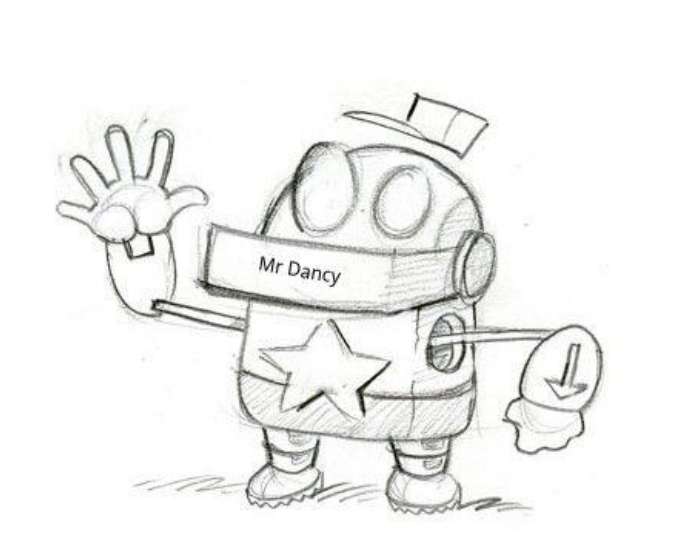
MULTIMEDIA EMBEDDED SYSTEMS REPORT

 **Erdem Durmaz-245907/Asil Karakulak- 245872**

18.06.2019

# Project Desciption of last design

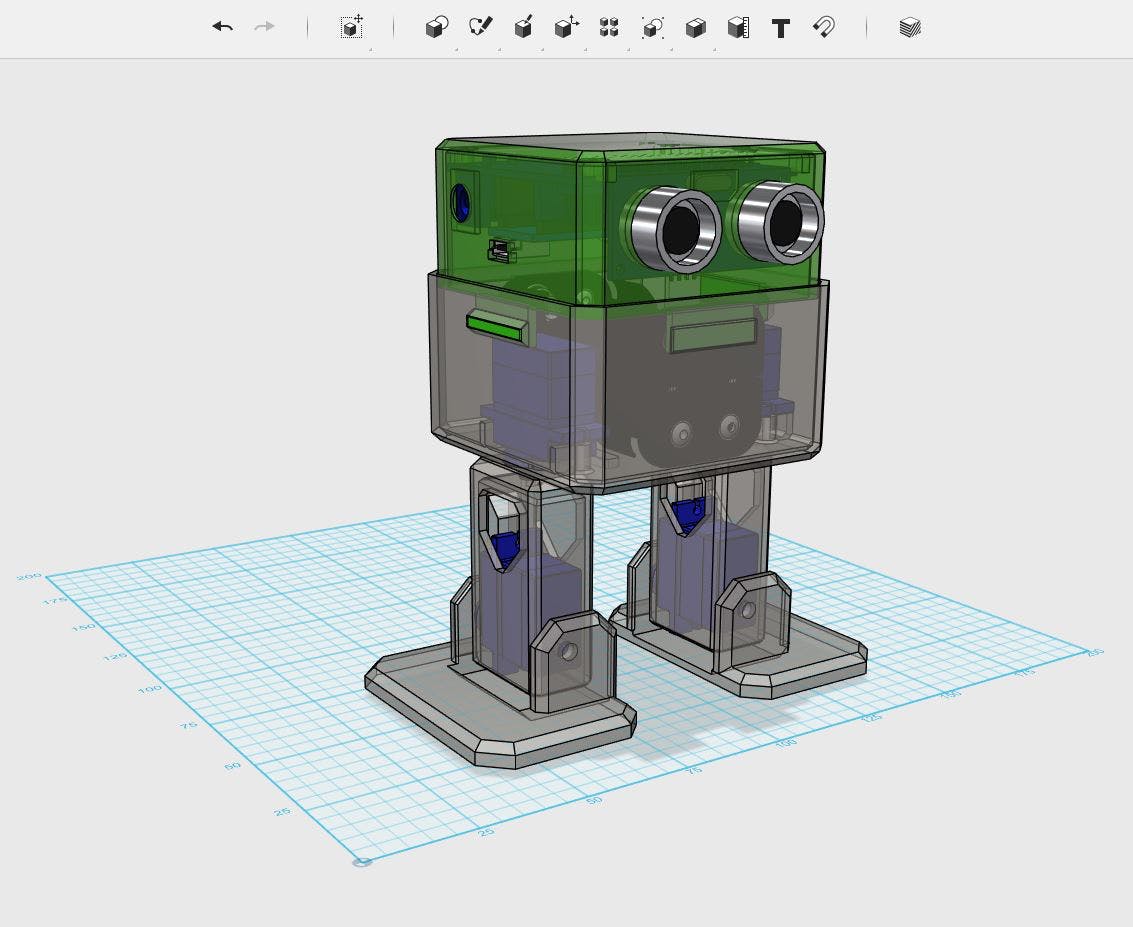
L.This is a robot can sense the object and dance with them.It has an ultrasonic sensor inside of him. Ultrasonic sensor transmits data through bluetooth to the app and via terminal it’s possible to how far away is the object.

