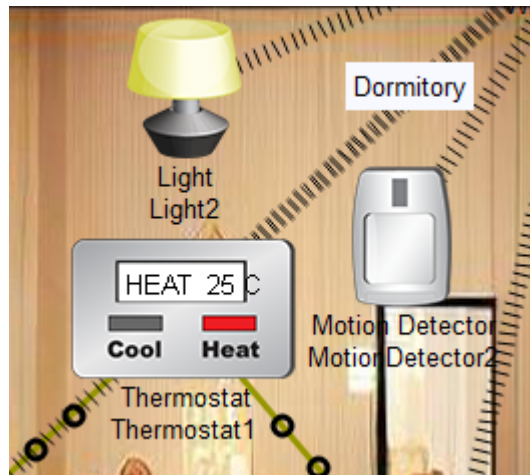
- Vom avea nevoie de 1 bec și de un senzor de mișcare pe care le vom conecta la serverul de IOT prin WiFi.
- Vom impune următoarea regulă pentru living room:

<input type="button" value="Edit"/> <input type="button" value="Remove"/>	Yes	Light1On	MotionDetector1 On is true	Set Light1 Status to On
	Yes	Light1Off	MotionDetector1 On is false	Set Light1 Status to Off

- Vom impune următoarea regulă pentru dormitor:

<input type="button" value="Edit"/> <input type="button" value="Remove"/>	Yes	Light2On	MotionDetector2 On is true	Set Light2 Status to On
--	-----	----------	----------------------------	-------------------------

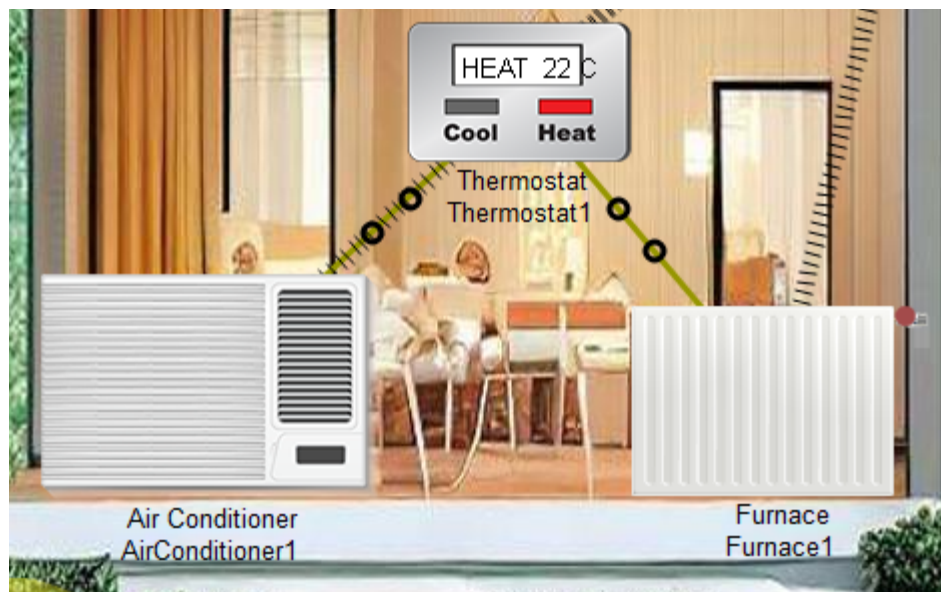
### 3.4. Diminuare bec dormitor



- Pentru scenariul în care este seara, ne aflăm în dormitor, detectorul de mișcare nu mai primește date, pentru că am adormit, putem face ca becul să-și diminueze luminozitatea transformându-se într-o veioză.
- Vom impune următoarea regulă:

Edit	Yes	Light2Off	MotionDetector2 On is false	Set Light2 Status to Dim
Remove				

### 3.5. Setare automată a temperaturii



- Pentru o temperatură setată după placul nostru vom avea nevoie de un ventilator, un termostat, un AC și un calorifer conectate prin cablu IOT: termostat (slot D2) - AC (slot D0), termostat (slot D1) - calorifer (slot D0).
- După ce am conectat toate aceste device-uri la serverul de IOT prin WiFi vom impune următoarele reguli:

