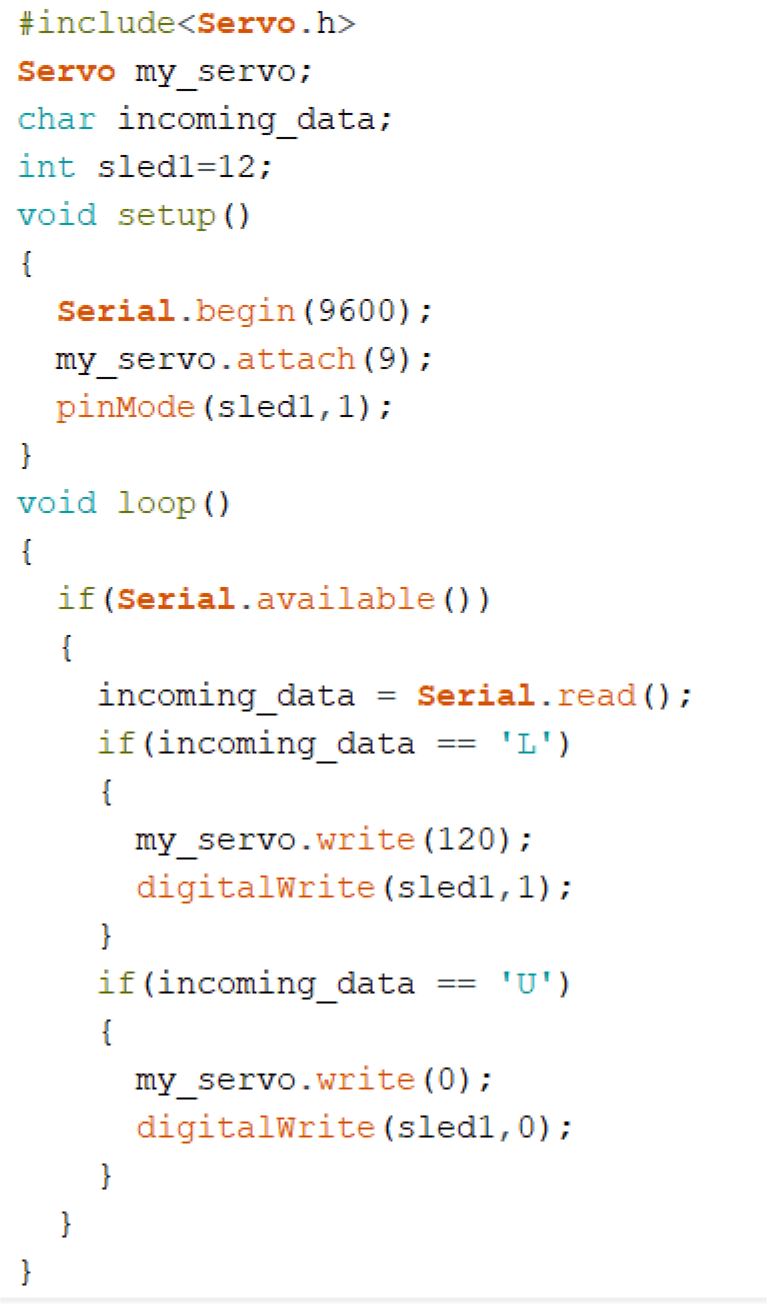
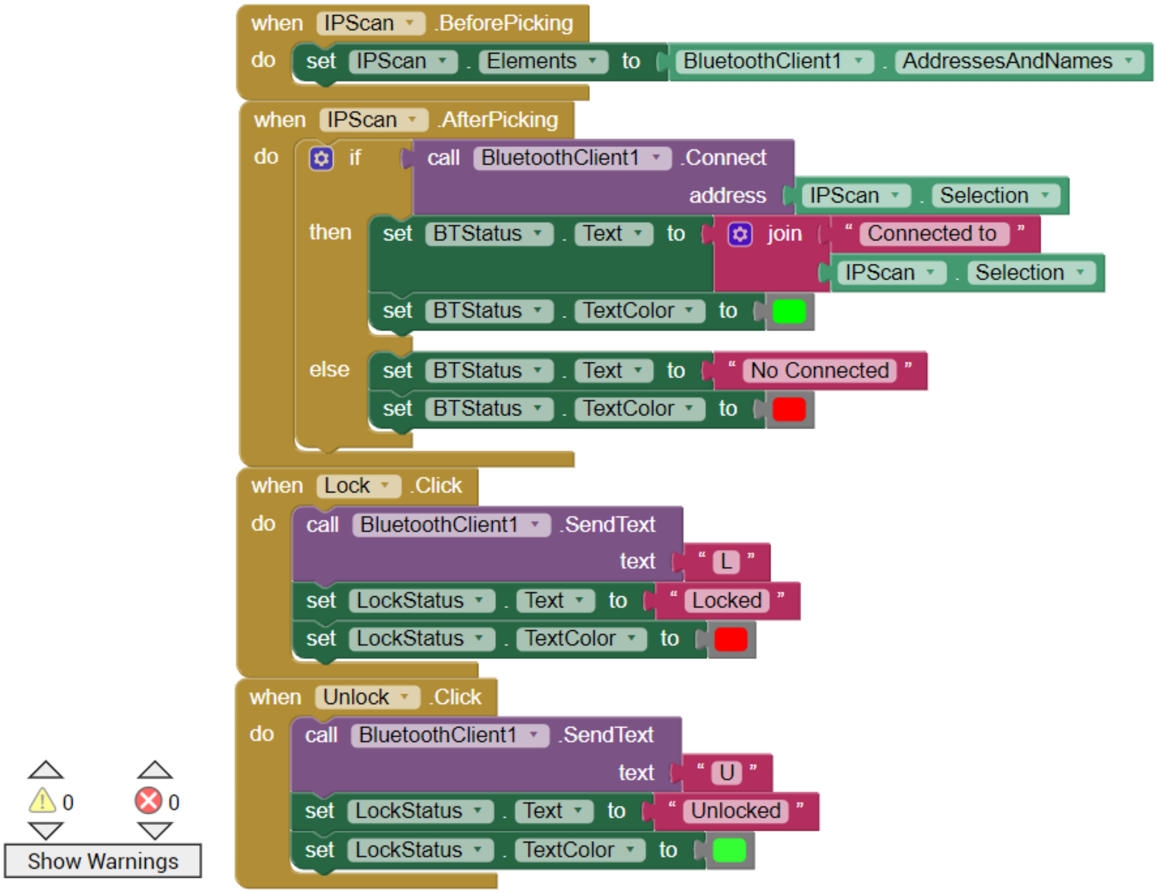
SMART DOOR LOCKING SYSTEM USING ARDUINO UNO:

Arduino code :-



```
#include<Servo.h>
Servo my_servo;
char incoming_data;
int sled1=12;
void setup()
{
  Serial.begin(9600);
  my_servo.attach(9);
  pinMode(sled1,1);
}
void loop()
{
  if(Serial.available())
  {
    incoming_data = Serial.read();
    if(incoming_data == 'L')
    {
      my_servo.write(120);
      digitalWrite(sled1,1);
    }
    if(incoming_data == 'U')
    {
      my_servo.write(0);
      digitalWrite(sled1,0);
    }
  }
}
```

MIT App inventor code:



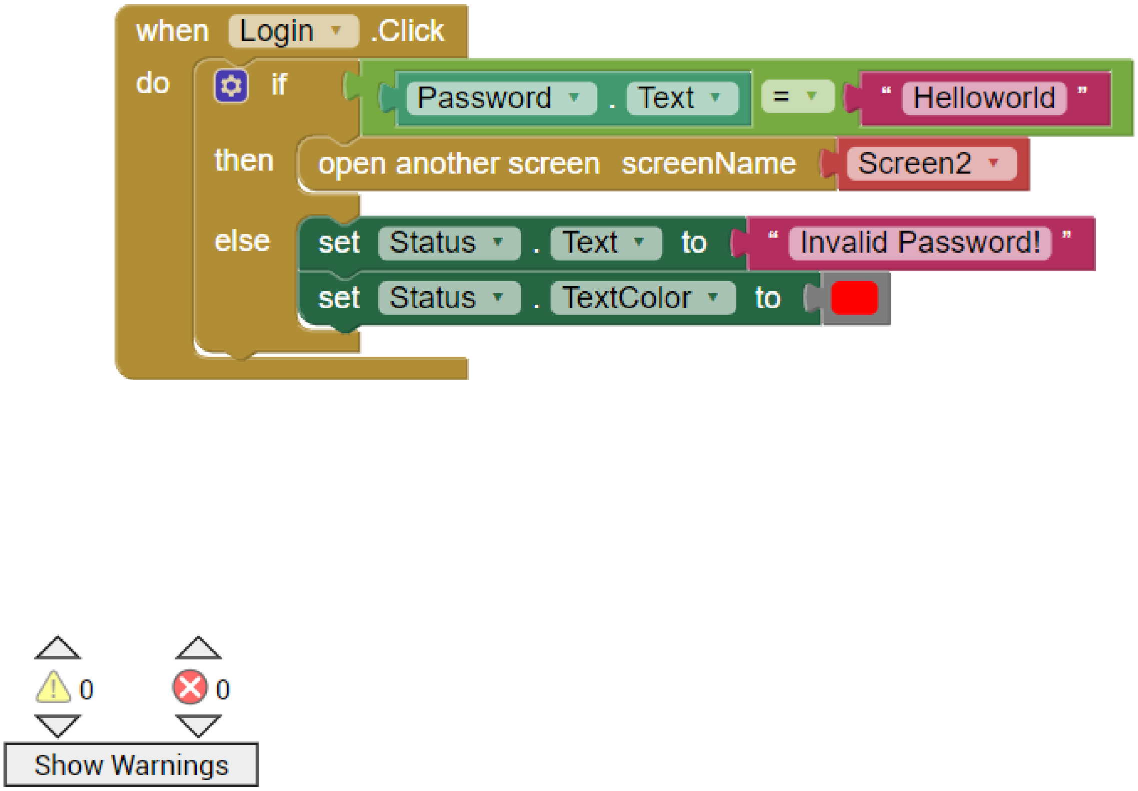
```
when IPScan.BeforePicking
do  set IPScan.Elements to BluetoothClient1.AddressesAndNames

when IPScan.AfterPicking
do  if   call BluetoothClient1.Connect
                        address IPScan.Selection
    then set BTStatus.Text to join " Connected to "
                                   IPScan.Selection
         set BTStatus.TextColor to 🟩
    else set BTStatus.Text to " No Connected "
         set BTStatus.TextColor to 🟥

when Lock.Click
do  call BluetoothClient1.SendText
                        text " L "
    set LockStatus.Text to " Locked "
    set LockStatus.TextColor to 🟥

when Unlock.Click
do  call BluetoothClient1.SendText
                        text " U "
    set LockStatus.Text to " Unlocked "
    set LockStatus.TextColor to 🟩
```