# 

# 

Objects closer than 20cm will be categorized as something to dance[partner]

And Mr Dancy will start to dance after that he will go back and give some personal space to himself .3D printed MODEL



# 

|  |  |  |
| --- | --- | --- |
| Status | DAta | requirement |
| Looking for a partner | CM | >=20CM |
| Partner has been found | CM | <=20CM |
| Give some space | CM | Going back 4 Steps 25cm |

# Sheet regarding workload

|  |  |  |
| --- | --- | --- |
| person | aim | hour |
| Erdem Durmaz-245907 | Recalibration of robot,Dance code,Sensor Coding, Bluetooth  Synch, All software regarding robot | 68.5 |
| Muhammet asil Karakulak | Assembly of the Hardware, Project Planning, Equipment support, orders, Logistic Support, Testing written code | 46 |
| Can vural  [was not present in final day] | App design[not used] | No data provided |

# Data provided by toggle time tracker

# 

# 

# 

# Code

distance= distance/58;

Serial.print(distance);

Serial.println("cm");

This is the code regarding ultrasonic sensor , and how it works

if(distance<20){

obstacleDetected = true;

Serial.println ("There You GO, a partner! LETS DANCE");

}else{

obstacleDetected = false;

Serial.println ("Well looking for someone to dance is there anyone?");

}

Second part Regarding dance and how otto reacts when there is entity

if(obstacleDetected){

Otto.sing(S\_surprise);

Otto.playGesture(OttoFretful);

Otto.playGesture(OttoFretful);

Otto.sing(S\_fart3);

Otto.walk(3,1600,1);

Otto.turn(34,1300,-1);

delay(50);

obstacleDetector();

}

else{

Otto.walk(1,1000,-1);

obstacleDetector();

}

}

This is the code which makes calibration with EEPROM memory also with global variables, pin locations on the shell.

#define N\_SERVOS 4

#define EEPROM\_TRIM true //Activate for calibration

#define TRIM\_RR 0

#define TRIM\_RL 40

#define TRIM\_YR 4

#define TRIM\_YL -7

#define PIN\_RR 2

#define PIN\_RL 3

#define PIN\_YR 4

#define PIN\_YL 5

#define INTERVALTIME 10.0

#define ECHOPIN 9 // Pin to receive echo pulse

#define TRIGPIN 8 // Pin to send trigger pulse

///////////////////////////////////////////////////////////////////

//-- Global Variables -------------------------------------------//

///////////////////////////////////////////////////////////////////

//-- Movement parameters

int T=1000; //Initial duration of movement

int moveId=0; //Number of movement

int moveSize=15; //Asociated with the height of some movements

//---------------------------------------------------------

bool obstacleDetected = false;

///////////////////////////////////////////////////////////////////

//-- Setup ------------------------------------------------------//

///////////////////////////////////////////////////////////////////

void setup(){

# CONCLUSION AND FINAL

Mr Dancy could be a nice tool for those who wants a robot which can detect objects and dance with them childs and home animals pets also might enjoy with the tool.

Project has ben shrinked because of mathematical also hardware problems regarding to sound recognition. There now it senses objects to dance with them