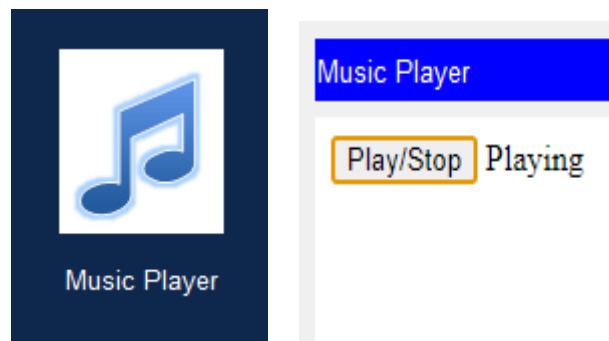


Actions	Enabled	Name	Condition	Actions
<div>Edit</div> <div>Remove</div>	Yes	TurnHeatOn	Thermostat1 Temperature <= 19.0 °C	Set Thermostat1 Status to Heating Set Window1 On to false Set Window2 On to false Set Fan1 Status to Off
<div>Edit</div> <div>Remove</div>	Yes	TurnACOn	Thermostat1 Temperature >= 28.0 °C	Set Thermostat1 Status to Cooling Set Window1 On to true Set Window2 On to true Set Fan1 Status to High

### 3.6. Ascultare muzica la system audio in studio prin Bluetooth



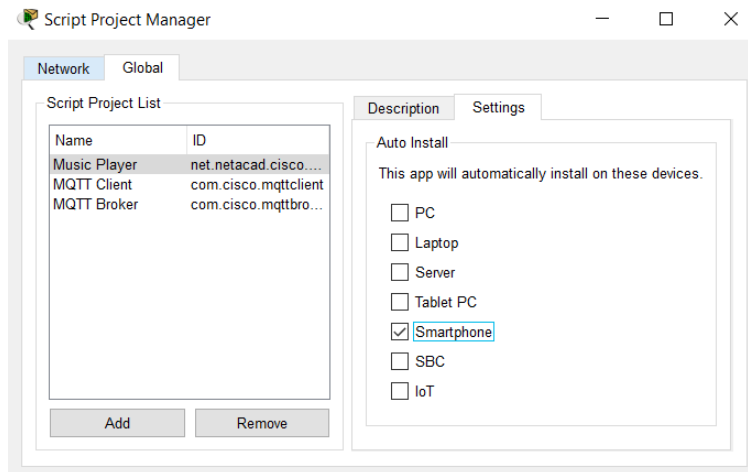
- În cazul în care suntem niște iubitori de muzica la calitate buna, putem asculta muzica la boxele din studio conectate la telefon prin Bluetooth. Pentru a da play muzicii vom accesa de pe smartphone aplicația Music Player apoi vom apasa **Play/Stop**.



- Pentru a schimba melodia va trebui ca sa intram in proiectul **net.netacad.cisco.example\_music\_player**, selectam fisierul **main.js**, apoi vom merge la linia **29**, de unde vom putea schimba adresa melodiei pe care dorim ca să o ascultăm.

```
29      btService.send(dstMac, dstService, "../Sounds/Smiley - Vals.wav");
```

- Obs: Aplicația Music Player nu este disponibilă by default pe smartphone așa că va trebui importată. În Cisco Packet Tracer vom da click pe Tools < Script Project Manager < Global < Add < Căutați unde ați descărcat din arhiva dată de mine: **app.pkp**. Apoi selectați Music Player și în partea dreaptă bifați **Smartphone**