Control Page

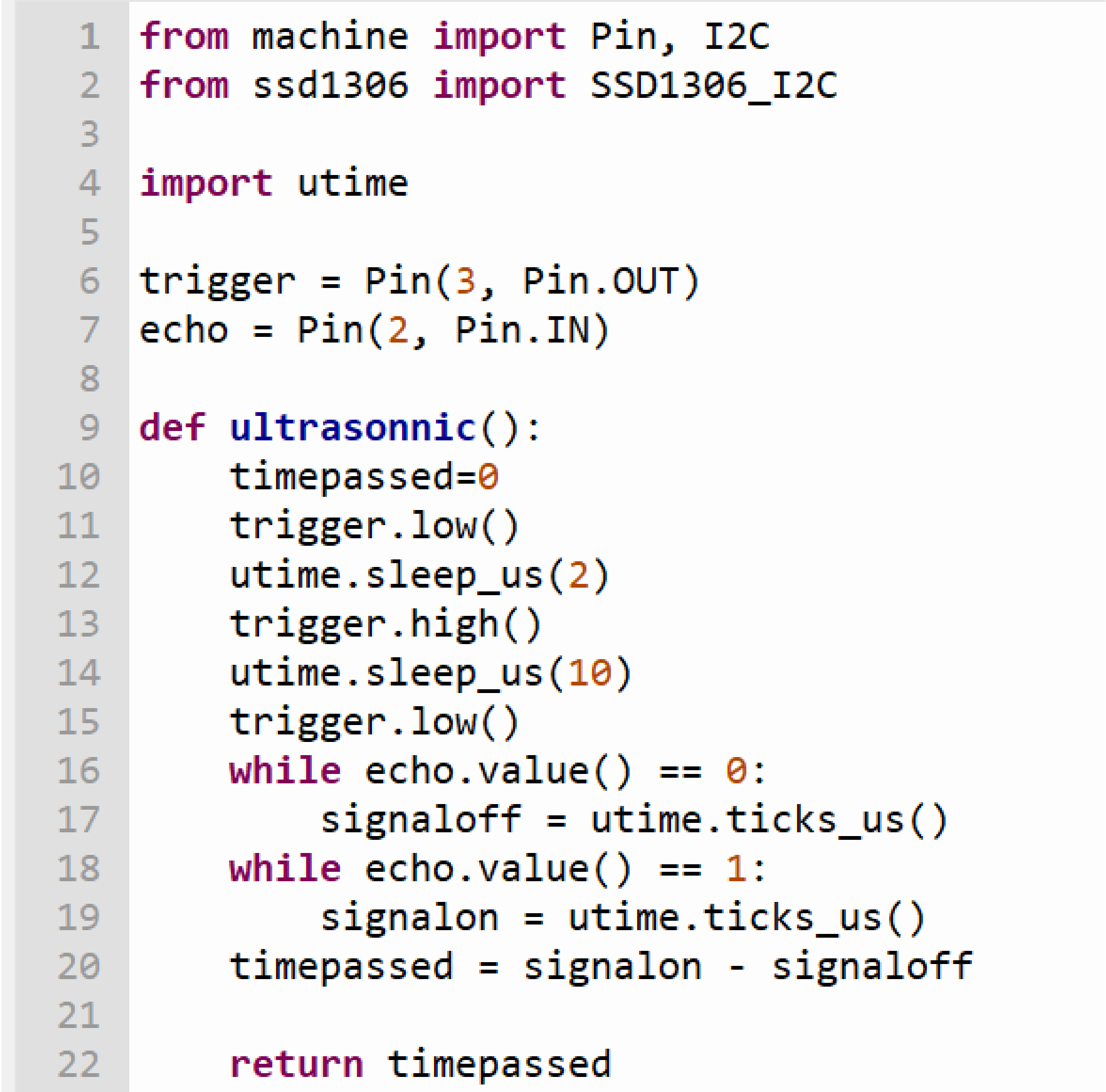


```
when Login .Click
do   if   Password . Text  =  " Helloworld "
     then open another screen  screenName  Screen2
     else set Status . Text to " Invalid Password! "
          set Status . TextColor to 🟥
```

Homepage

HC-SR04 ULTRASONIC DISTANCE SENSOR USING RPI:

Thonny Code:-



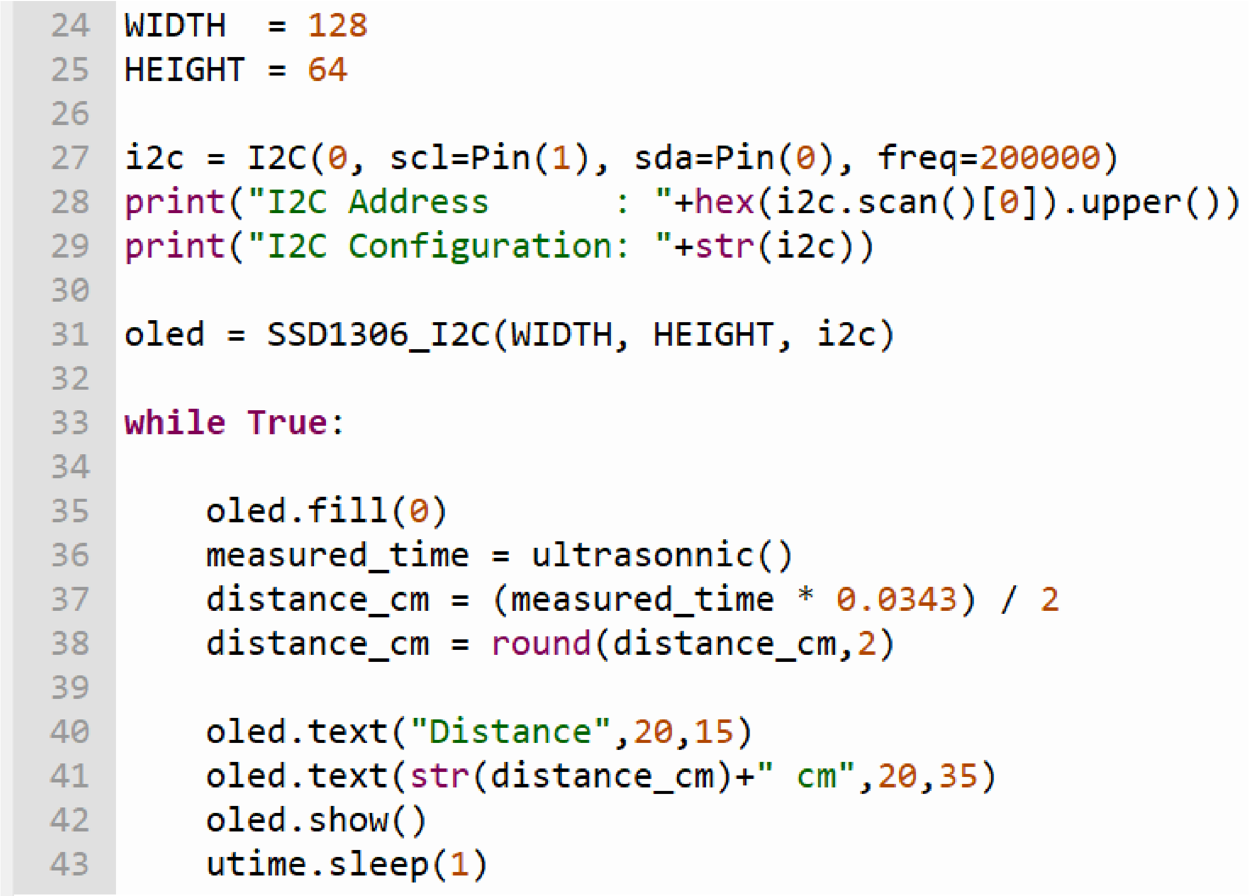
```python
from machine import Pin, I2C
from ssd1306 import SSD1306_I2C

import utime

trigger = Pin(3, Pin.OUT)
echo = Pin(2, Pin.IN)

def ultrasonnic():
    timepassed=0
    trigger.low()
    utime.sleep_us(2)
    trigger.high()
    utime.sleep_us(10)
    trigger.low()
    while echo.value() == 0:
        signaloff = utime.ticks_us()
    while echo.value() == 1:
        signalon = utime.ticks_us()
    timepassed = signalon - signaloff

    return timepassed
```



```
WIDTH  = 128
HEIGHT = 64

i2c = I2C(0, scl=Pin(1), sda=Pin(0), freq=200000)
print("I2C Address      : "+hex(i2c.scan()[0]).upper())
print("I2C Configuration: "+str(i2c))

oled = SSD1306_I2C(WIDTH, HEIGHT, i2c)

while True:

    oled.fill(0)
    measured_time = ultrasonnic()
    distance_cm = (measured_time * 0.0343) / 2
    distance_cm = round(distance_cm,2)

    oled.text("Distance",20,15)
    oled.text(str(distance_cm)+" cm",20,35)
    oled.show()
    utime.sleep(1)
```