ARDUINO CODE  
// Includes the Servo and lcd library

#include <Servo.h>

#include <dht11.h>

#define DHT11PIN 4

dht11 DHT11;

//Defines lcd pins

// Defines Tirg and Echo pins of the Ultrasonic Sensor

const int trigPin = 10;

const int echoPin = 11;

const int sound= A0;

//Defines piezo pin

const int piezoPin = 8;

// Variables for the duration and the distance

long duration;

int distance;

int notes[] = {262, 462, 862, 1662, 3262}; // Enter here the notes you like

Servo myServo; // Creates a servo object for controlling the servo motor

void setup() {

pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin, INPUT); // Sets the echoPin as an Input

pinMode(sound, INPUT);

Serial.begin(9600);

myServo.attach(12); // Defines on which pin is the servo motor attached

}

void loop() {

int soundValue = analogRead(soundSensorPin); // Read the analog value from the Sound Sensor

int threshold = 100;

// rotates the servo motor from 15 to 165 degrees

for(int i=15;i<=165;i++){

myServo.write(i);

if(soundValue > threshold){

tone(piezoPin, notes[1]);

delay(1000);

noTone(piezoPin);

delay(3000);

}

int chk = DHT11.read(DHT11PIN);

if (DHT11.temprature > tempthreshold){

tone(piezoPin, 2500);

delay(1000);

noTone(piezoPin);

delay(5000);

}

distance = calculateDistance();// Calls a function for calculating the distance measured by the Ultrasonic sensor for each degree

//beep sequence

if(distance > 40){

noTone(piezoPin);

delay(10);

noTone(piezoPin);

delay(30);

}

else if (distance <= 40 && distance > 30){

tone(piezoPin, notes[1]);

delay(1000);

noTone(piezoPin);

delay(30);

}

else if (distance <= 30 && distance > 20){

tone(piezoPin,notes[2]);

delay(500);

noTone(piezoPin);

delay(30);

}

else if (distance <= 20 && distance > 10){

tone(piezoPin,notes[3]);

delay(100);

noTone(piezoPin);

delay(30);

}

else {

tone(piezoPin,notes[4]);

delay(10);

noTone(piezoPin);

delay(30);

}

Serial.print(i); // Sends the current degree into the Serial Port

Serial.print(","); // Sends addition character right next to the previous value needed later in the Processing IDE for indexing

Serial.print(distance); // Sends the distance value into the Serial Port

Serial.print("."); // Sends addition character right next to the previous value needed later in the Processing IDE for indexing

}

// Repeats the previous lines from 165 to 15 degrees

for(int i=165;i>15;i--){

myServo.write(i);

distance = calculateDistance();

if(distance > 40){

noTone(piezoPin);

delay(10);

noTone(piezoPin);

delay(30);

}

else if (distance <= 40 && distance > 30){

tone(piezoPin, notes[4]);

delay(10);

noTone(piezoPin);

delay(30);

}

else if (distance <= 30 && distance > 20){

tone(piezoPin,notes[3]);

delay(10);

noTone(piezoPin);

delay(30);

}

else if (distance <= 20 && distance > 10){

tone(piezoPin,notes[2]);

delay(10);

noTone(piezoPin);

delay(30);

}

else {

tone(piezoPin,notes[1]);

delay(10);

noTone(piezoPin);

delay(30);

}

Serial.print(i);

Serial.print(",");

Serial.print(distance);

Serial.print(".");

}

}

// Function for calculating the distance measured by the Ultrasonic sensor

int calculateDistance(){

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH); // Reads the echoPin, returns the sound wave travel time in microseconds

//U(m/s)=dX(m)/dT(s)

//in this case Duration(time)= 2\*Distance/SpeedOfSound=>

//Distance=SpeedOfSound\*Duration/2

// In dry air at 20 °C, the speed of sound is 343.2 m/s or 0.003432 m/Microsecond or 0,03434 cm/Microseconds

distance= duration\*0.034/2;

return distance;

}