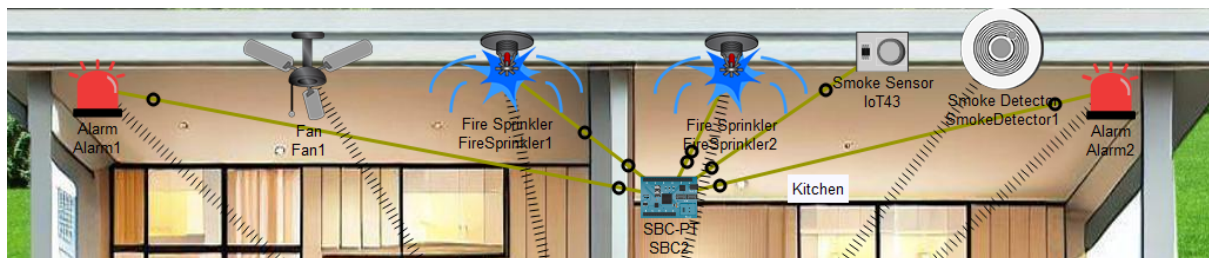


### 3.7. Pornire automată aparat cafea

- Sa zicem ca suntem un bautor de cafea dimineata dar nu ne place ca sa asteptam mult după cafea, putem ca sa ne folosim de un aparat de cafea și de un senzor de miscare conectate la serverul de IOT prin WiFi. Astfel ca atunci cand ne vom trezi prima data senzorul de mișcare ne va simți prezența și va porni aparatul de cafea
- Vom impune următoarea regulă:

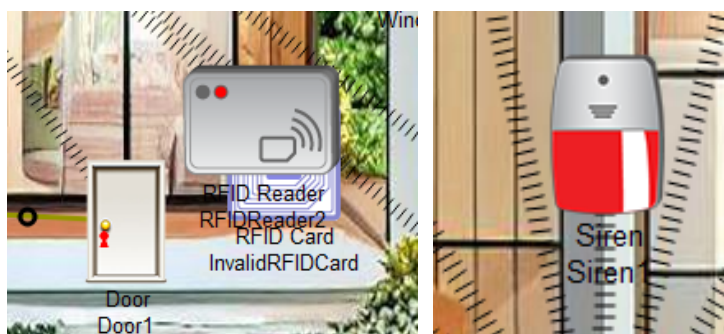
Edit Remove	Yes	MakeCoffee	MotionDetector3 On is true	Set CoffeeMaker On to true
Edit Remove	Yes	StopMakingCoffee	MotionDetector3 On is false	Set CoffeeMaker On to false

### 3.8. Sistem antiincendiu



- Pentru a preveni un incendiu vom avea nevoie de 2 stropitori de incendiu, 2 alarme conectate la WiFi, un senzor de fum și o placuta SBC pe care o vom conecta astfel:

### 3.9. Sistem antiefracție

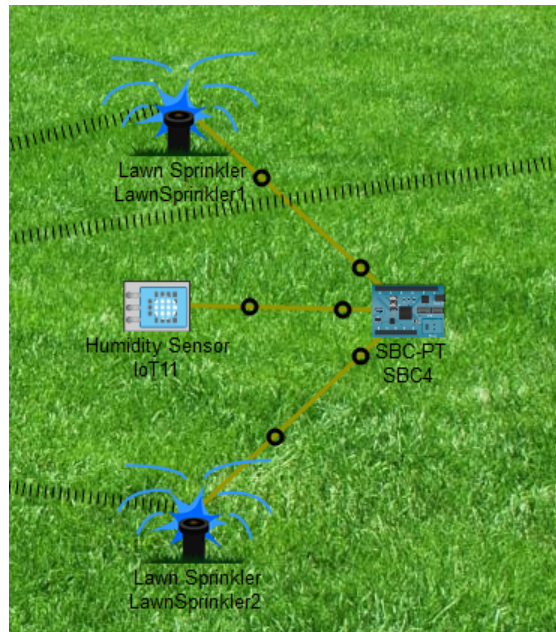




- În cazul în care cineva va încerca ca să intre în casa noastră folosind un card RFID invalid, vom putea folosi o sirena ca să ne anunțe
- Vom impune următoarea regulă:

Edit	Yes	DoorHacking	RFIDReader2 Status is Invalid	Set Siren1 On to true
Remove				

### 3.10. Automatizare udare a gazonului



- În cazul în care suntem foarte ocupați sau uităm mai mereu ca să udăm gazonul putem folosi un senzor de umiditate, o plăcuță SBC și 2 stropitoare pentru gazon.
- Vom conecta dispozitivele astfel: Senzor (Slot D0) - SBC (Slot D0), Stropitoare 1 (Slot D0) - SBC (Slot D1), Stropitoare 1 (Slot D0) - SBC (Slot D2). Conectăm și la WiFi stropitorile
- Vom folosi următorul cod pentru SBC:

```
from gpio import *
from time import *
from pyjs import *
import math

def main():

    while True:
        # Read the state of the humidity sensor
        humidity_state = digitalRead(0)
        print("Humidity state: " + str(humidity_state))
        humidity_percentage =
        math.floor(js_map(humidity_state, 0, 255, 0, 100))
```

```
print("Humidity percentage: " +
str(humidity_percentage));
```

```
# If the humidity sensor is under a level then:
```

```
if humidity_percentage <= 330:
```

```
    customWrite(1, "1")
```

```
    customWrite(2, "1")
```

```
    delay(5000);
```

```
    customWrite(1, "0")
```

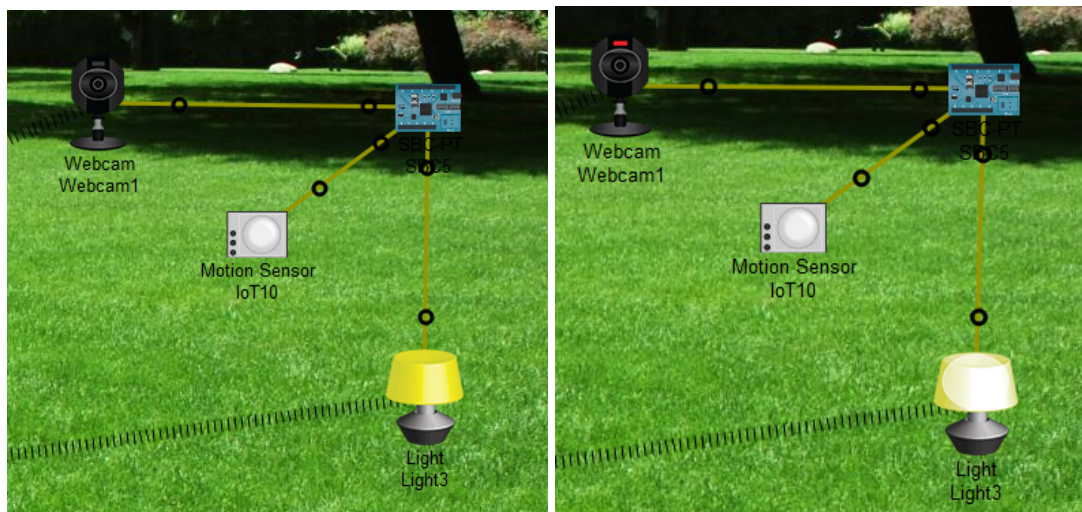
```
    customWrite(2, "0")
```

```
    delay(10000);
```

```
if __name__ == "__main__":
```

```
    main()
```

### 3.11. Sistem supraveghere gradina



- Pentru a putea avea o privire de ansamblu asupra gradinii chiar de pe telefon putem folosi un senzor de mișcare, o camera web și o lumina pentru a ajuta camera web ca capteze imagini pe timp de noapte.
- Vom conecta astfel: Senzor (Slot D0) - SBC (Slot D0), Webcam (Slot D0) - SBC (Slot D1), Lumina (Slot D0) - SBC (Slot D2). Conectăm și la WiFi camera web și lumina
- Vom folosi următorul cod pentru SBC:

```
from gpio import *
```

```
from time import *
```

```
def main():
```

```
    while True:
```

```

# Read the state of the motion sensor
motion_state = digitalRead(0);
print("Motion_state: " + str(motion_state))

# If the motion sensor is off, then
if motion_state > 0:
    #Turn on the webcam
    customWrite(1, "1")
    #Turn on the light
    customWrite(2, "2")
    delay(5000)

#Put the devices in standby
customWrite(1, "0")
customWrite(2, "0")

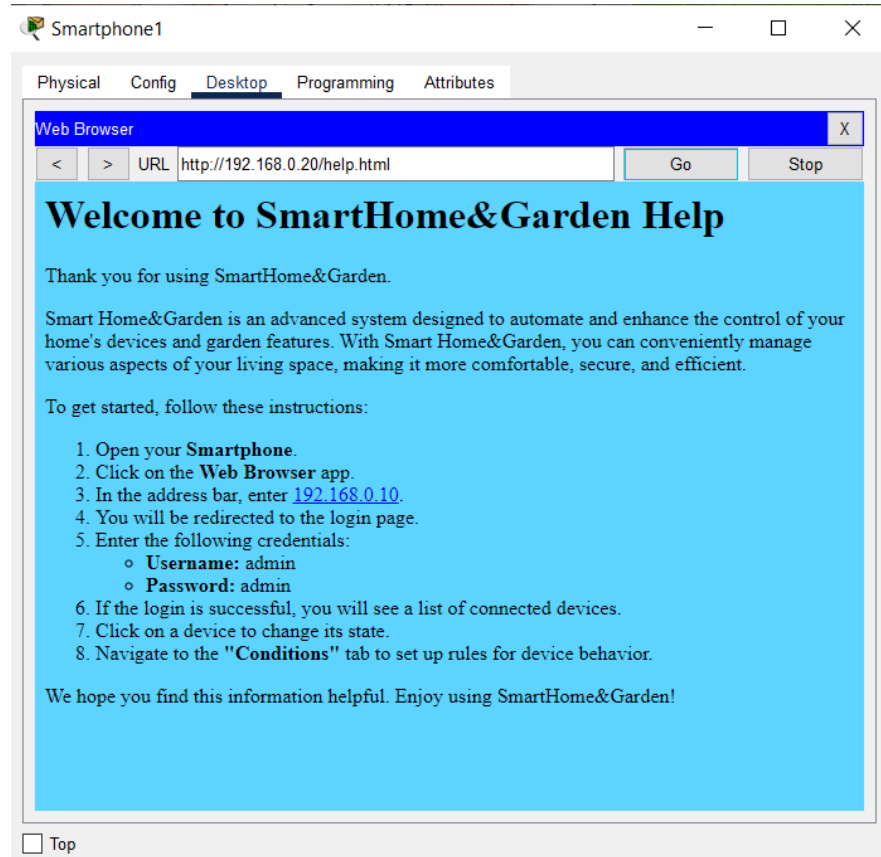
if __name__ == "__main__":
    main()

```

#### 4. Modalități de controlare online a casei

##### 4.1. Accesare pagina help

În cazul în care uităm cum sa utilizam sistemul putem întotdeauna ca sa accesam pagina noastra de help: 192.168.0.20/help.html pentru un quick reminder.



#### 4.2. Laptop & Smartphone

După cum am mai precizat Cisco Packet Tracer oferă diverse posibilități de control a dispozitivelor folosite, fie prin acces direct la serverul de IOT, de unde putem ca să schimbăm stadiul unui device fie putem ca să implementăm diverse reguli pentru a automatiza diverse procese. De asemenea Cisco Packet Tracer oferă și posibilitatea de a controla device-urile prin scriere de cod Python sau JavaScript, pentru a îndeplini diverse sarcini sau a ajuta acolo unde pagina serverului de IOT prezinta limitari. (de exemplu în cazul ușii: pentru aceasta nu se poate impune o regula ca ușă să se și deschidă, astfel a fost necesara adăugarea unei plăcuțe SBC).

